UNIVERZITA HRADEC KRÁLOVÉ

Pedagogická fakulta

Katedra aplikované kybernetiky Přírodovědecké fakulty

Historické šifry jako motivace k výuce programování

Diplomová práce

Autor:	Bc. Pavel Musílek
Studijní program:	N7504 – Učitelství pro střední školy
Studijní obor:	Učitelství pro střední školy – dějepis
	Učitelství pro střední školy – informatika
Vedoucí práce:	doc. RNDr. Štěpán Hubálovský, Ph.D.

Hradec Králové **2020**

Univerzita Hradec Králové
Pedagogická fakulta

Zadání diplomové práce

Autor:	**Pavel Musílek**
Studium:	P17P0815
Studijní program:	N7504 Učitelství pro střední školy
Studijní obor:	Učitelství pro střední školy - dějepis, Učitelství pro střední školy - informatika
Název diplomové práce:	**Historické šifry jako motivace k výuce programování**
Název diplomové práce AJ:	Using historical cyphers as a motivation to teach programming

Cíl, metody, literatura, předpoklady:

Cílem teoretické části práce je zmapovat historii šifrování s důrazem na klasické ruční substituční šifry od starověku po konec 20. století. Historii šifrování lze vnímat jako souboj mezi tvůrci šifrových systémů (kryptografy) a učenci, využívajícími znalosti z oblasti jazykovědy, matematiky a logiky k luštění šifrových textů bez znalosti použitého systému nebo hesla (kryptoanalytiky). Cílem praktické části práce je historické šifry využít jako motivaci k výuce programování. Praktická část práce bude zpracována formou sady propramátorských úloh různé obtížnosti, které budou sloužit k dešifrování a šifrování daného systému. Úlohy budou následně aplikovány v praxi a podrobeny dotazníkovému šetření u žáků i vyučujících.

BERLOQUIN, Pierre. Skryté kódy a velkolepé projekty. 1. vyd. Praha: Knižní klub, 2011. ISBN 978-80-242-2847-1. MUSÍLEK, Michal. Šifry a kódy [online]. 2010 [cit. 2012-01-10] Dostupné z: SINGH, Simon. Kniha kódů a šifer. 2. vyd. Praha: Argo a Dokořán, 2009. 384 s. ISBN 978-80-7363-268-7 (Dokořán), ISBN 987-80-257-0144-7 (Argo). VONDRUŠKA, Pavel. Kryptologie, šifrování a tajná písma. 1. vyd. Praha: Albatros, 2006. ISBN 80-00-01888-8.

Garantující pracoviště:	Katedra aplikované kybernetiky, Přírodovědecká fakulta
Vedoucí práce:	doc. RNDr. Štěpán Hubálovský, Ph.D.
Oponent:	Ing. Petr Voborník, Ph.D.

Datum zadání závěrečné práce: 7.10.2016

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a že jsem v seznamu použité literatury uvedl všechny prameny, z kterých jsem vycházel.

V Hradci Králové dne 7. 7. 2020 Pavel Musílek

Poděkování

Rád bych mnohokrát poděkoval doc. RNDr. Štěpánu Hubálovskému, Ph.D. za užitečné rady a vstřícný přístup při vedení diplomové práce. Zároveň bych rád poděkoval mému otci, který mě k šifrování přivedl a byl mi vždy dobrým učitelem a rádcem. Současně bych chtěl poděkovat všem dotazovaným respondentům.

Anotace

MUSÍLEK, P. *Historické šifry jako motivace k výuce programování*. Hradec Králové, 2020. Diplomová práce na katedře aplikované kybernetiky Přírodovědecké fakulty Univerzity Hradec Králové. Vedoucí diplomové práce Štěpán Hubálovský.

Cílem teoretické části diplomové práce je mapovat historii šifrování od starověku po přelom 19. a 20. století. Důraz je kladen na ruční substituční šifry, které jsou žáky zpravidla rychleji pochopeny a jejichž vizualizace a prostřednictvím počítačových programů je snadněji realizovatelná, ale dostatečný prostor je věnován také jednoduchým transpozičním šifrám. Historii šifrování lze vnímat jako souboj mezi tvůrci šifrových systémů (kryptografy) a učenci, využívajícími znalosti z oblasti jazykovědy, matematiky a logiky k luštění šifrových textů bez znalosti použitého systému nebo hesla (kryptoanalytiky). Praktické část práce využívá historické šifry jako motivaci k výuce programování. Autor vytvořil ucelenou sadu programátorských úloh různé obtížnosti, a to jak zadání motivující žáky k řešení zajímavých algoritmických problému a komplexnímu procvičování programátorských dovedností, tak vzorová řešení, sloužící jako nástroj šifrování a dešifrování prostřednictvím daného systému a současně jako názorná vizualizace. Zadání úloh spolu s dalšími informacemi je zpracováno ve formě metodické příručky pro učitele. Úlohy byly následně předloženy zkušeným učitelům informatiky s různě dlouhou pedagogickou praxí a podrobeny dotazníkovému šetření prostřednictvím sady otevřených otázek s širokou odpovědí.

Klíčová slova: šifrování, kryptologie, substituce, transpozice, historie, metodika, programování, algoritmizace, vizualizace

Annotation

MUSÍLEK, P. *Using historical cyphers as a motivation to teach programming*. Hradec Králové, 2020. Diploma thesis at the Department of Applied Cybernetics, Faculty of Science, University of Hradec Králové. Thesis supervisor Štěpán Hubálovský.

The aim of the theoretical part of the thesis is to map the history of encryption from antiquity to the turn of the 19th and 20th centuries. Emphasis is placed on manual substitution ciphers, which are usually understood more quickly by students and whose visualization through computer programs is easier to implement, but sufficient space is also devoted to simple transposition ciphers. The history of encryption can be seen as a duel between creators of cryptographic systems (cryptographs) and scholars, using knowledge from linguistics, mathematics and logic to decipher cipher texts without knowledge of the used system or password (cryptanalytics). The practical part of the thesis uses historical ciphers as a motivation to learn programming. The author created a comprehensive set of programming tasks of varying difficulty, both assignment motivating students to solve interesting algorithmic problems and comprehensive practice of programming skills, and sample solutions, serving as a tool for encryption and decryption through the system and as a visualization. Assignment of tasks together with other information is processed in the form of a methodological manual for teachers. Then the tasks were presented to experienced IT teachers with various lengths of teaching experience and subjected to a questionnaire survey through a set of open questions with a wide answer.

Key words: ciphering, cryptology substitution, transposition, history, methodology, programming, algorithmization, visualization

Obsah

ÚVOD

Téma historické šifry jako motivace k výuce programování jsem si vybral z obdobného důvodu, z jakého jsem si vybral téma své bakalářské práce, která se věnovala historii šifrování jako takové. Kombinuje v sobě obory lidské činnosti, které jsou mi blízké – historii a informatiku, coby vědu o informacích. Mé zájmy se ostatně odráží v mé studijní aprobaci.

Uvážíme-li, že šifrování je nedílnou součástí dějin informačních technologií, můžeme konstatovat, že dějiny IT se začaly psát již v dávném starověku. Potřeba utajení informací před nepovolanými osobami je člověku vlastní od doby, kdy se naučil využívat komunikační prostředky. Hledání ideálního způsobu, jak zprávu bezpečně utajit vedlo ke vzniku kryptologie – vědy o utajování informací. Ta prodělala několikrát za dobu své existence překotný vývoj a málokdo si dnes uvědomuje, že k dnešním takřka dokonalým šifrovým počítačovým systémům vedla cesta, jejíž počátky jsou spojeny s jednoduchou záměnou písmen abecedy mezi sebou, politickými intrikami i vojenskými střety a co víc, jmény jako Gaius Julius Caesar, Jan Hus či Leonardo Da Vinci. Historické šifry jsou zkrátka ze své podstaty opředeny rouškou tajemství, které přitahuje.

Dle mého názoru dokáže přilákat i žáky, a to jak ty, kteří mají blíže k dějepisu, coby humanitní disciplíně, tak ty, kteří mají bližší vztah k informatice, coby disciplíně přírodovědné disciplíně. A co víc, právě historické šifry tak mají potenciál stát se pojítkem mezi těmito dvěma obory, které setře mezioborové rozdíly a akcentuje jejich vzájemné vztahy. Ať tak či tak, domnívám se, že historické šifry mohou sloužit jako skvělá motivace k výuce programování. Po něm sice sílí společenská poptávka, nicméně se stále ve školství potkává s obavami a rozpaky u značné části žáků, kteří programování vidí jako příliš abstraktní. Věřím, že vhodně zvolená kombinace uživatelsky příjemného vývojového prostředí a vizuálního programovacího jazyka ve spojení s vybranými zajímavými historickými šiframi dokáže žáky k programování efektivně motivovat.

Cílem teoretické části diplomové práce je na základě literární rešerše představit různé typy klasických substitučních ručních šifer a ojediněle pak několika málo transpozičních

šifer v dějinném sledu od starověku po 20. století. Cílem praktické části práce je pak vytvořit sadu programátorských úloh, včetně vzorových řešení, které poslouží jako doplňující nástroj k výuce programování. Ty pak hodlám ověřit v praxi a podrobit dotazníkovému šetření u žáků i vyučujících.

Stěžejní literární oporou pro tvorbu teoretické části práce mi je kniha Pavla Vondrušky: Kryptologie, šifrování a tajná písma (2006) a stejně tak zahraniční publikace Kniha kódů a šifer od Simona Singha (2009). Nosným zdrojem informací a užitečných nástrojů pak autorské stránky Michala Musílka: www.musilek.eu. (2010) V neposlední řadě v mnohém vycházím ze své bakalářské práce. (Musílek, 2017) Zbývající autory uvádím v závěrečném seznamu literatury.

1 ZÁKLADNÍ POJMY A DĚLENÍ ŠIFER

1.1 Základní kryptologické pojmy

Než přejdeme k samotným šifrám, považuji za nutné čtenáře nejprve seznámit se základními kryptologickými pojmy, které se budou objevovat v následujících kapitolách. Pavlu Vondruškovi, odborníkovi v oblasti kryptologie, se to v jeho knize (Vondruška, 2006) podařilo naprosto pregnantně. V následujícím textu se dotknu pouze těch pojmů, které úzce souvisí s tématem mé práce.

Vůbec nejzákladnějším pojmem je samotná **kryptologie.** Jedná se o vědu, která se zabývá utajením obsahu zpráv. Tu můžeme dělit dále na **kryptografii** – nauku o různých metodách šifrování**, kryptoanalýzu** – nauku o luštění šifrových textů a **steganografii,** která se zabývá utajením samotné existence zpráv**.** Mezi příklady steganografie lze zařadit např. skrytí zprávy v obrazu, či užívání neviditelného inkoustu vepsaného mezi nesouvisející text, který měl nepovolaného držitele zprávy zmást a odvést pozornost od pravého textu. Slavný arabský učenec Šiháb al-Qualqa-šandí se již kolem roku 1400 zmiňuje o několika typech tzv. neviditelných inkoustů založených na chemických reakcích. Přesto že jde o nesmírně zajímavou vědu, steganografie má, jak plyne z výše uvedeného, blíže spíše k jiným přírodovědným disciplínám, než k informatice, a proto jsem se ji dotkl pouze v této kapitole. Naproti tomu zbývající výše uvedené pojmy se v práci objeví hned několikrát.

Pojem **otevřený text** označuje původní, dosud nezašifrovaný text zprávy. V tomto ohledu je jeho protikladem tzv. **šifrový text.** K tomu, abychom dosáhli vytvoření šifrového textu, je zapotřebí samotné **šifrování –** postup, který vede k převedení otevřeného textu do textu šifrového. Opačný postup, kterým z šifrového textu získáme text otevřený, nazýváme **dešifrování**. Nikoliv však luštění. Ačkoli cílem obou procesů je získat odpovídající otevřený text z šifrového textu, neznamenají tato dvě slova totéž. Zatímco ten, kdo dešifruje, zná dopředu všechny postupy a informace k tomu, aby mohl převést šifrový text na otevřený, luštitel je nepovolaný příjemce, který se snaží šifru prolomit. Jiří Janeček vidí častý omyl v zaměňování těchto dvou pojmů v matoucím překladu z angličtiny. Anglické slovo *„decrypting“* se totiž se totiž svému českému protějšku *„dešifrování“* foneticky podobá daleko méně než jiné anglické slovo *„decyphering,“* které ovšem znamená luštění.

Další důležitý pojem je tzv. **šifrová abeceda.** Předtím, než se pustíme do šifrování textu, je důležité efektivně provést zápis otevřeného textu. Pro takový zápis se v kryptografii zpravidla používá **mezinárodní abeceda** tvořená 26 písmeny latinky bez diakritiky. Co to v praxi znamená, jsem uvedl ve své bakalářské práci (Musílek, 2017): *„Z českého textu zpravidla před šifrováním odstraníme diakritiku (háčky a čárky nad písmeny) i interpunkci (čárky, tečky, středníky, vykřičníky, otazníky, dvojtečky a uvozovky). **Šifrová abeceda** může používat běžná písmena (latinku) nebo také docela jiné znaky (např. řeckou abecedu, či různé grafické značky, jako jsou křížky, šipky, hvězdičky apod.). Velmi často se ale používá i k zápisu šifrového textu mezinárodní abeceda, takže abeceda otevřeného textu a šifrová abeceda mohou být stejné.“*

1.2 Substituční a transpoziční šifry

Šifrové systémy můžeme rozdělit do dvou základních skupin – na šifry substituční a transpoziční (Vondruška, 2006). Princip **substituce** spočívá v záměně znaků otevřeného textu za znaky textu šifrového. Tento zdánlivě triviální princip nabízí řadu modifikací a vylepšení, nicméně i v původní podobě je poměrně efektivní, uvažujeme-li, že se někdo setká s daným šifrovým textem poprvé. Jedním z typických a zároveň nejstarších systémů tohoto typu je tzv. Caesarova šifra.

Transpozice mění pořadí písmen otevřeného textu. Šifrový text by měl připomínat spleť náhodně sestavených znaků, ale stejně, jako substituce, má i tento systém pevně stanovená pravidla, která umožňují text efektivně zašifrovat, stejně jako dešifrovat. Opravdu triviálním příkladem transpozice je psaní pozpátku, ačkoliv čtenáře může překvapit, že to bylo vlastní i tak geniální osobnosti, jakou byl Leonardo da Vinci.

2 Historické substituční šifry

2.1 Počátky šifrování – monoalfabetické substituční šifry

„V jednoduchosti je síla.“ Tento citát v kontextu šifrových systémů rozhodně neplatí. Šifry, při jejichž aplikaci dochází k jednoduché záměně, tedy nahrazení znaku otevřeného textu za daný znak textu šifrového, (přičemž se šifrová abeceda obvykle skládá ze znaků abecedy textu otevřeného) označujeme jako monoalfabetické substituční šifry. Takové šifry lze poměrně snadno prolomit díky frekvenční analýze textu. Dostatečně dlouhý text, zašifrovaný na základě jednoduché monoalfabetické substituce, totiž bude obsahovat některé znaky mnohem častěji než jiné. V českém textu půjde např. o písmena E (10,5 %), A (8,8 %), O (8,3 %), I (7,7 %), N (6,7 %), viz statistické údaje Centra zpracování přirozeného jazyka. (kolektiv, 2008) Pro srovnání v anglickém textu pak o písmena E (11,2 %), A (8,5 %), R (7,6 %), I (7,5 %), O (7,2 %). Komparací s více světovými jazyky zjistíme, že nejfrekventovanějšími písmeny daného jazyka jsou především samohlásky. Nejčastěji zastoupené znaky Na základě všech těchto údajů jsme schopni porovnat nejčastěji zastoupené znaky šifrového textu se znaky statisticky nejčastěji se vyskytujícími a následnou metodou pokus – omyl zprávu vyluštit. Vše poté je již jen otázkou vynaloženého času a trpělivosti.

Na obranu monoalfabetických substitučních šifer je ovšem třeba říct, že jsou vůbec nejstarší lidskou činností v oblasti šifrování (pomineme-li steganografii). Podle Pavla Vondrušky (2006) první šifra pochází již ze starověku, z doby zhruba kolem roku 1500 př. n. l. z oblasti Mezopotámie. O té však nevíme mnoho, jen to, že šifra pracovala se záměnou klínových písmen za jiná klínová písmena, která měla stejnou zvukovou podobnost.

2.1.1 Šifra Atbaš

O téměř 1000 let později vznikla další šifra monoalfabetického typu – ta však byla již o něco sofistikovanější. Vyskytuje se v Bibli a to dvakrát, konkrétně pak v knize Jeremiáš, kapitole 25, verš 26, který zní: *„Všechny krále severu, blízké i vzdálené, jednoho po druhém – všechna království země, co jich je na světe. A po nich se napije král Šešak!“* a kapitole 51, verš 41: *„Ach, jak byl ten Šešak lapen, jakou chloubu světa dobyli? Jak mohl být tak zpustošen Babylon mezi národy!“* Ono opakující se slovo Šešak je šifrovým

textem, jehož otevřená podoba je Babel (hebrejské označení města Babylon) Jak jsem již uvedl ve své bakalářské práci (Musílek, 2017): *„Tato záměna vznikla už v hebrejském textu, okolo roku 500 př. n. l., kdy pisatel zaměnil druhé písmeno hebrejské abecedy bét, předposledním šin a dvanácté lamed jedenáctým kaf. Vezmeme-li to popořadě, zamění se první písmeno abecedy alef posledním tav, druhé bet předposledním šin, tedy A-T, B-Š a odtud název šifry atbaš.“* Nejlépe tyto změny znázorníme vytvořením převodové tabulky, jejíž obsah však nebude tvořen původní hebrejskou abecedou, nýbrž současnou mezinárodní abecedou o 26 znacích:

Tabulka č. 1 Převodová tabulka k šifře Atbaš

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A

Tuto tabulku lze použít pro šifrování i dešifrování zprávy, přičemž uvažujme, že horní řádek tvoří znaky otevřeného textu a dolní řádek znaky šifrového textu. V obou případech vyhledáme dané písmeno a nahradíme ho písmenem ve stejném sloupci ale jiném, příslušném řádku. Princip této šifry demonstrujeme na příkladu: „Mluv se mnou pomalu a já rychle porozumím.“

```
OT: MLUV SE MNOU POMALU A JA RYCHLE POROZUMIM
ŠT: NOFE HV NMLF KLNZOF Z QZ IBXSOV KLILAFNRN
```

Zvýše uvedeného je čtenáři jistě patrné, že nejde ani tak o šifru v pravém slova smyslu, jako spíše o stopu, kterou se po sobě rozhodl spisovatel zanechat. Šifra fungující na podobném principu, vzniknuvší jen o pár století později, se však vepsala do dějin již jako plnohodnotná součást vojenské strategie.

2.1.2 Slavná Césarova šifra

Gaius Julius Caesar se proslavil jako významný politik a vojevůdce antického Říma. Jeho dílo s názvem *„Zápisky o válce galské“* seznamuje čtenáře nejen s průběhem tažení, vojenskou strategií či barbarskými zvyky, ale také s revolučním prvek vojenské historie - šifrováním. Jedná se tak zároveň o první historický pramen, dokládající použití substituční šifry pro vojenské účely. Podle Simona Singha (2009) přitom používal slavný římský vojevůdce hned několik různých šifer. Dokonce údajně vznikl jejich soudobý

přehled, ovšem ten se nám bohužel nedochoval. Dochoval se nám však detailní popis jedné z těchto šifer, pro níž se, pro svou ojedinělost, přirozeně, vžil název Caesarova šifra.

V zásadě jde o obdobnou šifru, jakou je Atbaš, tedy jednoduchou / monoalfabetickou substituci. Převodová tabulka však bude na první pohled vypadat trochu jinak:

Tabulka č. 2 Převodová tabulka k šifře Caesar

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Princip této šifry spočívá v posunu znaků o tři pozice během procesu šifrování neboli nahrazení každého znaku znakem, který stojí v abecedě o tři pozice dále. Postupujeme přitom stejně, jako u předchozí šifry – chceme-li zprávu šifrovat, písmeno z horního řádku nahradíme písmenem z dolního řádku, chceme-li zprávu dešifrovat, zvolíme postup opačný. Ukážu to na příkladu slavného výroku: „Přišel jsem, viděl jsem, zvítězil jsem!“

```
OT: PRISEL JSEM VIDEL JSEM ZVITEZIL JSEM
ŠT: SULVHO MVHP YLGHO MVHP CYLWHCLO MVHP
```

Čtenáři je na první pohled jistě nápadná podoba opakujícího se slova „jsem“, které se logicky opakuje i v jeho šifrové podobě. Pro znesnadnění luštění se šifrové texty často rozdělovaly do skupin o určitém počtu, například pěti znaků:

```
ŠT: SULVH OMVHP YLGHO MVHPC YLWHC LOMVH P
```

Přestože text působí na první pohled nečitelně, jde stále o poměrně slabý šifrový systém. Ve své bakalářské práci (Musílek, 2017) jsem se zamyslel nad obměnami této šifry, které jsou ovšem stále limitované: *„I když ho můžeme obměňovat tím způsobem, že dolní řádek posuneme o jiný počet písmen, zjistíme, že možných posunů, tedy v tomto případě klíčů, je jen 25. Posun o 1, 2, 3, 4, …, 24, 25 znaků. Při posunutí o 26 znaků se budou písmena v horním a dolním řádku stejná, jako kdybychom je vůbec neposunuli, a pak se začnou opakovat už dříve použitá posunutí.“* I zde tedy platí hypotéza, že prolomení jednoduché substituce by pro zkušeného luštitele byla jen otázka času.

2.2 Digrafické substituční šifry

Již v době starověku se rovněž objevují tzv. digrafické substituční šifry. Ty se v zásadě příliš neliší od jednoduché záměny, snad jen v tom, že daný znak abecedy otevřeného textu je vždy nahrazen, namísto jedním znakem, dvojicí znaků šifrové abecedy, nebo také číslic.

2.2.1 Polybiův čtverec

Typickým zástupcem tohoto systému je tzv. Polybiův čtverec, jehož autorem nebyl nikdo jiný než slavný řecký historik Polybios. Onen čtverec se sestává celkem z 25 polí, jejichž obsah tvoří 25 znaků abecedy (obvykle se v takové situaci spojí dva znaky do jednoho pole, případně se jeden znak vynechá, v tomto případě zanedbáme písmeno Q, které se v českém textu běžně prakticky nevyskytuje) a jeho záhlaví je označeno číslicemi od 1 do 5 pro řádky i sloupce. Každý znak, obsažený v jednom z 25 polí čtverce, bylo možno kódovat dvěma čísly – právě kombinací čísel pro řádek a sloupec.

Tabulka č. 3 Polybiův čtverec

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I	J
3	K	L	M	N	O
4	P	R	S	T	U
5	V	W	X	Y	Z

Zajímavostí je, že onen systém byl původně zamýšlen pro pochodňovou signalizaci. Příklad takové signalizace popisuje ve své knize Pavel Vondruška: *„Např. písmeno v prvém řádku a pátém sloupci by bylo odesláno pomocí jedné pochodně v levé ruce a pěti pochodní v pravé ruce.“* Doplňuje zároveň, že takové signály mohly být vysílané poměrně bezpečně rychle a také na velké vzdálenosti.

Jistě si říkáte, že prohlédnout takový systém bylo naprosto primitivní. Ve skutečnosti jsme však dosud nehovořili o finální podobě šifry. O šifře hovoříme teprve ve chvíli, kdy písmena v tabulce nejsou seřazena v obvyklém sledu, nýbrž na základě stanoveného **hesla** – tzn. nejčastěji dohodnutého slova o kterém ví pouze odesílatel a příjemce. Takové heslo se vypíše do polí čtverce jako první s tím, že žádný znak hesla se neopakuje. Zbývající pole poté vyplní znaky abecedy v tradičním pořadí.

Princip této šifry si ukážeme na konkrétním příkladu. Zašifruji větu: *„Alexandr zaútočí zítra!“* za užití hesla: BUKEFALOS

Tabulka č. 4 Polybiův čtverec modifikovaný dle hesla: BUKEFALOS

	1	2	3	4	5
1	B	U	K	E	F
2	A	L	O	S	C
3	D	G	H	I	J
4	M	N	P	R	T
5	V	W	X	Y	Z

```
OT: ALEXANDR ZAUTOCI ZITRA
ŠT: 2122145321423144 55211245232534 5534454421
```

Přesto, že digrafické substituční šifry, zastoupené Polybiovým čtvercem, se mohou na první pohled zdát sofistikovanější, jsou odborníky vnímány pouze jako jiná forma jednoduché záměny. Je však třeba konstatovat, že právě Polybiův čtverec, který umožňoval převod písmen na číslice, se stal základem mnoha dalších šifrových systémů, a to i těch nejnovějších, jako např. šifry BIFID z konce 19. století, o níž se rovněž zmiňuji ve své práci. Ale nepředbíhejme – zatím se posuneme k šifrám, které mají svůj původ ve středověku.

2.3 Homofonní substituce a nomenklátory

Středověk pevně svázaný universalismem jistě prodělal v řadě oborů lidské činnosti ve srovnání s antickým odkazem jistý úpadek. V oboru kryptologie však nikoliv. Panovníci si stále více uvědomovali důležitost efektivního utajení zpráv před svými nepřáteli, a proto rozvoj šifrových systémů podporovali. Za zmínku jistě stojí překotný pokrok v steganografii – neviditelný inkoust. Arabský učenec Šiháb al-Qalqa-šandí na přelomu 14. a 15. století popsal ve svém díle hned několik druhů neviditelných inkoustů založených na chemické bázi. (Vondruška, 2006)

Ani kryptografové věnující se zdokonalení substituce na sebe nenechali dlouho čekat. Jak jsem již ve své práci opakovaně zmínil, monoalfabetické substituční šifry skýtají jednu zásadní nevýhodu. Budeme-li trpěliví, dříve nebo později systém prolomíme, a to i kdyby autor dané šifry vytvořil sebe více komplikovaný systém zpřeházení jednotlivých znaků. K tomu nám dopomůže frekvenční analýza textu, kterou jsem rozvedl v kapitole 2.1. Ta byla poprvé popsána právě ve středověku, (v 9. století) arabským učencem jménem Abú Júsuf Jaqúb ibn Isháq ibn as-Sabbáh ibn Omrán ibn Ismail al-Kindí. (Singh, 2009). Na

základě tohoto uvědomění se středověcí politici a vojevůdci zabývali otázkou, která substituční metoda by mohla předčit monoalfabetickou substituci z hlediska bezpečnosti.

Jednou z možných, dokonalejších cest, odolných proti metodě frekvenční analýzy, bylo nahrazovat písmena několika různými znaky či symboly najednou. Znaky, které měly v šifrovém textu zastupovat jeden znak otevřeného textu se nazývají **homofony**, proto popsané metodě říkáme **homofonní substituce**. Princip tohoto systému pregnantně popsal Pavel Vondruška (2007): *„Tyto šifry, na rozdíl od jednoduché substituce, kde každý otevřený znak se převádí pomocí jednoho znaku šifrové abecedy, používají pro převod některých vybraných znaků (zpravidla těch nejfrekventovanějších) více šifrových znaků. Tím se výrazně ztížilo luštění, které je založeno právě na frekvenční analýze šifrového textu.“* Do textu byly navíc vkládány často tzv. klamače – znaky, které byly ojedinělé, nic neznamenaly a měly za cíl oklamat luštitele. Jedním z prvních uživatelů homofonní šifry byl údajně vévoda Simeone de Crema z italské Mantovy na počátku 15. století. Luštění podobných šifer bylo přeci jen mnohem náročnější, a tak vznikaly při dvorech evropských panovníků specializované služby věnující se luštění šifer. Kupříkladu slavný matematik François Viète, podle nějž nesou název tzv. Vietovy rovnice, rozluštil na francouzském dvoře hned několik homofonních šifer.

První řízená šifrovací a luštitelská služba vzniká v roce 1500 v Benátské republice. Hlavou instituce se stal Giovani Soro, jeden z největších luštitelů západní civilizace a také autor nomenklátoru, který použil slavný conquistador Hernando Cortéz. Není bez zajímavosti, že zaměstnanci podobných institucí patřili mezi vážené občany, což se odráželo nejen ve společenské prestiži, ale i výši platu. Za vyzrazení důležitých státních informací vyplívajících z tajné korespondence jim ovšem hrozila smrt. (Vondruška, 2006)

Přesto, že šifrování bylo pro svou roli v důležitých politických či vojenských operacích od jakživa tak trochu tajemnou a tabuizovanou záležitostí, pořádaly se později první soutěže, jejichž cílem bylo mimo jiné finančně motivovat inovátory v tomto oboru. Vítěz mohl získat až 100 dukátů, což byl přibližně roční plat tamního dvořana. Takovým vítězem se stal např. Marco Rafael v roce 1525, který na soutěži představil novinku z oblasti steganografie – nový způsob neviditelného psaní. (Vondruška, 2006)

Ale zpět k homofonní substituci. Mohlo by se zdát, že téměř geniální a bez hlubší odbornosti neprolomitelný systém byl na světě. V čem byla potíž? Tyto šifry musely mít

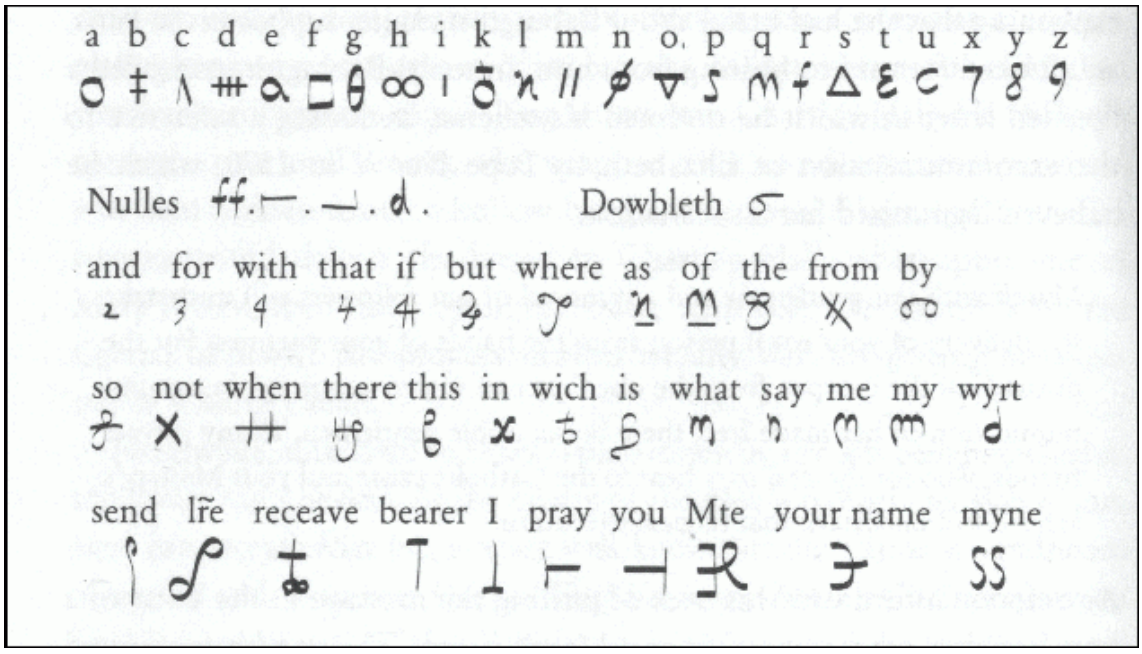
víc znaků, než má abeceda. Zdokonalení dosáhla nahrazením celých slov (pochopitelně jen těch nejfrekventovanějších) jedním znakem. Kombinací výše uvedených prvků, tedy homofonní šifry, klamačů a znaků či symbolů pro celá slova vznikly tzv. **nomenklátory**. Později, přidáváním dalších a dalších znaků, či číselných kódů pro celá slova (případně i sousloví, či krátké věty) se nomenklátory přetvořily na tzv. **kódové knihy** První nomenklátory pravděpodobně pochází rovněž z Itálie, konkrétně z doby kolem roku 1379 z pera Italského kryptografa Gabrieli di Lavinde. (Vondruška, 2006). Mnohem známější nomenklátor však pochází z pozdější doby, konkrétně z 16. století.

2.3.1 Nomenklátor Marie Stuartovny

Psal se rok 1586 a skotská královna Marie Stuartovna byla souzena pro velezradu. Údajného spiknutí proti její sestřenici, královně Alžbětě se účastnila i řada anglických katolických šlechticů, Mariiných podporovatelů, v čele s Anthony Babingtonem. Ti všichni byli popraveni až na samotnou Stuartovnu. K jejímu odsouzení totiž chyběl pádný důkaz. Alžbětin tajemník a šéf anglické špionáže v jedné osobě, sir Francis Walsingham, se zavázal k prokázání viny skotské vzdoro-královny.

Podle Simona Singha (2009) byli podporovatelé Stuartovny odhodlaní udělat vše proto, aby svou chráněnkyni vysvobodili a rozpoutali rekatolizaci v zemi. K tomu účelu však nutně potřebovali podporu ostatní katolických evropských velmocí, které by byly ochotné na Anglii zaútočit. K podniknutí těchto kroků potřebovali poslední, avšak nejdůležitější věc, Mariin písemný souhlas. Zprávu od Babingtona, která měla panovnici seznámit s jeho plány, se nabídl doručit Gilbert Gifford – katolík, který ji doručoval veškerou korespondenci. Gilbert ovšem pracoval jako dvojitý agent. Důležité informace, týkající se Marie Stuartovny a plánů jejích přisluhovačů vyzrazoval výše zmiňovanému siru Walsinghamovi. Stěžejní důkaz, tedy zpráva od rebela Babingtona určená Stuartovně, byla díky Gilbertovi ve Walsinghamových rukou. Ta však byla zašifrována, a to pomocí nomenklátoru, který spiklenci vytvořili.

Tzv. Nomenklátor Marie Stuartovny obsahuje 23 symbolů pro 23 písmen abecedy. Vynechána jsou přitom písmena j, v a w. Dále obsahuje 35 symbolů pro přesně stanovená slova a fráze, čtyři klamače (nuly) a speciální symbol, který znamenal, že následující znak je zdvojený.



Obr. 1 Nomenklátor Marie Stuartovny – převzato z www.simonsingh.net

Takový, byť poměrně sofistikovaný systém, nebyl pro královskou špionáž překážkou a vrchní tajemník pro šifry, Thomas Phelippes, brzy předložil dešifrovaný text. Tehdy bylo možné usvědčit jen některé rebely, nikoliv však všechny a konečně samotnou Stuartovnu. Proto se sir Walsingham rozhodl vyčkat, věděl, že Marie dřív nebo později na dopis odpoví. A tak se i stalo. Marie s plány svého ochránce písemně souhlasila a nepřímo si tak podepsala rozsudek smrti. Rozsudek podepsala nepřímo i všem předním rebelům, a to díky rafinovanému Walsinhamovu nápadu. Ten pověřil kryptoanalytika Thomase Phelippese, aby do odpovědného dopisu pro Babingtona připsal Mariino přání seznat jména všech svých podporovatelů. Pochopitelně bez jejího vědomí. Nic netušící Babington v dalším šifrovaném dopisu královně ochotně představil jména všech oddaných pánů. Walsingham měl všechny důkazy a dostál tak svému slibu. Dopadení spiklenci byli krutě trestáni a povětšinou nakonec popraveni, a to včetně samotného Babingtona. Soud obvinil Marii Stuartovnu z velezrady a skotská královna byla 8. února roku 1587 ve věku 44 let popravena – kat ji hlavu setnul údajně až na třetí pokus.

Důvěrná korespondence mezi Marií Stuartovnou a sirem Babingtonem vlastně důvěrná nikdy nebyla. Kryptoanalýza v té době byla na vzestupu, frekvenční analýza textu znamenala zásadní zlom v luštění a Anglie nebyla zdaleka jedinou zemí, kde byla praxe dvorních kryptoanalytiků. Jak jsem uvedl ve své bakalářské práci, pokud by jejich korespondence probíhala bez šifrování, dost možná by byli více diskrétní a získat důkazy

pro jejich usvědčení by bylo obtížnější. Tuto myšlenku kvituje Simon Singh (2009) svým komentářem: *„Šifra Marie Stuartovny jasně ukazuje, že špatné šifrování je horší než žádné.“*

Jak je tedy možné, že se systém nomenklátorů udržel dalších cca 300 let? Pavel Vondruška (2006) odpověď na tuto otázku zdůvodňuje tím, že tato metoda má jednu klíčovou výhodu a tou je její jednoduchost a rychlost aplikace. Teoreticky je každý schopen vytvořit vlastní kódovou knihu a tu pro šifrovanou komunikaci použít. Kritérium bezpečnosti ovšem za těmito přednostmi systému pokulhává. A právě potřeba zvýšení bezpečnosti dala vzniknout novému převratnému systému.

2.4 Polyalfabetická substituce s periodickým heslem

Přesto, že nomenklátory se aktivně využívaly téměř do konce 19. století, ukázaly se hned z kraje svého fungování jako ne zcela spolehlivé z hlediska bezpečnosti. To byl přitom pro vojenskou, diplomatickou či obchodní komunikaci faktor naprosto zásadní. Proto kryptologové nadále hledali nové cesty k pomyslnému ideálnímu šifrovacímu systému, který bude co nejlépe odolávat luštění.

Myšlenka takového sytému se zrodila v hlavě Leona Battisty Albertiho, pravděpodobně roku 1476. Tento renesanční člověk, který vynikal v mnoha různých oborech v souladu s pravým významem těchto slov, tehdy popsal originální šifrovací systém, který svou existencí znamenal překotný krok kupředu v oblasti šifrování. Snad proto vešel Alberti do historie ve známost jako „otec západní kryptologie.“ V čem spočívala genialita nápadu? Princip jeho systému spočíval v pravidelném střídání dvou či více monoalfabetických substitucí a jednalo se tak o tzv. **polyalfabetickou substituci**.

Měl celou řadu pokračovatelů, kteří na jeho dílo navázali – Johannes Tritheimus, Giambattista della Porta a konečně Blaise de Vigenère, na jehož práci – tzv. Vigenèrově šifře – si polyalfabetickou substituci představíme.

2.4.1 Vigenèrova šifra

K šifrování a dešifrování nám nebude stačit jednoduchá, dvouřádková převodová tabulka jako u předchozích systémů, ale použijeme velkou čtvercovou tabulku, kterou publikoval již zmíněný Johannes Tritheimus pod názvem ***tabula recta***, mohli bychom ji ovšem také najít pod názvem **Vigenèrův čtverec**:

	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Šifrování se řídí tzv. ***periodickým heslem***. V praxi to znamená, že postupně střídáme jednotlivá písmena dohodnutého klíče. Tučně vyznačené znaky v řádcích tabulky představují znaky klíče, tučně vyznačené znaky v sloupcích tabulky znaky otevřeného textu. V průsečíků řádku a sloupce, tedy kombinaci znaku klíče se znakem otevřeného textu najdeme znak, který pak zapíšeme do šifrového textu. Při dešifrování textu postupujeme tak, že v řádku určeném příslušným znakem klíče najdeme dané písmeno šifrového textu a od něj jdeme svisle vzhůru do prvního řádku tabulky, kde najdeme příslušný znak otevřeného textu.

Princip si opět demonstrujeme na jednoduchém příkladu. Jak jsem již uvedl výše, bývá zvykem rozdělit text po skupinách, proto jej opět rozčlením do pětic znaků. Pomocí periodického klíče: *„VIGENERE“* zašifruji citát: *„Špatný žák, který nepřevyšuje svého mistra.“*

```
KLÍČ:   VIGEN EREVI GENER EVIGE NEREV IGENE REVIG E
OT:     SPATN YZAKK TERYN EPREV ISUJE SVÉHO MISTR A
ŠT:     NXGXA CQEFS ZIECE IKZKZ VWLNZ ABIUS DMNBX E
```

Polyalfabetická substituce s periodickým heslem, využívající tabulku typu tabula recta se v budoucnu dočkala dvou dalších obměn. Poslední známá pochází z doby kolem roku 1850, jejím autorem je anglický admirál sir Francis Beaufort. Z poslední uvedené datace je patrné, že polyalfabetická substituce se dočkala také dlouhého trvání a co víc, až do druhé poloviny 19. století byly šifry tohoto typu považovány za zcela nerozluštitelné.

2.4.2 Luštění polyalfabetické substituce s periodickým heslem

Do roku 1863 byla nejslavnější polyalfabetická Vigenèrova šifra známá jako „*Le chifre indéchiffrable“*, v překladu *„nerozluštitelná šifra.“* Do doby, kdy důstojník pruské armády ve výslužbě Friedrich Wilhelm Kasiski publikoval ve své knize *Die Geheimschriftenunddie Dechiffrirkunst* (Tajné šifry a umění je dešifrovat) postup, jakým lze polyalfabetickou substituci s periodickým heslem dešifrovat. Hovoříme o způsobu známém jako Kasiského test. Již ve své bakalářské práci (Musílek, 2017) jsem zmínil kontroverzní tvrzení Simona Singha (2009), podle kterého stejný postup objevil anglický matematik Charles Babbage o téměř deset let dříve. Pravděpodobně jej nepublikoval proto, že Velká Británie byla toho času ve válečném stavu. Zveřejněním tohoto objevu by se Britové vědomě připravili o možnost luštit relativně snadno vojenské šifry, které Rusové používali v Krymské válce.

Nás ani tak nebude zajímat prvenství v této oblasti, jaké spíše jeho podstata. V každém jazyce je nejen určitá četnost písmen, ale také četnost tzv. bigramů, tedy dvojic po sobě jdoucích písmen. Stává se, že bigram připadne na stejnou dvojici písmen stanoveného klíče neboli periodického hesla. Princip Kasiského testu spočívá v hledání dvojic shodných bigramů v šifrovém textu, určení vzdálenosti mezi znaky této dvojice a v hledání společného dělitele těchto vzdáleností, který odpovídá délce použitého hesla. V momentě, kdy známe délku hesla, můžeme písmena šifrového textu rozpočítat do skupin připadajících na stejné písmeno hesla a na tyto skupina následně aplikovat frekvenční analýzu textu.

Toto zjištění radikálně změnilo dosud obecně přijímaný názor na Vigenèrovu šifru, který říkal, že šifra je nerozluštitelná. Opět se tak otevřelo téma hledání nových a efektivnějších systémů.

V následujících podkapitolách se seznámíme s ještě komplikovanějšími substitučními šiframi. Postupy jejich luštění jsou složitější, než frekvenční analýza nebo Kasiského test. Proto už budu dále uvádět pouze postup šifrování a dešifrování, nikoliv princip luštění.

2.5 Bigramové a polygramové substituční šifry

Všechny dosavadní šifrovací substituční systémy ukázaly své přednosti i nedostatky. Jako nejúspěšnější se z dlouhodobého hlediska ukázala polyalfabetická substituce zastoupena zejména Vigenèrovou šifrou. Kasiského test ovšem tento systém prolomil, a tak ideál „neprolomitelné šifry“ pozbyl na platnosti.

Bylo třeba hledat způsob, jak zvýšit odolnost substitučních šifer proti luštění pomocí četnosti znaků. Myšlenka nahrazení každého znaku více homofony, se ukázala jako nepraktická a nedostačující. Objevila se tak myšlenka záměny celých skupin otevřeného textu za skupiny šifrového textu. Pakliže dojde k záměně bigramů otevřeného textu za bigramy šifrového textu, hovoříme o tzv. bigramové substituční šifře. Obecně můžeme o šifrách fungujících na této bázi hovořit jako o polygramových substitučních šifrách, kdy lze tímto způsobem tvořit záměny o libovolném počtu znaků.

2.5.1 Šifra Playfair

Typickým zástupcem bigramové substituční šifry je šifra Playfair, která byla navržena v roce 1854 anglickým přírodovědcem a kryptologem Charlesem Wheastonem. Své jméno nese po jeho příteli Lyonu Playfairovi, významném britském poslanci a nadšenému propagátorovi této šifry. Šifru používala zejména britská armáda.

K šifrování využívá čtvercovou tabulku o 5 x 5 polích. Do té vyplníme nejprve písmena podle zvoleného hesla, a to postupně po řádcích. Přitom je třeba myslet na to, že žádný znak se neopakuje, proto, pakliže budeme mít heslo, ve kterém se některé znaky objevují vícekrát, opakující se znak přeskočíme. Zbývající prázdná pole pak vyplníme zbývajícími písmeny abecedy v obvyklém pořadí. Zde je třeba připomenou fakt, že šifrová abeceda má 26 písmen, a tudíž je potřeba jedno z písmen vynechat, nebo sloučit dvě písmena do jednoho pole. Pro následnou demonstraci vynechávám písmeno Q, které se dle statistik v českém textu nejnižší četnost. Pokud by se v otevřeném textu přeci jen objevilo, vyřešíme tento znak spojením písmen K a V.

Princip opět demonstruji na jednoduchém příkladu. Pro lepší přehlednost budeme postupovat po jednotlivých krocích. Nejprve zvolíme heslo. Pokud zvolíme jako heslo mé jméno: *„Pavel Musílek“*, bude mít tabulka tuto podobu:

Tabulka č. 5 Šifrovací tabulka Playfair modifikovaná dle hesla: Pavel Musílek

P	A	V	E	L
M	U	S	I	K
B	C	D	F	G
H	J	N	O	R
T	W	X	Y	Z

Do prvních pěti polí doplníme bez potíží křestní jméno, kde se žádný znak neopakuje, tedy: P, A, V, E a L. Do druhého řádku doplníme příjmení s tím, že ty znaky, které jsou již obsaženy v prvním řádku, se nebudou opakovat, tedy: M, U, S, I a K. Heslo tedy zaplnilo první dva řádky, další řádky vyplní zbývající písmena mezinárodní abecedy v abecedním pořadí, přičemž písmeno Q vynecháme z výše uvedených důvodů.

S takto vyplněnou tabulkou můžeme přistoupit k samotnému šifrování. To probíhá po zmíněných bigramech, tedy nikoliv po jednotlivých znacích, ale jejich dvojicích. Předtím, než tedy přistoupíme k šifrování, je zapotřebí otevřený text rozdělit do dvojic. Oba znaky bigramu mohou, ale také nemusí být ve stejném řádku či sloupci. Zde se dostáváme k principu samotného šifrování pomocí tabulky, který je vázán souborem přesně stanovených pravidel:

- Leží-li oba znaky ve stejném řádku, pak je každý znak bigramu nahrazen znakem ležícím vpravo od něj. Pokud jde o znak uložený v posledním poli řádku, ten je nehrazen prvním znakem téhož řádku.
- Leží-li obě znaky ve stejném sloupci, pak je každý znak bigramu nahrazen znakem ležícím pod ním. Pokud jde o znak uložený v posledním poli sloupce, ten je nahrazen prvním znakem téhož sloupce.
- Leží-li dané znaky v jiném řádku a sloupci, pak je každý znak bigramu nahrazen znakem ležícím v průsečíku jeho vlastního řádku a sloupce obsahující druhý znak bigramu.

Tabulka č. 6 Šifrovací tabulka Playfair modifikovaná dle hesla: Pavel Musílek s graficky vyznačenými postupy pro šifrování

P	A	V	E	L		P	A	V	E	L		P	A	V	E	L
M	U	S	I	K		M	U	S	I	K		M	U	S	I	K
B	C	D	F	G		B	C	D	F	G		B	C	D	F	G
H	J	N	O	R		H	J	N	O	R		H	J	N	O	R
T	W	X	Y	Z		T	W	X	Y	Z		T	W	X	Y	Z

Ve výše uvedených tabulkách máme naznačeny dané posuvy v souladu s popsanými pravidly. Dešifrování vychází z týchž pravidel. Zvlášť si je popisovat nemusíme, stačí pouze uvést, že v prvním i druhém pravidle nahradíme slovo „vpravo“ slovem „vlevo“, slovo „první“ slovem „poslední“ a naopak. Třetí pravidlo zůstane beze změny.

Aby demonstrace byla kompletní, zašifrujeme následující zprávu: *„Baron Lyon Playfair se narodil v Indii.“*

OT: BA RO NL YO NP LA YF AI RS EN AR OD IL VI ND IX YI
ŠT: CP HR RV EY HV PV EO EU NK VO LJ NF KE ES XN SY EF

V případě této věty dochází ke kuriozní situaci, kterou ovšem řeší další z pravidel. Pakliže vznikne situace, kdy je dvojice tvořena stejným písmenem, vložíme mezi ně v rámci otevřeného textu písmeno, nebo písmena, (v závislosti na tom, zda má otevřený text lichý, či sudý počet znaků) která se vyskytují v textu jen vzácně, přičemž přihlížíme k četnosti znaků. (v případě českého jazyka tedy půjde např. o písmena X, nebo W) Stejným způsobem bychom doplnili písmeno na konec textu, pokud by počet znaků v textu byl lichý a poslední znak tak stál samostatně.

Upřeme-li svou pozornost na všechna opakující se písmena a jejich šifrovou podobu, uvědomíme si praktický význam bigramových, či polygramových šifer. Kupříkladu nejčastěji se opakující písmeno I má v šifrové textu hned několik podob (postupně): U, K, S, S, F. Jednotlivé znaky nejsou nahrazeny vždy pouze jedním konkrétním znakem, symbolem, či skupinou znaků, ale pokaždé jsou nahrazeny jiným znakem, podle toho, v jakém bigramu se vyskytují – právě v tom spočívá genialita šifry Playfair a všech dalších šifer, které pracují na bázi bigramového, či polygramového systému. Ty se dočkaly i jistých inovací a vylepšení, jako např. šifra s názvem BIFID.

2.5.2 Šifra BIFID

Kdyby pro nic jiného, tak už jen pro svůj původ je šifra BIFID naprosto unikátní šifrou. Její autorov Felix Marie Delastelle totiž nezastával žádnou vojenskou či politickou funkci, ani nebyl matematikem či lingvistou. Pracoval jako účetní skladů v námořním přístavu a kryptologie mu byla pouze koníčkem. Přesto, coby kryptograf-amatér stojí za jednou z nejproslulejších polygramových šifer.

To, co je na šifře BIFID opravdu pozoruhodné je to, že jako první substituční šifra v historii nešifruje přímo otevřený text do textu šifrového, ale používá tzv. **mezitext**.

Tabulka č. 7 Šifrovací tabulka k šifře BIFID vyplněná dle hesla: Kryptologie

	1	2	3	4	5
1	K	R	Y	P	T
2	O	L	G	I	E
3	A	B	C	D	F
4	H	J	M	N	S
5	U	V	W	X	Z

Tato šifrovací tabulka nám při prvním pohledu jistě připomene tabulku, kterou využívá šifra Playfair. Ta principiálně i funguje na stejné bázi. První dva řádky tvoří znaky určeného hesla: „Kryptologie.“ Rozdíl spočívá v očíslování řádků a sloupců, které pro nás bude stěžejní při tvorbě zmíněného mezitextu.

Poté, co vytvoříme tabulku, rozdělíme otevřený text po pěticích znaků. (na rozdíl od šifry Playfair zde otevřený text nedělíme po dvojicích, ale po pěticích znaků, proto také nehovoříme o bigramové, ale polygramové substituční šifře) Poté můžeme přistoupit k tvorbě zmíněného mezitextu. Pod každý znak dané pětice zapíšeme do prvního řádku číslo daného řádku, ve kterém se daný znak nachází v rámci šifrovací tabulky. Do druhého řádku zapíšeme číslo daného sloupce. Poté, co vytvoříme tento dvouřádkový mezitext, přichází konečně proces šifrování. Jak získáme šifrový text, jsem důkladně popsal již ve své bakalářské práci (Musílek, 2017): *„Z tohoto dvouřádkového mezitextu pak získáme šifrový text tak, že bereme první a druhou číslici v horním řádku, pak třetí a čtvrtou číslici v horním řádku, potom poslední číslici v horním řádku a první číslici v dolním řádku, následuje druhá a třetí číslice v dolním řádku a nakonec čtvrtá a pátá*

číslice v dolním řádku. V každé z těchto dvojic znamená první číslice číslo řádku v šifrovacím čtverci a druhá číslo sloupce. Nakonec tedy získáme pětici písmen, která tvoří příslušnou část šifrového textu. Pokud na konci textu vyjde kratší skupina než pětimístná, tak zde celý postup zkrátíme.“

Princip si ukážeme na konkrétním příkladu. S použitím výše zobrazeného tabulky zašifruji citát: *„Být či nebýt, to je oč tu běží!“*

```
OT:                  BÝTČI NEBÝT TOJEO ČTUBĚ ŽÍ
SOUŘADNICE ŘÁDKŮ:    31132 42311 12422 31532 52
SOUŘADNICE SLOUPCŮ:  23534 45235 51251 35125 54
ŠT:                  AYLFD JAPVF RJERU AWGUE VX
```

Pokud text zprávy dešifrujeme, vytváříme mezitext po řádcích a do otevřeného textu pak převádíme dvojice číslic stojících pod sebou v horním a dolním řádku mezitextu.

2.6 Od substitučních k transpozičním šifrám

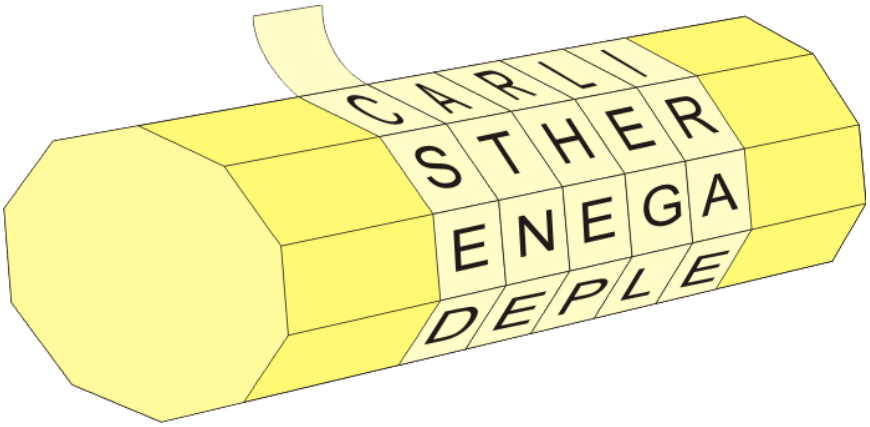
Tímto výčtem šifer a jejich popisem uzavíráme stěžejní část teoretické části práce, která si kladla za cíl čtenáře seznámit s průřezem substitučních šifrových systémů napříč historií – od starověku po moderní dějiny 20. století. Souběžně se substitučními šiframi se používaly a postupně zdokonalovaly také šifry transpoziční. Proto bude příští kapitola věnována alespoň malé „ochutnávce“ z dějin šifer transpozičních. Jelikož ty nejsou hlavním předmětem teoretické části této práce, omezuji se zde pouze na tří zajímavé šifry (zástupci pro starověk, raný novověk a konec 19. století), z nichž dvě jsou důležité pro praktickou část práce.

3 HISTORICKÉ TRANSPOZIČNÍ ŠIFRY

3.1 Skytalé

Ve starém Řecku byla nejčastěji uplatňovaným, postupů pro utajení zpráv steganografie. Jeden z nejpopulárnějších způsobů se nám dochoval v díle Dějiny od „Otce dějepisu“ – Herodota. Ten popisuje situaci, kdy byla poslu oholena hlava, na ní byl poté napsán vzkaz a jen co poslu narostly vlasy, mohl se vydat na cestu k příjemci. Ten přikázal posla oholit, zprávu si přečetl a bylo-li třeba, mohl svému spojenci odpovědět stejně. Takový způsob doručování zpráv byl sice poměrně bezpečný, ale nesmírně zdlouhavý. Co do rychlosti byla jistě spolehlivější šifra zvaná Skytalé.

Šifra Skytalé byla hojně používána starými Řeky, zejména Sparťany, zhruba od 7. století před naším letopočtem. Názvem Skytalé se nemyslí pouze samotný šifrový systém založený na principu transpozice, ale také šifrovací zařízení, které používá. To bylo tvořeno válcem, případně hranolem s podstavou ve tvaru pravidelného mnohoúhelníku (nejčastěji dřevěným) na nějž byl navinutý pruh pergamenu. (viz obrázek níže)



Obr. 2 Skytalé – převzato z WikimediaCommons (licence CreativCommons BY-SA 3.0)

Jak probíhalo samotné šifrování a doručení zprávy? Odesílatel napsal zprávu po řádcích rovnoběžných s osou válce (osou pravidelného mnohoúhelníku) na pergamen šroubovitě obtočený kolem dřevěné hole. Po odvinutí pergamenu měl text svou šifrovou podobu. Poté, co byla zpráva doručena, příjemce ji navinul na svou hůl a zprávu si mohl pohodlně přečíst. Zde je třeba zdůraznit, že nutnou podmínkou k úspěšnému dešifrování zprávy, je, aby odesílatel a příjemce měli hůl o stejných rozměrech. Pokud by se tedy zpráva dostala do nepovolaných rukou, bez znalosti přesných rozměrů hole nebylo možné zprávu rozluštit.

3.2 Richelieova transpozice

Armand Jean du Plessis, vévoda de Richelieu, též známý jako kardinál Richelieu. Postava nechvalně interpretovaná Alexandrem Dumasem v jeho asi nejznámějším díle s názvem Tři mušketýři. Co víme s jistotou je, že šlo o významného duchovního, státníka, prvního ministra francouzského krále Ludvíka XIII. a také autora pozoruhodné transpoziční šifry. Kardinála k šifrování přivedl Antoine Rossignol, jeden z nejuznávanějších luštitelů 17. století. Není bez zajímavosti, že Rossignol je tvůrcem tzv. Velké šifry, která byla podobně jako Vigenérova šifra považována několik století za nerozluštitelnou. V roce 1890 objevil vojenský historik Victor Gendron část soukromé korespondence Ludvíka XIV., která byla zašifrovaná touto šifrou. Étiennu Bezeriesovi se nakonec podařilo systém prolomit, po třech letech usilovné práce. Informace získané z rozluštěných dopisů byly pro příznivce záhad dechberoucí. O jedné z těchto informací jsem se zmínil již ve své bakalářské práci (Musílek, 2017): *„Díky tomu známe mimo jiné pravdu o muži se železnou maskou, jehož případ je popsán v jednom z dopisů. Řada odborníku se mylně domnívala, že se jednalo se o panovníkovo dvojče. Ve skutečnosti šlo o velitele Viviena de Bulonde, který ohrozil tažení francouzské armády do Itálie tím, že zbaběle uprchl z nechráněné pohraniční pevnosti Cuneo.“*

Nyní ale už k samotné Richelieově transpozici, chceme-li Richelieově jednoduché blokové transpozici. Postup je následující. Nejprve je zapotřebí otevřený text rozdělit do stejně dlouhých bloků neboli skupin znaků, přičemž délka bloku je dána počtem znaků zvoleného hesla. Přeskupení znaků v blocích je rovněž dáno počtem znaků hesla a je stejné, jako přeskupení znaků hesla do pořadí dle abecedy.

Tento postup si opět demonstrujeme na jednoduchém příkladu. Zašifrujeme větu: *„Jeden za všechny, všichni za jednoho.“* pomocí hesla: LUDVIK. Písmena hesla LUDVIK očíslujeme 123456. Tato písmena pak seřadíme dle abecedního pořadí, čímž získáme permutaci DIKLUV. Pořadí očíslovaní hesla se nám tedy změní na 356124. Tuto permutaci následně aplikujeme na otevřený text:

```
OT:       JEDENZ AVSECH NYAVSI CHNIZA JEDNOH O
HESLO:    356124 356124 356124 356124 356124
ŠT:       DNZJEE SCHAVE ASINYV NZACHI DOHJEN O
```

Richelieu poté celý text přepsal do podoby dlouhého sled malých písmen bez mezer, aby nijak nenaznačil délku použitého hesla, tedy: *dnzjeeschaveasinyvnzachidohjeno*

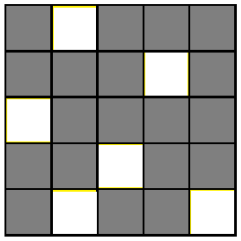
3.3 Fleissnerova mřížka

Tuto šifrovací pomůcku významnou měrou pomohl zpopularizovat Jules Verne ve svém slavném románu Mathias Sandorf z roku 1885. Svůj původ má ovšem z doby o čtyři roky dřívější, kdy ji publikoval Fleissner von Wostrowitz ve své knize věnované kryptografii. Její princip je docela prostý, za to poměrně efektivní. Nejlépe si jej vysvětlíme na konkrétním příkladu.

Nejprve je třeba vytvořit si samotnou mřížku. Tou je čtverec tvořený určitým počtem políček, z nichž některá políčka vystřihneme. Políčka k vystřižení nelze zvolit zcela náhodně, je třeba ohlídat, aby se při otáčení mřížkou o 90°, 180° a 270° otvory vytvořené vystiženými políčky nikdy nepřekrývaly. Tuto mřížku poté přiložíme na papír. Do prázdných políček postupně vypisujeme znaky otevřeného textu. V momentě, kdy jsme políčka zaplnili, pootočíme mřížku o 90° – tento postup 3x zopakujeme. Po odejmutí mřížky nám zůstane souvislý čtverec vyplněný znaky textu. (Klíma, 1994)

Princip šifry si demonstrujeme na jednoduchém příkladu. Zašifruji větu: *„Tisíc cest vede k jednomu cíli.“*

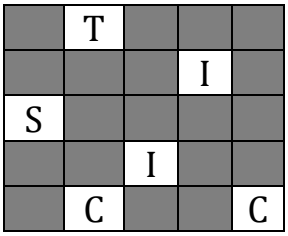
Tabulka č. 8 Tzv. Fleissnerova mřížka



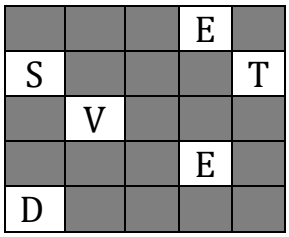
OT: TISÍC CEST VEDE K JEDNOMU CÍLI

Tabulka č. 9 Tzv. Fleissnerova mřížka v procesu šifrování

Krok 1:



Krok 2:



KROK 3:

E			K	
		J		
				E
	D			
			N	

KROK 4:

				O
	M			
			U	
C				I
		L		

Tabulka č. 10 Rozmístění znaků šifry po odstranění Fleissnerovy mřížky

E	T	E	K	O
S	M	J	I	T
S	V		U	E
C	D	I	E	I
D	C	L	N	C

Po odejmutí mřížky můžeme do zbývajícího volného pole uprostřed vyplnit poslední znak textu, pakliže má, jako v tomto případě, lichý počet znaků:

Tabulka č. 11 Finální podoba text po odstranění Fleissnerovy mřížky

E	T	E	K	O
S	M	J	I	T
S	V	I	U	E
C	D	I	E	I
D	C	L	N	C

Výsledný text tabulky poté přepíšeme po pěticích znaků.

ŠT: ETEKO SMJIT SVIUE CDIEI DCLNC

Fleissnerova mřížka je hned po Skytalé druhá nejznámější šifrovací pomůcka. Pro svou jednoduchost byla aktivně využívána německou armádou v době první světové války.

4 ÚVOD K PRAKTICKÉ ČÁSTI PRÁCE

Hlavním cílem praktické části práce je historické šifry využít jako motivaci k výuce programování. Důvodů, proč jsem si zvolil tento cíl, je hned několik. V první řadě sehrál důležitou roli můj zájem o historické šifry jako takové. Historii šifrování totiž považuji za nesmírně zajímavou a poutavou kapitolu dějin informačních technologií. Můj zájem do značné míry také reflektuje můj studijní obor – Učitelství informatiky a dějepisu. Sám jsem se nikdy neprofiloval pouze humanitním, či pouze přírodovědným směrem. Tím se dostávám k dalšímu z důvodů. Zařazení nauky o historických šifrách do výuky informatiky by mohlo, dle mého názoru, posloužit k akcentování mezioborových vztahů mezi dějepisem a informatikou. Domnívám se, že tato provázanost může pozitivním způsobem ovlivnit vztah žáka k dosud méně preferovanému předmětu, bez ohledu na to, zdali je spíše přírodovědného, či společenskovědního zaměření. Nejdůležitějším důvodem je snaha pomoci učitelům v motivaci žáků k výuce programování. Poptávka po zařazení algoritmizace a programování do výuky na školách intenzivně stoupá. Přesto lze ve společnosti vypozorovat jisté antipatie k studiu přírodovědných a technických oborů, programování nevyjímaje. I proto je stále obtížnější vést žáky ke kritickému a logickému myšlení.

Výše uvedené důvody mne vedly k vytvoření metodické příručky pro učitele, v rámci níž jsem navrhl úlohy různé obtížnosti a jejich vzorová řešení, ale nechybí ani stručný popis zvoleného programovacího jazyka a vývojového prostředí. Zadání úloh jsou postavena na realizaci a vizualizaci šifrovacích algoritmů substitučních a transpozičních historických šifer, které by měly být pro žáky nejen snadno pochopitelné, ale především zajímavé a tím i motivující. Díky výše zmíněné mezioborové provázanosti je zvýšena pravděpodobnost, že žáci humanitně zaměření získají díky vazbě na dějiny informatiky zájem o programování a nadšenci do programování v sobě objeví zájem o historii.

Nyní už k samotné metodické příručce. Metodické příručky jsou obecně jakési učebnice pro učitele, přesněji řečeno podpůrné materiály, které by učiteli měly pomoci při plnění vytyčených cílů. Metodika samotná může znamenat nauku o metodě v teoretické rovině, či metodu jako takovou. Podle Josefa Maňáka (2003) je metoda cestou k cíli, která je rozhodujícím prostředkem k dosahování cílů uvědomělé činnosti. V souladu s uvedenými skutečnostmi jsem se pokusil vytvořit vlastní metodickou příručku.

Příručku otevírá úvod, který čtenáři předkládá možný pohled na problematiku výuky programování na školách a autorovu vizi historických šifer jako motivace k výuce programování (viz výše). První část příručky dává učiteli návod, jak žáky motivovat. Oporou by mu v tom měly být uvedené historické skutečnosti a zajímavosti, které představují historické šifry jako něco tajemného a záhadného, co stojí za to zkoumat. Zbývající části poskytují učiteli návod, podle kterého může žáky seznámit s vývojovým prostředím, pro nějž jsou koncipovány úlohy, které žáci mají řešit. Jedná se o výběr základních a pro daný typ úloh potřebných funkcí zvoleného programovacího jazyka, nikoliv referenční příručku, která by postihovala všechny prvky zvoleného programovací jazyka nebo všechny funkcionality vývojového prostředí. Zde je třeba zdůraznit, že příručka vychází z předpokladu, že žáci již byli v minulosti seznámeni se základy algoritmizace a programování.

Pokud jde o vhodný programovací jazyk a vývojové prostředí, rozhodl jsem se po zvážení jednoznačně pro vizuální programovací jazyk Scratch. Vedlo mě k tomu několik okolností. Jak jsem ve své práci opakovaně uvedl, cílem je žáky k programování motivovat. Během mé pedagogické praxe na základní škole jsem se přesvědčil o tom, že vizuální jazyky jsou pro úvod do programování vhodnější než jazyky textové. Ty žáky obvykle odradí hned v prvopočátku – vnímají je jako příliš abstraktní a složité. Nutnost věnovat pozornost syntaxi, tj. přesnému psaní anglických slovíček a matematických značek, které tvoří příkazy jazyka, odvádí žáky od soustředění na podstatu problému a vhodnou algoritmizaci. Naopak vizuální jazyky a obzvlášť jazyk Scratch disponují příjemným grafickým prostředím, jsou dokonale intuitivní a především – není potřeba znát příkazy jazyka a jejich přesnou syntaxi. Tu v případě jazyka Scratch nahrazuje skládání bloků příkazů do sebe podobně, jako u dětské stavebnice. Podobný názor jako já má řada kolegů z řad učitelů informatiky, s nimiž jsem hovořil. Výhody jazyka Scratch pro úvod do algoritmizace a programování pregnantně shrnula do několika bodů Alena Halousková (2012) ve své diplomové práci. Jedná se o: zařazení jazyka mezi tzv. vizuální programovací jazyky, dostupnost v české verzi, dostatek programátorských konceptů k prvnímu seznámení s programováním, možnost nahrávat projekty na internetovou platformu, možnost tvořit multimediální programy a konečně, podle mého názoru, dva nejzásadnější argumenty, které mě nakonec utvrdily v přesvědčení, že Scratch je nejlepší volbou, statisticky podložená oblíbenost u dětí a dospívajících (většina z několika stovek tisíc tvůrců je zastoupena uživateli ve věku mezi 10 – 20 lety) a dostupnost (Scratch

funguje jako freewarová platforma zcela zdarma a jako on-line webová aplikace, není tudíž potřeba jej instalovat).

Součástí příručky jsou zadání programátorských úloh. Ta jsem adresoval přímo žákům. Vyučující tak má možnost využít tato zadání dvojím způsobem – může je zapracovat do své vlastní přípravy na vyučovací hodinu a žákům potřebné informace sdělit a pracovat s nimi dál formou frontální či skupinové výuky, nebo hotová zadání použít jako pracovní listy pro žáky a zadat je k samostatné práci. Ideální je přitom, podle mého názoru, kombinace obou přístupů – do metodické příručky jsem proto zařadil doporučení pro učitele první šifru z dané metodické řady vytvořit společně s žáky a další úlohy zadat žákům k samostatné práce. Výuka informatiky je dle Průchy (2009) téměř vždy pevně svázána s využitím počítačem, jako didaktického prostředku sloužícího názorně-demonstračním metodám (předvádění a pozorování, instruktáž) a metodám dovednostně-praktickým (napodobování, nácvik dovedností). Názornost je pro výuku informatiky nesmírně důležitá.

Zbývající úlohy pak doporučuji zadat k samostatné práci, a to i v situaci, kdy žáci nemají příliš mnoho zkušeností s programováním. Jak to? První metodická řada je tvořena třemi úlohami zaměřenými na naprogramování monoalfabetických substitučních šifer. Konkrétně se jedná o šifru Atbaš (viz. kapitola 2.1.1) Caesarovu šifru (viz. kapitola 2.1.2) a šifru mistra Jana Husa. Poté, co žáci pod vedením vyučujícího pochopí na příkladu princip první šifry, nebude pro ně problém vytvořit dvě zbývající, jelikož k tomu bude stačit jiné nastavení pořadí písmen v šifrové abecedě. Tento postup je z hlediska programování velmi užitečný, jelikož odpovídá myšlence Niklause Wirtha, že programování = algoritmy a datové struktury. Tzn., že vhodně zvolená datová struktura umožní snadnou modifikaci výsledného programu jen změnou obsahu datové struktury, v tomto případě seznamu, aniž by bylo nutné jakkoliv měnit algoritmus. Druhou metodickou řadu tvoří čtyři úlohy se zaměřením na programování několika různých typů transpozičních šifer. Z hlediska obtížnosti je tato řada jistě náročnější, už jen proto, že zde žáci pracují s novými prvky, se kterými v první metodické řadě nepracovali. Přesto by mělo postačit tvořit první z úloh společně a zbývající zadat k samostatné práci. Poslední úloha (Fleissnerova šifra) je asi nejnáročnější, zde se nabízí prostor pro mimořádně nadané, či pokročilé žáky.

Závěrem bych rád uvedl, že se nedomnívám, že existuje jeden univerzální postup a metoda. Dobrý pedagog postupy a metody kombinuje, vybírá si z nich to nejlepší a přínosné pro danou problematiku. S ohledem na provázanost tohoto materiálu s dějepisem je jistě vhodné zapojit i metody, které jsou více frekventované v humanitních předmětech, jako např. slovní metody ve formě vyprávění nebo metoda v podobě diskuse, či inscenace, kdy se žáci mohou vžít např. do role kryptologů, vojevůdců apod., což může vést k efektivnější fixaci učiva.

V samotné metodické příručce pro lepší přehlednost necituji zdroje, z nichž jsem čerpal při tvorbě úloh. Pro úplnost zde uvedu, že jsem čerpal zejména z práce Pierra Berloquina (2011), Simona Singha (2009) a především Pavla Vondrušky (2006). Jelikož metodická příručka tak, jak jsem ji dal k posouzení skupině učitelů informatiky, má vlastní číslování kapitol a obrázků, rozhodl jsem se ji vyjmout z toku textu diplomové práce a zařadit ji jako samostatnou přílohu.

5 Úvod k empirické části práce

Dle původního zadání bylo mým úkolem úlohy následně aplikovat v praxi a podrobit je dotazníkovému šetření u žáků i vyučujících. Tento záměr do značné míry zkomplikovala nepředvídatelná situace spojená s pandemií Covid-19, kdy byla zavedena mimořádná vládní opatření. Empirické části práce se zásadním způsobem dotklo opatření o uzavření českých škol, které vešlo v platnost 11. března 2020. Bylo mi tak v podstatě znemožněno provést plánovaný výzkum v plném rozsahu, vyučující i žáci pracovali v nestandartních podmínkách a distanční výuka nedávala prostor pro plnohodnotné využití mé příručky v hodinách informačních technologií. Požádal jsem tedy vedoucího diplomové práce o provedení drobných změn v zadání práce. Vedoucí práce mi vyhověl a modifikoval zadání tak, že věty: *„Praktická část práce bude zpracována formou sady programátorských úloh různé obtížnosti, které budou sloužit k dešifrování a šifrování daného systému. Úlohy budou následně aplikovány v praxi a podrobeny dotazníkovému šetření u žáků i vyučujících.“* nahradil formulací: *„Praktická část práce bude zpracována formou metodické příručky pro učitele, jejíž součástí bude sada programátorských úloh různé obtížnosti, které budou sloužit k dešifrování a šifrování daného systému. Metodická příručka včetně úloh a jejich vzorových řešení bude následně podrobena dotazníkovému šetření u vyučujících.“*

V souladu s upraveným zadáním práce jsem se rozhodl učitelům informatiky poskytnout metodickou příručku včetně vzorových řešení, dostupných ve formátu .sb3, a také dotazník. Na mou výzvu zareagovalo celkem 16 respondentů – vyučujících s různě dlouhou pedagogickou praxí.

Nyní již k samotné metodice výzkumu. Pro dotazníkové šetření jsem zvolil strukturovaný dotazník s široce otevřenými otázkami. Odpovědi na široce otevřené otázky mohou obsahovat připomínky, nápady a názory, které mohou být velmi podnětné pro rozvoj práce, což uzavřené otázky nenabízí. Dotazník se sestává s celkem sedmi otázek:

1. Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?
2. Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?

3. V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?
4. V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?
5. Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?
6. Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?
7. Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?

Cílovou skupinu dotazníkového šetření tvoří učitelé informatiky s různou délkou pedagogické praxe. Pro zachování anonymity respondentů neuvádím jejich jména a místo působení. Respondenty představuji pouze pod kódem sestávajícím se z následujících údajů: pohlaví, rok narození a roky pedagogické praxe. Tedy pro příklad: dotazník vyplněný mužem narozeným v roce 1987 s šesti lety pedagogické praxe nese kódové označení: M8706. (v případě ženy s obdobnými údaji by se jednalo o kód Z8706)

6 Vyhodnocení dotazníků

Následující část práce je věnována analýze odpovědí na dotazníkové otázky. Dříve než začneme analyzovat odpovědi na jednotlivé otázky, podívejme se na složení skupiny respondentů. O vyplnění dotazníku bylo požádáno 30 učitelů informatiky, z toho 15 žen a 15 mužů různého věku, a tedy i s různou délkou pedagogické praxe. Vyplněny se vrátily dotazníky od 16 respondentů, z toho 7 žen a 9 mužů. Nejstarším účastníkem šetření je muž narozený v roce 1963, nejmladší je žena narozená v roce 1996. O něco podrobněji lez vyčíst věkové složení z tabulky č. 12.

Tabulka č. 12 Věkové složení skupiny respondentů

Narozeni v dekádě	Ženy	Muži	Celkem
60. léta 20. století	0	1	1
70. léta 20. století	0	0	0
80. léta 20. století	0	5	5
90. léta 20. století	7	3	8

Věkovému složení zhruba odpovídá délka pedagogické praxe. Mezi důvody, které způsobují výjimky, patří např. prodloužení délky studia, působení v jiném oboru, nebo naopak pedagogická praxe již v době studia. Skutečnou míru korelace jsem ověřil výpočtem s pomocí tabulkového procesoru MS Excel s využitím statistické funkce CORREL(matice1;matice2), která po zadání odpovídajících dat pro rok narození a délku praxe vrátí korelační koeficient s hodnotou -0,986, tj. velmi blízký hodnotě -1, znamenající silnou závislost.. Znaménko mínus značí nepřímou závislost (antikorelaci): čím nižší číslo roku narození, tím delší pedagogická praxe.

Tabulka č. 13a Data pro výpočet korelace mezi rokem narození a délkou praxe – muže

Respondent	1	2	3	4	5	6	7	8	9
Rok narození	1963	1980	1985	1986	1987	1989	1993	1994	1995
Délka praxe	32	15	10	9	6	8	3	1	2

Tabulka č. 13b Data pro výpočet korelace mezi rokem narození a délkou praxe – ženy

Respondent	1	2	3	4	5	6	7
Rok narození	1991	1991	1993	1994	1995	1995	1996
Délka ped. Praxe	3	5	2	2	1	2	3

6.1 Analýza opovědí na 1. otázku

V první otázce jsem se respondentů zeptal: *„Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?“* Většina respondentů hodnotí možnost zařazení tématu historického šifrování do výuky informatiky jako pozitivní, více, či méně kritické jsou jen 4 odpovědi ze 16, které popisují možná rizika. Pravděpodobně nejkritičtější je hodnocení respondentky Z9603, které uvedu v nezkrácené podobě: „*Záleží na provedení. Při propojení těchto dvou témat hrozí sklouznutí k “oslím můstkům” a k nelogickému propojování. Naopak využití historických souvislostí, důvodů, proč byla která technologie potřeba, může být ku prospěchu věci (příklad – Alan Turing a rozluštění Enigmy + 2. světová válka)*“. Tato kritika je však velmi obecná a ke zlepšení mé konkrétní metodiky mi mnoho informací nepřinesla. Poněkud skeptický je respondent M8510, když píše: *„S příklady pro programování je ale problém, protože některé žáky nemusí nadchnout skoro nic.“*, ale vzápětí pokračuje: *„Ale zašifrovat zprávu pomocí programu ve skupinkách a pak dané zprávy vyhodnocovat může být zajímavá aktivita pro žáky při hodinách.“*.

Vývoj subjektivního názoru popisuje respondent M8609: *„Moc se mi líbila motivační část příručky. Ač jsem byl původně skeptický, vaše řeč mě opravdu motivovala, takže splnila svůj účel. Při čtení jsem si vzpomněl na dva své žáky 8. ročníku ZŠ, kteří jsou oba zapálení do historie, obzvláště středověkých a novověkých konfliktů a bitev. Oba mají také velký zájem o informatiku, takže vím, že taková kombinace se mezi žáky opravdu objevuje. Historické šifrování jako téma programování by jistě uvítali.“* Tato odpověď mě ujistila, že téma diplomové práce je přínosné pro praxi, jestliže se dostane do rukou zapáleného učitele informatiky.

Statistickou analýzou textu jsem vytipoval plnovýznamová slova, která se v odpovědích vyskytují opakovaně. Zařadil jsem pouze slova, která se vyskytla v odpovědích nejméně

pěti respondentů. Výsledek shrnuje tabulka č. 14. Pozitivním výsledkem je vysoká četnost slova „zajímavé“ a slov jemu příbuzných.

Tabulka č. 14 Nejčastěji se vyskytující slova v odpovědích na otázku č. 1

Slovo s vysokou četností	Počet respondentů, kteří je použili	Slova příbuzná a jejich četnost z hlediska počtu respondentů
Programování	9	program (2)
Historické	7	historie (2)
Informatika	7	–
Výuka	7	–
Vhodné	5	–
Zajímavé	5	zájem (3) zajímavější (1) zajímá (1) zajímat (1)

6.2 Analýza opovědí na 2. otázku

Druhá otázka se týkala vhodnosti volby programovacího jazyka: *„Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?“*

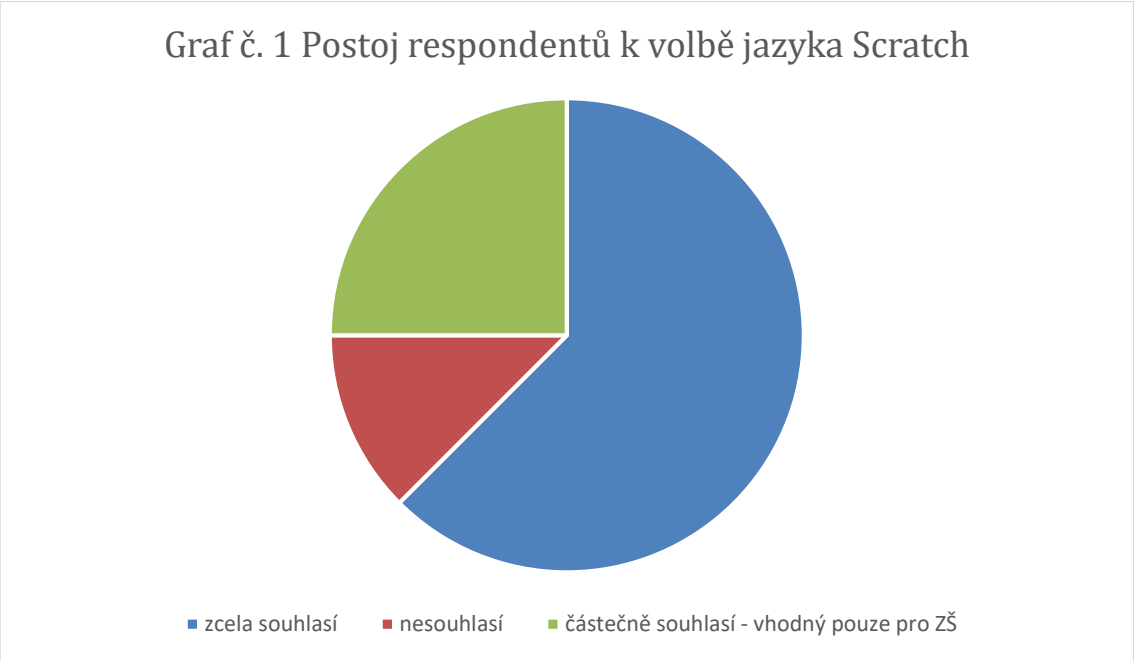
Nejprve se zmíním o možných alternativách, které respondenti navrhují v odpovědi na podotázku. Jedenkrát se zde vyskytuje odkaz na jazyk C#, jedenkrát Kodu (od Microsoft Research Lab). Respondent M9303 sice zmiňuje další dětské programovací jazyky (Baltík, Karel), ale za nevhodnější pro výuku základů programování považuje Scratch. Respondentka Z9603 je i v odpovědi na druhou otázku kritická: *„Nemám dobré zkušenosti s “obrazovými” programovacími jazyky. I když může být textový programovací jazyk složitější k proniknutí, jemné nuance a ošetření a pochopení toho, co se vlastně v programu děje, podle mne, obrazový programovací jazyk neobsáhne.“* Jako alternativu nabízí jazyk Pascal.

Návrh alternativního vývojového prostředí Google Colab pro jazyk Python podrobně odůvodňuje respondent M9502, jehož odpověď uvádím v nezkrácené podobě: *„Scratch podle mě není nejvhodnějším programovacím jazykem pro výuku programování, jelikož se u něho žáci stejně musí naučit všechny skupiny bloků, k čemu se používají a kde je mají*

hledat ve webovém rozhraní, tak aby je mohli vložit do scénáře. Bylo by tedy vhodnější žaky rovnou učit např. v jazyku Python ve spojení s Google Colab (webové IDE), který má dostatečně jednoduchou syntaxi a ve spojení s Google Colab by mohla být dosažena i určitá forma vizualizace a interakce při použití notebooků, kde by se míchal kód společně s výukovým textem. Zároveň pak mohou tento jazyk využít v praxi nebo při dalším studiu na střední nebo vysoké škole.“

Pokud jde o vlastní souhlas s volbou programovacího jazyka Scratch jako vhodného jazyka pro výuku programování, plně s ní souhlasí 10 respondentů, částečně souhlasí další 4, kteří jej považují vhodný pouze pro výuku na základních školách, ale pro výuku na středních by volili jiný jazyk, zcela nesouhlasí dva respondenti, jejichž odůvodnění jsem uvedl výše (respondentka Z9603 a respondent M9502). Zajímavé je, že respondentka Z9402, která považuje jazyk Scratch vhodný pouze pro výuku na základních školách jej současně vidí pro tento účel jako jednoznačně nejlepší řešení: *„Scratch, jako programovací jazyk pro žáky druhého stupně, mi přijde ideální. Z důvodu vizualizace mohou žáci lépe porozumět programování. Myslím, že díky vizualizaci se mohou lépe orientovat v krocích, které programují, a i metodou pokus omyl vyřešit zadané úkoly.“*

Opačný názor na využití zvoleného jazyka ve středoškolské výuce má respondentka Z9501: *„Ano, souhlasím. Scratch je skvělý programovací jazyk, který i starším studentům na středních školách názorně ukáže základy programování. Nemusejí se nijak trápit zapamatováním si složitých zápisů jednotlivých algoritmů, vše již pouze sestavují k sobě a nastaví požadované parametry. Takže pro výuku základů programování naprosto ideální program pro všechny věkové kategorie.“*



Tabulka č. 15 Nejčastěji se vyskytující slova v odpovědích na otázku č. 2

Slovo s vysokou četností	Počet respondentů, kteří je použili	Slova příbuzná a jejich četnost z hlediska počtu respondentů
Jazyk	15	–
Programovací	13	–
Programování	12	–
Škola	11	–
Žák	7	–
Základní	6	základ (4) základna (1)
Vhodný	5	vhodnější (1) nejvhodnější (1)

6.3 Analýza opovědí na 3. otázku

V třetí otázce jsem se respondentů dotazoval: „*V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?*“ Všichni respondenti našli v mé práci něco zajímavého a přínosného. Např. respondent M8908 vidí její klady hned v několika rovinách, proto jeho odpověď uvádím v plném znění: „*Metodickou příručku hodnotím kladně z hlediska její struktury. Učitel je zde totiž zpočátku seznámen s programovacím jazykem Scratch a domnívám se, že i učitel, který se s tímto jazykem*

nikdy nesetkal, se na základě uvedených informací dokáže velmi rychle ve Scratchi zorientovat. Kladně také hodnotím popis a řazení samotných úloh – od nejjednodušších po složitější. Od jednoduchých substitučních šifer přechází autor k šifře Leonarda da Vinci, kde se propojuje práce s textovými řetězci a grafikou, a poté přechází k nejsložitějším transpozičním šifrám.“

Dva respondenti (Z9302 a M8609) dali v odpovědích najevo, že pro ně samotné byl obsah příručky přínosný. „*Mě osobně seznámila s dalšími nástroji Scratche, které jsem neznal. Například mě doteď nenapadlo použít jako proměnnou text. Tomu, kdo Scratch alespoň trochu ovládá, rozšíří příručka obzory, jako je rozšířila mně.“*, přiznává respondent M8609.

Respondent Z9501 vidí největší potenciál v možnosti operovat se zadáními a jejich vzorovými řešeními vytvořenými v programu Scratch. „*Největší předností je samozřejmě možnost čerpat z již vytvořené sady úloh a správných řešení, které autor učitelům nabízí k dispozici. Učiteli tak ušetří poměrně dost času s vymýšlením úloh, a hlavně později s opravou těchto úloh.“* Deset respondentů mezi hlavní klady řadí akcentaci mezioborových vztahů mezi informatikou a dějepisem. Jedním z nich byl respondent M8015: „*Nejvíce se mi líbil historický příběh každé šifry a domnívám se, že právě historie zařazených šifer by mohla být dobrou motivací pro žáky, aby se šifru pokusili vytvořit v programu Scratch.“,* zároveň dodává: „*A více zvídaví žáci se pravděpodobně následně pokusí vytvořit v tomto programu šifru vlastní.“*

Tabulka č. 16 Nejčastěji se vyskytující slova v odpovědích na otázku č. 3

Slovo s vysokou četností	Počet respondentů, kteří je použili	Slova příbuzná a jejich četnost z hlediska počtu respondentů
Příručka	11	–
Scratch	8	–
Šifry	7	dešifrování (1) šifrování (1)
Žák	7	–
Historický	6	historie (2)
Učitel	6	–
Úlohy	5	–

6.4 Analýza opovědí na 4. otázku

Předmětem čtvrté otázky dotazníku byly možné rezervy práce. Respondentů jsem se ptal: *„V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?“* Kromě nekritických odpovědí čtyř respondentů, kteří v metodické příručce jako takové neshledávají žádné zásadní nedostatky, se ve zbývajících dotaznících objevily často velmi podnětné názory. Nejčastěji opakující se výtka se vztahovala k popisu zvoleného jazykového prostředí, část respondentů totiž považuje kapitoly věnující se seznámení s jazykovým prostředím za příliš stručné, jak ostatně uvádí např. respondent M8015: „*Nezbytným předpokladem k přečtení této příručky je předchozí alespoň základní znalost programu Scratch, protože při popisu programu je příručka velmi stručná. Tato příručka tedy nenahrazuje komplexní návod programu, což autor ve 2. kapitole zmiňuje.“*

Na možné úskalí související s učiteli, kteří s výukou programování nemají mnoho zkušeností, poukázal respondent M8609: „*Příručka dle mého skromného názoru není vhodná pro učitele začínající s programováním, jak by se podle prvních stránek mohlo zdát. Ve druhé a třetí kapitole sice seznamuje učitele s prostředím, ale toto seznámení je velice stručné, někdy si vystačí s „postačí se seznámit s…“, což, pokud to není dáno do celkového kontextu, učitelům sice pomůže s řešením úlohy, nikoliv však s pochopením programovacího jazyka. Pokud je tedy příručka pro učitele-nováčky, seznámení s prostředím není rozhodně dostačující a nevyhovují rozhodně ani úlohy, které jsou pro začátek až moc kombinované.“* Zde se však domnívám, že nedošlo k plnému pochopení mého záměru. Je sice pravda, že příručka je určena primárně učitelům, ale je zde také uvedeno, že kapitoly vztahující se k samotnému jazyku Scratch jsou pouze popisem nejnutnějšího základu, který by měli minimálně znát žáci, aby byli schopni efektivně programovat dané úlohy, nikoliv k tomu, co by měl k tomu znát učitel. Schopnosti a dovednosti učitelů informatiky v oblasti programování, s nimiž text příručky počítá, ovšem zpochybňuje respondentka Z9302: *„Bohužel i v této době je plno učitelů informatiky, především starší generace, kteří programování dosud neučili. Nyní je však trendy ve výuce k tomu “nutí” a některým z nich to přináší značné problémy.“*

Respondent M8706 svou odpovědí poukázal na problematiku vztahující se k nízké hodinové dotaci předmětu. „*Dále jako velký problém vidím nedostatek času ve výuce informatiky na ZŠ pro použití této příručky a samotné výuky šifrování.“*

Za asi nejzajímavější podnět k případnému rozšíření metodické příručky lze považovat odpověď respondenta M8908: *„Příručka by mohla obsahovat i stručný komentář autorských řešení úloh, která jsou součástí příručky. Autor by se také mohl zamyslet nad tím, co by žákům mohlo při řešení činit problémy a v rámci metodické příručky navrhnout způsob, jak těmto komplikacím předcházet (například formou návodných otázek nebo úloh…).“*

Tabulka č. 17 Nejčastěji se vyskytující slova v odpovědích na otázku č. 4

Slovo s vysokou četností	Počet respondentů, kteří je použili	Slova příbuzná a jejich četnost z hlediska počtu respondentů
Příručka	9	–
Žák	9	–
Úlohy	6	–
Nedostatky	5	nedostatečné (1)

6.5 Analýza opovědí na 5. otázku

V otázce č. 5 jsem se respondentů ptal na využití úloh, jež jsou součástí příručky: *„Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?“* Asi nejkomplexnější odpověď, včetně nevšedního způsobu využití úloh v praxi, poskytl respondent M6332: „*Úlohy jsou zajímavé a pečlivě jsou zpracována i vzorová řešení. Proto bych se snažil v nějaké podobě využít všechny úlohy. Žákům bych asi zadal k samostatné práci ty jednodušší, což jsou podle mého názoru úlohy na substituční šifry. Složitější, tedy transpoziční šifry, bych zařadil do výuky např. v rámci soutěživých her, spojených s pobytem v přírodě v okolí školy, kdy žáci objeví tajemný dopis a s pomocí počítače a vzorového řešení např. šifrování Fleissnerovou mřížkou dopis šifrují, nebo dešifrují. Originální je šifra Leonarda da Vinci, která navíc ukazuje, jak zdánlivý hendikep (v tomto případě leváctví) se může změnit i ve výhodu.“*

Většina respondentů se shoduje na jednom zásadním faktoru, který hraje klíčovou roli při výběru úloh. Jedná se o jejich obtížnost, kdy je třeba postupovat s ohledem na schopnosti žáků. Ne jinak svůj postoj prezentuje respondent M8908 a nabízí i možný postup: „*Do výuky bych zařadil úlohy založené na principu substitučních šifer. Na těchto úlohách lze s žáky velmi snadno procvičit práci s cykly a práci s textovými řetězci. Jak jsem již zmiňoval, zařadil bych šifru Leonarda da Vinci, protože by byla pro žáky snadno*

pochopitelná a zároveň by se pomocí ní naučili propojovat práci s textovými řetězci a počítačovou grafikou. Transpoziční šifry Skytalé a Fleissnerova mřížka bych použil spíše jako úlohy pro nadanější žáky, protože jejich naprogramování (délka šifrovaného textu, práce s kostýmy a pohyb samotných postav/písmen…) je pro mnohé žáky velmi obtížné.“

Respondenti ve svých odpovědích zohledňují ale i jiná kritéria, jako např. hodinovou dotaci předmětu: „*Zatím nevím, které úlohy bych vybral, zaujaly mě v podstatě všechny. Ale vzhledem k časové náročnosti plnění školních vzdělávacích plánů bych vybral jednu substituční a jednu transpoziční šifru.“* (respondent M8015), či časovou provázanost s právě probíraným tématem v hodinách dějepisu „*Co bych zařadila, by záleželo zrovna na tom, co by žáci probírali v historii. Složitější úlohy bych zařadila až ve vyšších ročnících.“* (respondentka Z9402)

Tabulka č. 18 Nejčastěji se vyskytující slova v odpovědích na otázku č. 5

Slovo s vysokou četností	Počet respondentů, kteří je použili	Slova příbuzná a jejich četnost z hlediska počtu respondentů
Šifra	12	–
Úloha	11	–
Výuka	9	–
Všechny	8	–
Žák	8	–

6.6 Analýza opovědí na 6. otázku

Odpovědi na šestou otázku: *„Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?“* byly o něco stručnější než ty předchozí, ale i přesto poskytly řadu zajímavých a podnětných návrhů. Respondent M8510 má jako jediný z respondentů přímou zkušenost se zařazením šifrování do výuky mimo výuku předmětu informačních technologií. „*Šifrování je u nás ve škole zařazeno v předmětu Číslicová technika, která se vyučuje na elektro oborech i ve strojírenství.“* Zde by jistě byl prostor pro využití mé příručky. V odpovědích někteří respondenti uvedli možnost využití nejen v hodinách přímo provázaného dějepisu, ale např. také v hodinách anglického jazyka, matematiky či výtvarné výchovy. Stejně tak přicházeli v úvahu i různé

kroužky, semináře pro žáky apod.

Někteří respondenti se oprostili od chápání otázky v užším kontextu prostředí školní výuky a navrhli i jiné alternativy: „*Poznatky z této příručky lze poměrně dobře využít i v rámci akcí školy jako jsou různé výlety, adaptační kurzy apod. Aktivity na principu únikových her při teambuildingu. Dokážu si představit takovéto úkoly i během suplovaných či nestandartních vyuč. hodin. Kdy žáci soutěžní formou rozvíjejí své myšlení.“* rozvinul myšlenku respondent M9401.

Z pozice učitele dějepisu mě zaujal konkrétní nápad respondentky Z9502 pro využití vzorových řešení úloh, jež jsou součástí příručky: „*Logické by bylo využít příručku v hodině dějepisu při výkladu o šifrách. Akorát programování bych zde zmínila pouze okrajově, jako možnost. Případně by bylo možné použít již naprogramované úlohy jako ukázku, jak šifry fungují. Žáci by tak mohli lépe pochopit princip šifer.“*

S ohledem na velké množství různých návrhů využití metodické příručky jsem vytvořil přehlednou tabulku, která reflektuje všechny uvedené možnosti a počet jejich navrhovatelů.

Tabulka č. 19 Alternativy využití metodické příručky mimo výuku informatiky

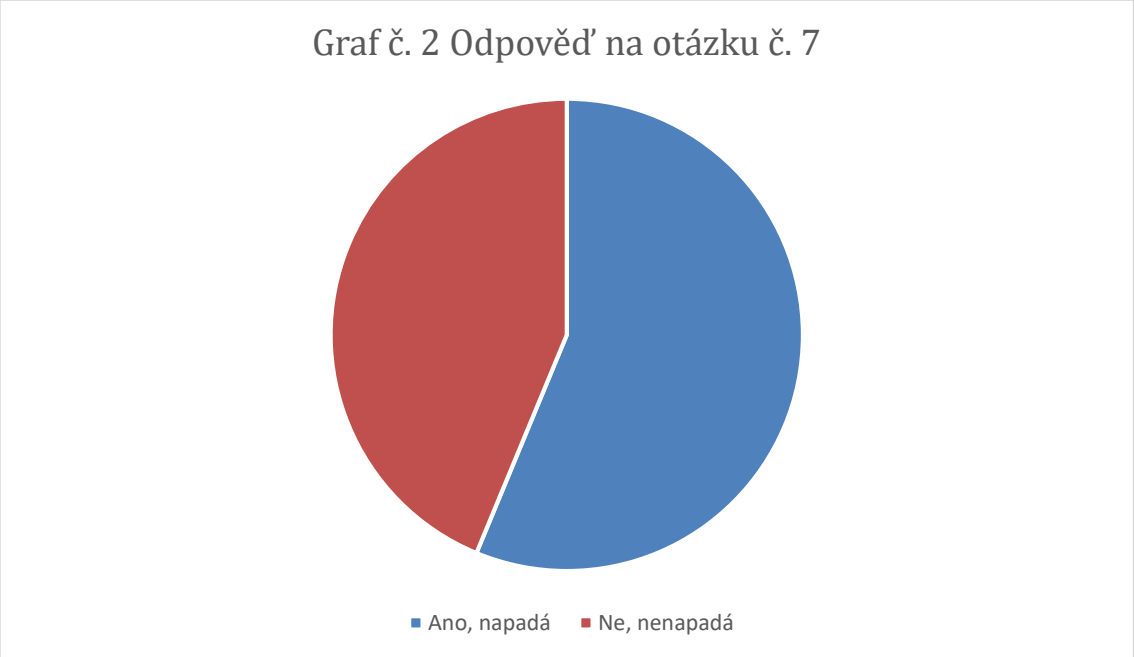
Návrhy respondentů	Počet respondentů, kteří možnost navrhli
dějepis (školní předmět)	5
kroužek programování	3
školní akce (výlety, adaptační kurzy, únikové hry)	3
anglický jazyk (školní předmět)	2
matematika (školní předmět)	2
výtvarná výchova	2
Žádné	2
číslicová technika (školní předmět)	1
letní tábory	1
robotický kroužek	1
suplování a netradiční vyuč. Hodiny	1
tělesná výchova (školní předmět)	1
turistické aktivity	1

Tabulka č. 20 Nejčastěji se vyskytující slova v odpovědích na otázku č. 6

Slovo s vysokou četností	Počet respondentů, kteří je použili	Slova příbuzná a jejich četnost z hlediska počtu respondentů
Žáci	9	–
příručka	8	–
hodiny	7	–
Dějepis	6	–
Šifry	6	–
Kroužek	5	–

6.7 Analýza opovědí na 7. otázku

Poslední dotazníková otázka byla položena ve znění: *„Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?“* Zde už respondenti nebyli tolik tvůrčí, jako v odpovědích na otázku předchozí. Jak ukazuje graf, téměř polovinu respondentů žádné jiné vhodné využití vzorových řešení úloh nenapadlo.



Odpovědi byli povětšinou obdobné, jako u otázky č. 7 – ukazovali na možnost využití ve výuce dějepisu, matematiky na letních táborech apod. S ohledem na tento fakt zde zcela výjimečně necituji žádného z respondentů a čtenáře v případě zájmu odkazuji na přílohy práce. Vzhledem ke stručnosti odpovědí je počet slov s vysokou četností zanedbatelný. (viz. tabulka č. 21)

Tabulka č. 21 Nejčastěji se vyskytující slova v odpovědích na otázku č. 6

Slovo s vysokou četností	Počet respondentů, kteří je použili	Slova příbuzná a jejich četnost z hlediska počtu respondentů
Nenapadá	5	–
řešení	5	–
Využití	5	využít (3)

6.8 Diskuze výsledků výzkumu

Díky dotazníkovému šetření jsem získal potřebná data k výzkumu zaměřenému na metodickou příručku věnovanou využití historických šifer jako motivaci k výuce programování a autorských řešení úloh, jež jsou její součástí. Z původně oslovených 30 respondentů zareagovalo na mou výzvu 16 z nich, jedná se tedy o 53% úspěšnost, která mi postačila k efektivnímu vedení výzkumu. Ten se zakládal na porovnání odpovědí respondentů k daným otázkám a byl podpořen statistickou analýzou textu. Není bez zajímavosti, že některá slova se opakovala často napříč otázkami. Uvedená souhrnná tabulka ukazuje, která slova to byla – představuje konečný počet odpovědí, ve kterých byla daná slova použita. (tabulka se omezuje pouze na pět slov s nejvyšší četností)

Tabulka č. 22 Nejčastěji se vyskytující slova celkem

Slovo s vysokou četností	Celkový počet odpovědí, ve kterých bylo toto slovo použito (včetně slov příbuzných)
Programování	35
žák	33
Příručka	28
Šifry	27
Úlohy	22

Z výše uvedené tabulky je patrné, že kromě slov obsažených ve formulovaných dotazníkových otázkách: **programování**, **příručka** a **šifry** je slovem s nejvyšší četností slovo **žák.** Z toho jasně vyplývá, že vyučující ve svých odpovědích zohledňovali v první řadě své žáky, což z mého pohledu nepochybně svědčí o jisté profesní zralosti dotazovaných respondentů. Samozřejmě jsem si plně vědom, že textová analýza založená na četnosti slov nevede k jednoznačným závěrům. Přínos výzkumu spočívá především ve získaných odpovědích zkušených i méně zkušených kantorů na otevřené otázky. Ty

bezpochyby představují cenný zdroj inspirace a podnětů, které by mohly vést ke zkvalitnění či rozšíření sledované metodické příručky. Obecně mohu konstatovat, že názory na ni byly vesměs kladné a tématu mé práce otevřené. Absolutně se přitom ztotožňuji s názorem respondentky s kódovým označením Z950: *„Samozřejmě nejlepším způsobem, jak zjistit nedostatky či rezervy, je vyzkoušet příručku v praxi.“* Což ovšem, jak jsem již uvedl, nebylo proveditelné s ohledem na nepříznivou epidemiologickou situaci. Dotazovaní učitelé však mají metodologickou příručku včetně autorských řešení úloh k dispozici a podle všeho většina z nich je plánuje, jakmile to bude možné, vyzkoušet v praxi. Díky tomu věřím, že mé počínání mělo smysl.

ZÁVĚR

Cílem teoretické části práce bylo zmapovat historii šifrování s důrazem na klasické ruční substituční šifry od starověku po konec 20. století. Při realizaci tohoto cíle jsem do značné míry vycházel ze zkušeností získaných během tvorby bakalářské práce. Kapitoly jsem ovšem tentokráte nedělil podle dějinných etap, nýbrž podle daných substitučních šifrových systémů. Popsal jsem principy fungování klasických ručních substitučních šifer od starověku až po 20. století, od starověké jednoduché záměny přes digrafické substituční šifry, homofonní substituci a nomenklátory, dále polyalfabetickou substituci s periodickým heslem, až po nejvyspělejší bigramové a polygramové substituční šifry. K těmto systémům jsem uvedl vždy minimálně jednoho typického zástupce, zasadil jej do historického kontextu a na jednoduchém příkladu prakticky demonstroval jeho funkci. S ohledem na praktickou část práce, jejímž obsahem měly být i vybrané transpoziční šifry, jsem se okrajově zmínil o tomto typu šifer a uvedl tři typické zástupce – po jednom pro starověk, středověk a novověk, abych tak i v tomto případě pokryl kýženou historickou škálu. Tím jsem splnil stanovený cíl teoretické části práce.

Cílem praktické části práce bylo historické šifry využít jako motivaci k výuce programování. Za tímto účelem jsem měl za úkol vypracovat sadu programátorských úloh různé obtížnosti, které mají sloužit k šifrování a dešifrování daného systému. Tyto úlohy měly být následně aplikovány v praxi a podrobeny dotazníkovému šetření u žáků i vyučujících. Jak jsem ve své práci uvedl, realizaci praktické části práce značně zkomplikovala epidemiologická situace a s ní spojená vládní opatření, která mj. vedla k uzavření českých škol. Za takové situace nebylo možné dostát všem stanoveným závazkům v plném rozsahu a vedoucí práce se proto rozhodl zadání mírně upravit. I nadále bylo mým úkolem vytvořit sadu programátorských úloh, které ovšem měly být nově součástí metodické příručky určené učitelům, a to včetně vzorových řešení. Tato příručka, zahrnující i ony úlohy a jejich vzorová řešení, pak měla být podrobena dotazníkovému šetření u vyučujících.

Zřejmě nejzásadnějším krokem k naplnění cíle praktické části práce bylo rozhodnout se pro vhodné jazykové prostředí. Takové, které by bylo s programováním začínajícím žákům uživatelsky blízké a příjemné, aby vzbuzovalo v žácích chuť programovat. Věřím, že volba vizuálního jazyka Scratch byla správná. Na základě této volby jsem v tomto prostředí vytvořil sadu vzorových řešení k úlohám, jejichž zadání jsem vytvořil předtím.

Jedná se tedy o soubor autorských úloh, které jsou součástí metodické příručky pro učitele, kteří hledají nové způsoby a cesty, jak své žáky motivovat k výuce programování. Poté jsem vybraným vyučujícím informatiky rozeslal dotazníky, které měly prověřit zájem o danou problematiku a vhodnost užití na základě jejich subjektivního hodnocení. Vyplněné dotazníky přinesli nepřeberné množství údajů, které jsem následně vyhodnotil ve výzkumné části práce za dodržení předem stanovených metod. Cíle praktické části práce jsem tedy také splnil.

Tvorbou práce jsem se mnohému naučil. Hned zpočátku jsem si uvědomil, jak obtížný cíl jsem si vlastně uložil. Ne ani tak stran technického provedení, jako spíše psychologického rázu. Motivovat žáky k čemukoliv je v dnešní konzumní a informacemi přehlcené společnosti stále těžší a těžší. Přitom všichni dobře víme, jakou sílu motivace má. Jsem proto i nadále přesvědčen, že i kdyby mé snažení mělo být jen pokusem, i kdyby se měla jiná cesta ukázat správnější, snažil jsem se a udělal jsem dobře. Neboť by bylo hloupé přestat jen na chvíli věřit, že motivovat má smysl. Slavný americký řečník Zig Ziglar řekl: *„People often say that motivation doesn´t last. Well, neither does bathing that´s why we recommend it daily.“;* v překladu: *„Lidé často říkají, že jim motivace nevydrží. No, to ani koupel. Proto ji doporučujeme denně.“* Budu hrdý na to, když má práce někomu z kolegů poslouží jako „voňavá pěna do koupele“.

LITERATURA

BERLOQUIN, Pierre. *Skryté kódy a velkolepé projekty.* 1. vyd. Praha: Knižní klub, 2011. ISBN 978-80-242-2847-1.

BOONE, J. V. *Brief History of Cryptology.* 1st ed. Annapolis: Naval Institute Press, 2005. ISBN 1-59114-084-6.

HALOUSKOVÁ, Alena. *Učebnice programování jazyka Sratch* [online]. 2013 [cit. 2020-05-22] dostupné z:<https://is.muni.cz/th/vrs79/>

JANEČEK, Jiří. *Rozluštěná tajemství.* 2. vyd. Praha: Petr Tychtl - Nakladatelství XYZ, 2008. ISBN 978-80-86864-96-9.

KLÍMA, Vlastimil. Utajené komunikace – 4. díl. Od novověku do 20. století. *Chip.* srpen 1994, 4. ročník, číslo 8, s. 118 – 121. ISSN 1210-0684.

Kolektiv. *Frekvence písmen, bigramů, trigramů, délka slov* [online]. Brno: Centrum zpracování přirozeného jazyka Fakulty informatiky Masarykovy univerzity, 2008. [cit. 2012-01-10] Dostupné z: <http://nlp.fi.muni.cz/cs/Frekvence_pismen_bigramu_trigramu_delka_slov>

MAŇÁK, Josef. *Nárys didaktiky*. 3. vyd. Brno: Masarykova univerzita, 2003. ISBN 80-210-3123-9.

MUSÍLEK, Michal. *Šifry a kódy* [online]. 2010 [cit. 2012-01-10] Dostupné z: <http://www.musilek.eu/michal/sifry.html?menu=mat>

MUSÍLEK, Pavel. *Historie šifrování.* Hradec Králové, 2017. Bakalářská práce na Přírodovědecké fakultě Univerzity Hradec Králové. Vedoucí diplomové práce Štěpán Hubálovský. 30 s.

PRŮCHA, Jan (ed.). *Pedagogická encyklopedie*. Vyd. 1. Praha: Portál, 2009. ISBN 978-80-7367-546-2.

SINGH, Simon. *Kniha kódů a šifer.* 2. vyd. Praha: Argo a Dokořán, 2009. 384 s. ISBN 978-80-7363-268-7 (Dokořán), ISBN 987-80-257-0144-7 (Argo).

VONDRUŠKA, Pavel. *Kryptologie, šifrování a tajná písma.* 1. vyd. Praha: Albatros, 2006. ISBN 80-00-01888-8.

VONDRUŠKA, Pavel. Přehled a historie polyalfabetických šifer. *Crypto-World, informační sešit GCUCMP,* prosinec 2007, ročník 9, číslo 6, s. 2 – 11. ISSN 1801-2140.

VORLÍČEK, Jaroslav. Řešení úloh č. 7-9.*Crypto-World, informační sešit GCUCMP,* prosinec 2003, ročník 5, číslo 12, s. 9 – 20. ISSN 1801-2140.

PŘÍLOHY

A. Metodická příručka (včetně dotazníku)

B. Dotazník

C. Odpovědi respondentů na otázky dotazníku

A. Metodická příručka

Historie šifrování jako motivace k výuce programování

Metodická příručka pro učitele

ÚVOD

Pojetí výuky informatiky na základních školách je stále častěji předmětem nejrůznějších odborných i laických diskusí. Názor, že je zapotřebí ve výuce informatiky více podporovat informatické a algoritmické myšlení u žáků se objevuje stále častěji. Jde v podstatě o reakci na kritiku některých odborníků, kteří jsou nespokojeni se současným stavem IT vzdělávání na školách. Ten je podle nich v mnoha ohledech zastaralý, jelikož vychází z RVP z roku 2004, zatímco dynamický svět informačních technologií v uplynulých letech prodělal a stále prodělává ohromný vývoj. Ať už s danou kritikou plně souhlasíme, souhlasíme částečně, či nesouhlasíme, je nezpochybnitelné, že programování nás stále více obklopuje a proniká do nejrůznějších oborů lidské činnosti. Programování umožňuje např. robotizaci firemních procesů a stále více pracovních pozic tak obsazují stroje namísto lidské síly. Jakkoliv se nám při pohledu zblízka může zdát proces automatizace neúprosný, je třeba si uvědomit následující. Programování je obor, který má podle odhadů odborníků velmi slibnou budoucnost. Poptávka po kvalitních programátorech roste a je dost dobře možné, že v budoucnu řada profesí zcela nebo částečně vymizí a znalost algoritmizace a programování se stane jednou z klíčových dovedností k získání bohaté škály pracovních pozic.

Odhlédneme-li od výše uvedeného a podíváme se na programování očima běžného žáka v prostředí základní školy v České republice, kde se podle průzkumů algoritmizace ani programování na většině škol neučí, může to být pohled plný nejistoty a obav z neznámého. A najít způsob, jak děti motivovat k praktickému programování není vůbec snadné.

Cílem této metodické příručky je usnadnit učitelům informatiky práci v oblasti výuky programování na základních školách, a to souborem zpracovaných úloh v jednom z tzv. dětských programovacích jazyků s názvem Scratch. Přednosti vývojového prostředí Scratch přehledně shrnul Michal Musílek (2013, dostupný online: https://docplayer.cz/14988624-Project-scratch-michal-musilek.html) a právě výhody zde uvedené byly rozhodující pro volbu konkrétního programovacího jazyka.

Závěrem chci uvést, v čem vidím potenciál tohoto projektu. Prvním faktorem je, jak jsem již uvedl, zvolené vývojové prostředí. Jeho hlavní výhodu proti jiným programovacím jazykům vnímám v nahrazení nutnosti znát příkazy daného jazyka a jejich přesnou syntaxi intuitivním sestavováním příkazů do celku složeného z do sebe vzájemně

zapadajících dílků, což je činnost žákům základní školy dobře známá. Většina z nich si ještě nedávno hrála nebo dokonce ještě stále občas hraje se stavebnicí typu LEGO a lze říct, že sestavovaní algoritmů v prostředí Scratch jim může tuto zábavnou činnost připomínat. Druhým faktorem je volba zadání úloh. Všechny úlohy spojuje téma historických šifer. Domnívám se, že akcentování mezioborových vztahů je mocný nástroj, který může žáky přivést od více preferovaného předmětu k dosud méně oblíbenému. Vycházím z předpokladu, že žáci, kteří jsou spíše humanitně zaměření a upřednostňují např. dějepis, mohou prostřednictvím daných úloh získat zájem o dosud méně preferovanou informatiku a naopak žáci, kteří dosud měli zájem o obory technického zaměření, si mohou prostřednictvím historicky laděných programátorských úloh najít cestu k dějepisu.

Text příručky vychází z předpokladu, že žáci již v minulosti byli obeznámeni se základy algoritmizace a programování.

1 MOTIVACE

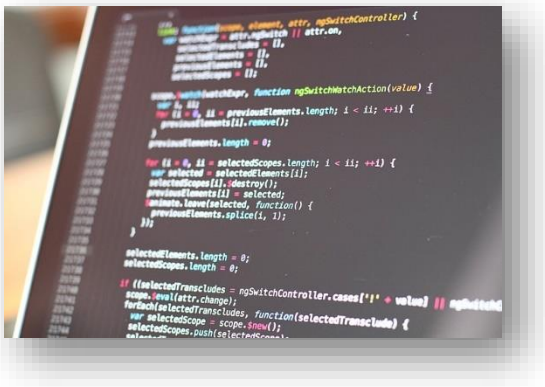
V úvodu práce by bylo vhodné žáky nejprve motivovat. Možná dosud ani netušili, kolik společného mohou mít předměty dějepis a informatika. Odedávna se lidé zabývali otázkou, jak někomu bezpečně a bez obav odeslat zprávu s tím, že i kdyby se náhodou dostala do nesprávných rukou, není se čeho obávat. Na utajení informací před nepovolanými osobami totiž často závisel úspěch politických intrik, či strategie použité v důležitých bitvách. Již ve starověku měli zejména politici a vojevůdci potřebu utajit některé důležité zprávy před svými soupeři a nepřáteli. Dalo by se říct, že právě tehdy vzniká obor lidského zájmu, který je živý dodnes. Jedná se o tzv. kryptologii – vědu zabývající se šifrováním a dešifrováním zpráv. S příchodem počítačů se stala základem pro elektronické šifrování, bez nějž by se informační technologie v současnosti prakticky neobešly. Ani původní historické šifry nebyly zapomenuty a často se dnes využívají jako forma zábavy a trávení volného času (např. na dětských táborech, jako součást únikových her, v rámci tzv. geocachingu, nebo ve speciálním typu soutěží, které spojují luštění šifer s orientací v neznámém terénu, tzv. šifrovačkách).

Žáky je poté třeba obecně obeznámit s tím, co bude jejich úkolem. Zadání obsahuje několik úloh, jejichž cílem je vytvořit v daném programovacím jazyku algoritmus realizující zadaný šifrový systém na základě uvedených instrukcí.

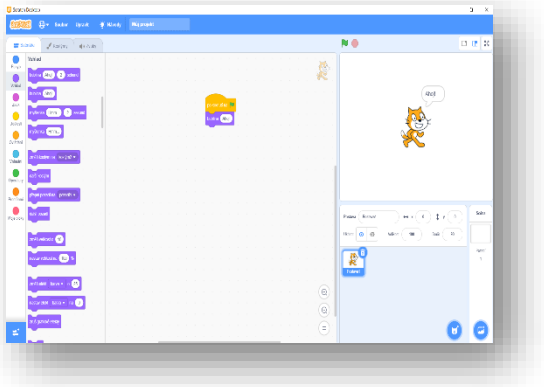
2 PROGRAMOVACÍ JAZYK SCRATCH

Dalším krokem by mělo být žáky stručně seznámit s prostředím, ve kterém budou pracovat. Následující text se soustřeďuje prvotně na ty informace, které souvisí s řešením zadaných úloh a nepředstavuje komplexní informace k programovacímu jazyku Scratch. Pokud v něm žáci již pracovali při řešení jiných typů úloh, je možné tento krok přeskočit a přejít k zadání úloh na šifrování.

Scratch je v první řadě **programovací jazyk**. Programovací jazyky můžeme rozdělit do dvou základních skupin - na programovací jazyky **textové** a programovací jazyky **vizuální**. Na obrázcích níže máte pro lepší představu zástupce obou skupin. Textový jazyk zastupuje na obr. 1 jazyk jménem Java. Vizuální jazyk (obrázek vpravo) zastupuje na obr. 2 právě námi zvolený jazyk Scratch.



Obr. 1 Textový programovací jazyk



Obr. 2 Vizuální programovací jazyk

Rozdíl tkví v tom, že textové programovací jazyky nám umožňují vytvářet programy na základě psaní textu (písmeno po písmenu, znak po znaku) v podobě řádků s jednotlivými příkazy programu, zatímco vizuální jazyky umožňují vytváření programů díky manipulaci s grafickými objekty, připomínajícími kostky stavebnice.

Jak je vidět na obrázku vpravo, v pravém horním rohu vývojového prostředí Scratch je tzv. **„scéna.“** Podobně jako herci na scéně předvádí své umění, program na scéně prezentuje své výsledky. Respektive, prezentuje je také jakýsi herec. **Výsledky** programu totiž oznamuje tzv. **„postava.“** Na scéně je základní postava **kocoura** a u ní textová bublina. Jednoduše - poté, co program žáci vytvoří, bude to právě tento kocour, kdo oznámí výsledek jejich programu.

K tomu, aby jednotlivé postavy, včetně kocoura, dělaly, co mají, slouží tzv. **příkazové bloky**, které se skládají do sebe podobně, jako třeba **stavebnicové kostky**. Jednotlivé příkazové bloky do sebe zapadají pouze v případě, že jsou k sobě skládány správným způsobem tak, aby byla správná **syntaxe** složených příkazů. To žáky v roli programátorů osvobozuje od nutnosti neustále kontrolovat formální správnost syntaxe příkazů a umožňuje jim plně se soustředit na podstatu problému, tj. na vytvoření programu.

3 PŘÍKAZOVÉ BLOKY

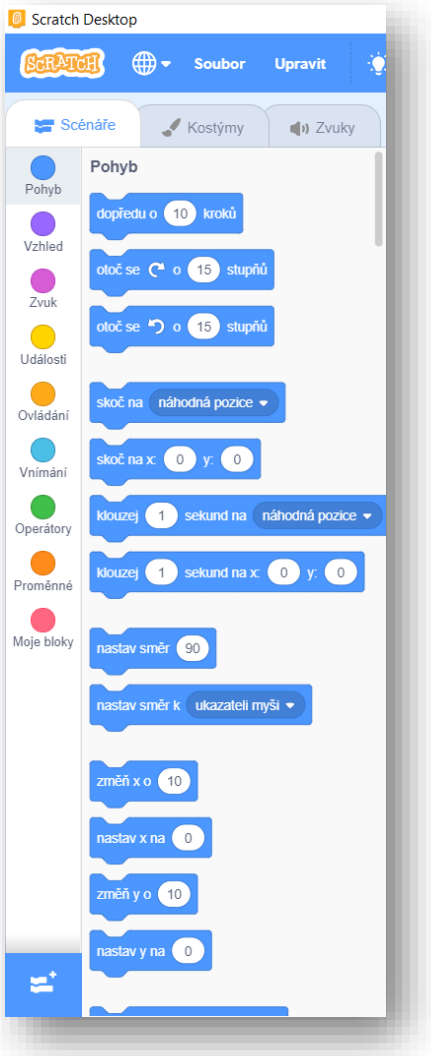
Příkazové bloky dělíme ve Scratchi do celkem devíti skupin a ze zásobníku se vkládají do složek zvaných **scénáře.** Pro lepší přehlednost jsou skupiny bloků rozlišeny **barvou** a dané formy příkazů **tvarem**. Nespornou výhodou těchto bloků je, že žáci nemusí znát zpaměti slovíčka reprezentující příkazy jazyka a mohou se tak intenzivněji zaměřit na samotnou algoritmizaci.

Žáci budou muset při řešení úloh pracovat se skupinami bloků: **Pohyb, Vzhled, Události, Ovládání, Vnímání, Operátory, Proměnné a Moje bloky** (v programu jsou seřazeny ve sloupci vlevo, viz obrázek). Zbývající příkazový blok: **Zvuk** můžeme v tuto chvíli zanedbat – není nezbytné s ním pracovat.

3.1 PŘÍKAZOVÉ BLOKY – DĚLENÍ PODLE BARVY

UDÁLOSTI (žluté bloky)

Příkazy v této skupině slouží k tomu, abychom vůbec postavu a tím pádem celý program aktivovali. Zde nastavíme, které příkazy „odstartují“ navazující příkazy. Např. můžeme nastavit, která z ikon, kláves, či zpráv daný program spustí.



Obr. 3 Příkazové bloky

OVLÁDÁNÍ (okrové bloky)

Blok Ovládání obsahuje všechny základní řídící struktury, které můžeme znát i z jiných programovacích jazyků. Obsahuje podmíněné příkazy, větvení funkcí, různé formy cyklů. Můžeme také klonovat (rozšiřovat) dané objekty programu.

VNÍMÁNÍ (světle modré bloky)

Vnímání obsahuje nástroje, které slouží k tomu, aby postava vnímala zbývající obsah programu, respektive s ním komunikovala. Při řešení daných úloh nám postačí seznámit se s příkazem otázka (tzn. *„ptej se na“*), kdy si definujeme odpověď, která provede předem definované změny v programu.

VZHLED (bloky švestkové barvy)

Vzhled nám umožňuje definovat, jak postava bude vypadat. Při naší práci nebude ani tak důležitá grafická a estetická stránka, ale důležitou roli pro nás bude hrát v komunikaci s uživatelem, kdy budeme moci vytvářet komiksové bubliny s definovaným textem.

POHYB (tmavě modré bloky)

Pohyb nám umožňuje pohybovat s postavou a dalšími objekty scény. V momentě, kdy budeme pracovat s transpozičními šiframi, bude pro nás tato možnost prakticky nezbytná, jelikož ty jsou postaveny na změně pozic daných znaků textu. (Pohyb aplikujeme na postavy s kostýmy znaků příslušné abecedy.)

OPERÁTORY (zelené bloky)

Operátory obsahují základní i pokročilejší matematické funkce, ale i aritmetické a logické operátory či generátor náhodných čísel.

PROMĚNNÉ (oranžové bloky)

Práce s proměnnými pro nás bude naprosto klíčová. Substituční šifry jsou postaveny na práci s proměnnými. Daným znakům otevřeného textu na základě daného systému přiřadíme znaky textu šifrového.

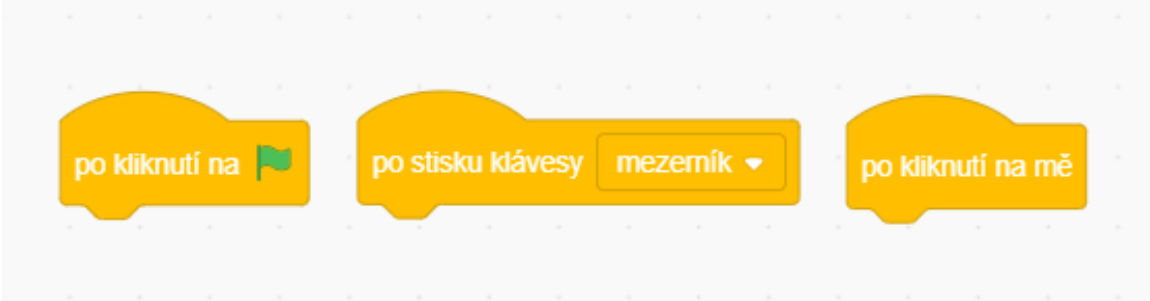
MOJE BLOKY (červené bloky)

Moje bloky nám umožňují vytvářet jakési podprogramy, které budou součástí hlavního programu, díky tomu nemusíme daný program vypisovat vícekrát.

3.2 PŘÍKAZOVÉ BLOKY – DĚLENÍ PODLE TVARU

HRANATÉ KOSTKY S VÝSTUPKEM A OBLOUKOVÝM HORNÍM OKRAJEM

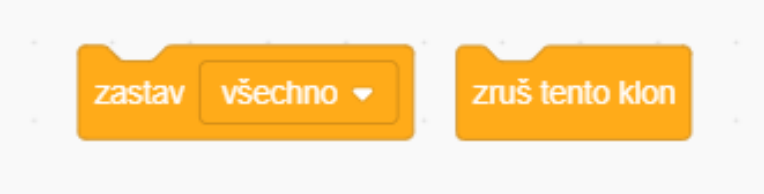
Pro příkazy spouštějící dané skripty. Typický příklad je příkaz typu START.



Obr. 4 Příkazové bloky typu START

HRANATÉ KOSTKY SE ZÁŘEZEM A ROVNOU SPODNÍ HRANOU

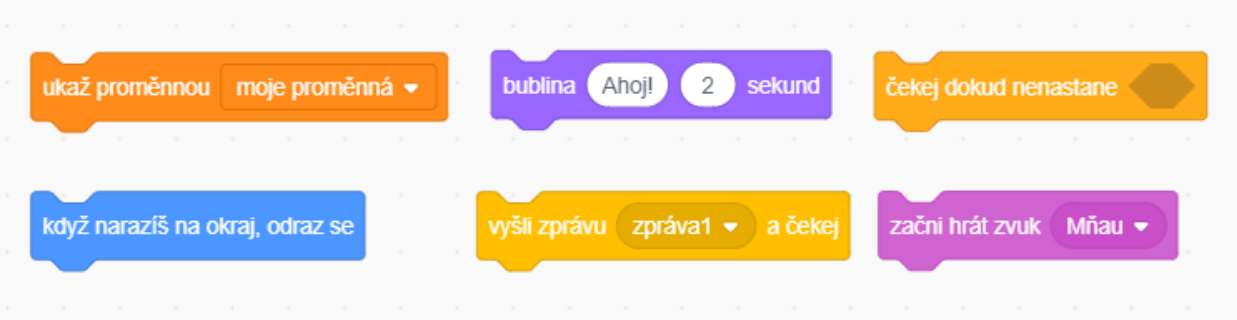
Pro příkazy ukončující dané skripty nebo také nekonečný cyklus. Typický příklad je „zastav“.



Obr. 5 Ukončující příkazové bloky

HRANATÉ KOSTKY S VÝSTUPKEM A ZÁŘEZEM

Pro výkonné příkazy a řídící příkazy podmínek a konečných cyklů.



Obr. 6 Výkonné příkazové bloky

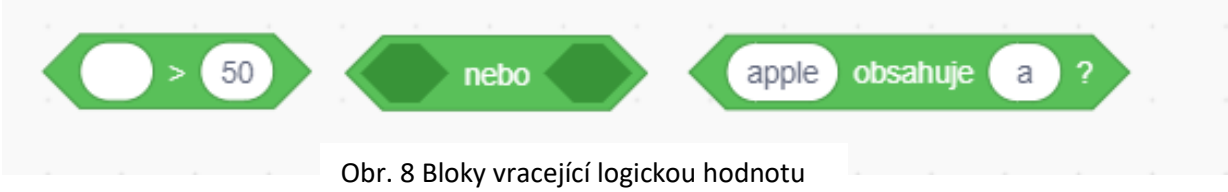
OVÁLNÉ KOSTKY

Pro funkce vracející číselnou hodnotu proměnných a seznamy.

Obr. 7 Bloky vracející číselnou hodnotu

ŠESTIÚHELNÍKOVÉ KOSTKY

Pro funkce vracející logickou hodnotu. (ANO/NE)



Obr. 8 Bloky vracející logickou hodnotu

4 TVORBA POSTAV

Poslední důležitou funkcí, se kterou budou žáci při plnění úloh pracovat, je tvorba nových postav. U postavy můžeme nadefinovat její vzhled a přiřadit k ní dané bloky příkazů. Velkou výhodou, kterou je třeba zdůraznit, je, že postavy a jejich „obsah“ můžeme kopírovat, či exportovat a využít ji třeba v jiné z úloh.

5 PRAKTICKÁ UKÁZKA

Než přejdete k zadání samotných úloh, je vhodné vše demonstrovat na triviálním příkladu. Tím může být např. tvorba jednoduché sčítačky. Při ní využijeme poměrně širokou škálu příkazových bloků. Je vhodné, aby si žáci sami vyzkoušeli realizaci společně s vyučujícím, který každý krok postupně okomentuje. Jednu z možných variant řešení vidíme na obr. 9.

Obr. 9 Program, který sečte dvě zadaná čísla

6 ÚLOHY

Jak již bylo zmíněno v úvodu, následující úlohy jsou věnovány programování doložených historických šifrovacích systémů. Úlohy jsou seřazeny vzestupně od jednodušších po složitější. První tři šifry tvoří metodickou řadu, kdy první šifru může učitel ukázat a další dvě zadat žákům k samostatné práci, i když nemají s programováním velké zkušenosti. Stačí totiž pouze jinak nastavit pořadí písmen v šifrové abecedě. (jde o jednoduché substituční šifry) Tento postup je z hlediska programování velmi užitečný. Odpovídá myšlence Niklause Wirtha, významného švýcarského informatika, že programování = algoritmy a datové struktury. V tomto případě vhodně zvolená datová struktura umožní snadnou modifikaci výsledného programu jen změnou obsahu datové struktury, v tomto případě seznamu, aniž by bylo nutné jakkoliv měnit algoritmus. Dalšími jednoduchými obměnami bychom získali šifry Albam, či Augustovu šifru. Rozšíření algoritmu by umožnilo operovat s několika posuvnými šiframi (typu Caesar, ale s různými posuvy) a pracovat s polyalfabetickými substitucemi jako Vigenérova šifra, Beaufortova šifra či šifra Giambattista della Porta (to už je ovšem spíš úroveň pro střední školy, nikoliv základní, a proto zde úlohy k vytvoření takových šifer nejsou zařazeny). Zbývající čtyři zadání také tvoří metodickou řadu. Vedou k tvorbě šifer transpozičních, jejichž princip spočívá v záměně pořadí znaků otevřeného textu. I zde je vhodné, aby učitel první šifru tvořil společně s žáky a teprve poté zbývající zadal k samostatné práci.

Aby bylo dostáno propojování mezioborových vztahů mezi dějepisem a informatikou, je v zadání každé z úloh uvedeno něco málo z historického kontextu doby vzniku dané šifry. Úlohy pokrývají období od starověku až po počátek 20. století a jsou tak zároveň sondou do oblasti historie šifrování.

Součástí metodické příručky jsou i vzorová řešení uvedených úloh, která mohou učiteli usnadnit práci při konstrukci řešení.

1. ÚLOHA: ŠIFRA ATBAŠ

První šifra, kterou se pokusíme naprogramovat, je šifra substituční. Princip takové šifry spočívá v nahrazení znaků otevřeného textu (textu, který chceme zašifrovat) znaky textu šifrového. Jde o jeden z nejstarších způsobů šifrování a objevuje se již ve starověku. A právě ze starověku pochází i šifra, kterou se nyní pokusíme naprogramovat.

Věděli jste, že v Bibli (Starý zákon, Jeremiáš, kapitola 25, verš 26 a kapitola 51, verš 41) se namísto názvu města Babel, (česky Babylon) objevuje pojmenování Šešak? Zde byla uplatněna šifra známá jako Atbaš. Jde právě o jednoduchou substituci, kdy první písmeno hebrejské abecedy alef („A“) bylo nahrazeno posledním tav („T“), druhé bet („B“) předposledním šin („Š“) atd. (odtud název A-T-B-Š … Atbaš). Koho to zajímá, může si hebrejskou abecedu, která má 22 písmen, vyhledat např. na Wikipedii. Použijeme-li stejný přístup na mezinárodní šifrovou abecedu (latinku bez diakritiky; chcete-li anglickou abecedu) pak A bude nahrazeno písmenem Z, B písmenem Y atd. Neboli, první písmeno bylo nahrazeno posledním, druhé předposledním atd. Pravidla záměny písmen při šifrování (a v případě šifry Atbaš i při dešifrování) vhodně znázorníme zkrácenou převodovou tabulkou:

Tabulka č. 1 Převodová tabulka šifry ATBAŠ

A	B	C	D	E	F	G	H	I	J	K	L	M
Z	Y	X	W	V	U	T	S	R	Q	P	O	N

Abychom se ujistili, že principu šifry rozumíme, ukážeme její funkčnost na jednoduchém příkladu:

Zašifruji zprávu: Chodím do základní školy. (pro zvýšení bezpečnosti šifry je zvykem zrušit rozdělení do slova a nahradit ho rozdělením do pětic znaků)

```
OTEVŘENÝ TEXT: ZMATE NIJAZ YKU
ŠIFROVÝ TEXT:  ANZGV MRQZA BPF
```

Pokuste se v programovacím prostředí Scratch vytvořit program realizující uvedenou šifru. Program musí umět převést všechny znaky české abecedy (bez diakritiky a písmene „ch“ - tedy 26 znaků) a musí být schopen zašifrovat libovolně dlouhý text (maximální délka textu bude omezena pouze parametry vývojového prostředí Scratch)

Funkčnost programu můžete ověřit tím, že zadáte příkladovou větu do systému a výsledky porovnáte.

2. ÚLOHA: CAESAROVA ŠIFRA

Julius Caesar je bezesporu jedno z nejproslulejších historických jmen. Zapsalo se do dějin jako jméno znamenitého vojevůdce, politika, diktátora a muže, který miloval římský lid. Málokdo ovšem ví, že využíval šifru, dnes známou jako Caesarova. O ní se zmiňuje ve svém slavném díle *Zápisky o válce Galské.* A ano, pokud jste si domysleli, že ji využíval právě pro vojenskou komunikaci, máte pravdu! I kdyby se barbaři dokázali šifry zmocnit silou, nepodařilo by se jim šifru rozluštit.

Princip této šifry je přitom triviální. Spočívá v nahrazení každého písmene písmenem, které stojí v abecedě o tři pozice dále. Tedy písmeno A z otevřeného textu je v šifrovém textu nahrazeno písmenem D. Nejlépe to demonstruje převodová tabulka:

Tabulka č. 2 Převodová tabulka Caesarovy šifry (shora dolů šifrujeme, zdola nahoru

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Abychom se ujistili, že principu šifry rozumíme, ukážeme si její funkčnost na jednoduchém příkladu:
Zašifruji zprávu: „Viděl jsem, přišel jsem, zvítězil jsem!“ (rozdělím si pro lepší přehlednost po pěticích znaků)

```
OTEVŘENÝ TEXT: PRISEL JSEM VIDEL JSEM ZVITEZIL JSEM
ŠIFROVÝ TEXT:  SULVHO MVHP YLGHO MVHP CYLWHCLO MVHP
```

Pokuste se v programovacím prostředí Scratch realizovat uvedenou šifru. Program musí umět převést všechny znaky české abecedy (bez diakritiky a písmene „ch“ - tedy 26 znaků) a musí být schopen zašifrovat neomezeně dlouhý text.

Funkčnost programu můžete ověřit tím, že zadáte příkladovou větu do systému a výsledky porovnáte.

3. ÚLOHA: ŠIFRA MISTRA JANA HUSA

Mistr Jan Hus je nesporně významnou osobností českých dějin. Historie si jej připomíná zejména jako významného reformátora a osobnost, jejíž odkaz dal vzniknout husitské revoluci, která otřásla dějinami českých zemí v první polovině 15. století a jejíž učení inspirovalo řadu pozdějších reformačních proudů v Evropě. Zajímavostí, která stojí za povšimnutí, je fakt, že mistr Hus mimo jiné také šifroval. Některé části svých dopisů, adresovaných svým nejbližším z kostnického vězení, na první pohled nedávaly smysl. Na druhý pohled bylo zřejmé, že se jedná o zašifrovanou zprávu.

Přesto by bylo, při vší úctě, poněkud přehnané o mistru tvrdit, že byl nějak významným kryptografem. (ten, kdo se věnuje šifrování). K šifrování svých zpráv využil princip starý stovky let – jednoduchou substituci. Princip spočíval v přeměně samohlásek na písmena, které v abecedě následují po nich. Nejlépe to demonstruje převodová tabulka:

Tabulka č. 3 Převodová tabulka šifry Mistra Jana

A	B	C	D	E	F	G	H	I	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	F	F	G	H	K	K	L	M	N	P	P	Q	R	S	T	V	V	W	X	Y	Z

Jistě vám při pohledu je tabulku něco nápadné, chybí v ní totiž jedno písmeno. Vývoj české abecedy prodělal řadu změn a písmeno J se v české abecedě ustálilo teprve v 16. století, nejde tudíž o překlep, ale historickou přesnost.

Abychom se ujistili, že principu šifry rozumíme, ukážeme si její funkčnost na jednoduchém příkladu:
Zašifruji zprávu: „Kostnice“

```
OTEVŘENÝ TEXT: KOSTNICE
ŠIFROVÝ TEXT:  KPSTNKCF
```

Pokuste se v programovacím prostředí Scratch vytvořit uvedenou šifru. Program musí umět převést všechny znaky české abecedy (bez diakritiky a písmene „ch“ - tedy 25 znaků, zohledňujeme zde dobovou podobu abecedy a vynecháváme také písmeno J) a musí být schopen zašifrovat neomezeně dlouhý text.

Funkčnost programu můžete ověřit tím, že zadáte příkladovou větu do systému a výsledky porovnáte.

4. ÚLOHA – LICHÁ-SUDÁ ŠIFRA

Následující uvedená šifra je velmi stará a zároveň poměrně jednoduchá, takže se nedaří ji přesně datovat, a tak přesně zařadit mezi historické šifry. Historicky je doložena obdobná, ale o něco složitější transpoziční šifra, založená dokonce na použití šifrovací pomůcky, tzv. Skytalé. Než k ní a jí podobným sofistikovanějším transpozičním šifrám přejdeme, ukážeme si princip transpozičních šifer na této nejjednodušší variantě. Na rozdíl od 1. až 3. úlohy, kdy jsme pracovali se substitučními šiframi, budeme nadále pracovat se šiframi transpozičními. Jejich princip spočívá v záměně pořadí písmen otevřeného textu.

Tato šifra nese název lichá-sudá, protože před vlastní transpozicí písmena otevřeného textu rozdělíme na lichá a sudá ve vztahu k jejich pořadí v textu. Lze si to představit také tak, že písmena se „rozpočítají“ podobně jako v hodině tělesné výchovy systémem „první – druhé – první – druhé – první – druhé atd.“. Takovéto rozpočítání lze označit napsáním jedniček a dvojek do pomocného řádku pod otevřený text. Do šifrového textu pak zapíšeme nejprve všechna lichá písmena, tj. „první“ (označená jedničkami), potom teprve všechny sudá písmena, tj. „druhá“ (označená dvojkami). Ukažme si to na příkladu:

```
OTEVŘENÝ TEXT: KDOSE MOCPT AMOCS EDOZV I
POMOCNÝ ŘÁDEK: 12121 21212 12121 21212 1
ŠIFROVÝ TEXT:  KOEOP AOSDZ IDSMC TMCEO V
```

Pokuste se v programovacím prostředí Scratch realizovat uvedenou šifru. Program musí umět převést zobrazit zadané znaky mezinárodní šifrové abecedy jako kostýmy postav v prostředí Scratch a následně je přeskupit, a tak zašifrovat. Vzhledem k nutnosti realizace písmen textu prostřednictvím postav stačí, když bude program schopen zašifrovat text s délkou do 23 písmen.

Funkčnost programu můžete ověřit tím, že zadáte příkladovou větu do systému a výsledky porovnáte.

5. ÚLOHA – SKYTALÉ

Starověké Řecko lze jistě bez nadsázky nazvat kolébkou antické vzdělanosti. Nikoho tedy nepřekvapí, že zde vznikla jedna z nejslavnějších šifrovacích pomůcek vůbec – tzv. Skytalé. Dějiny starověkého Řecka nám ovšem také představily bezpočet udatných válečníků, vojevůdců a vojenských konfliktů. Právě v nich našla Skytalé své uplatnění.

Řecké slovo skytalé ve volném překladu znamená válec. Mohlo se jednat pravděpodobně o dřevěný váleček, na který byl navinut pruh pergamenu. (viz. obrázek níže) Na pergamen se od jednoho konce k druhému napsal otevřený text zprávy. Po rozvinutí text nedával smysl. Poté, co byla zašifrovaná zpráva doručena, adresát opět navinul na váleček daný pruh pergamenu. Bylo nezbytné, aby se odesílatel a příjemce předem dohodli na průměru válce, jinak by text nebylo možné dešifrovat správně.

Tip: Zkuste doma nahradit dřevěný váleček skleničkou a pergamen běžným papírem. Poté, co šifru vytvoříte, naviňte pruh s textem např. na plechovku od barvy, poté opět na původní sklenici a výsledky porovnejte.

Obr. 10 Skytalé

Na rozdíl od předchozí šifry „lichá-sudá“, skytalé nedělí písmena šifrového textu jen na sudá a lichá (tj. jen do dvou skupin), ale na více skupin psaných do odpovídajícího počtu řádků po obvodu šifrovacího válce (válcové tyče) na pergamen na tento válec navinutý. Její Druhý rozdíl mezi šifrou sudá-lichá a šifrou Skytalé je v tom, že se vzájemně vymění postupy pro šifrování a dešifrování. Ukažme si její princip na příkladu:

```
OTEVŘENÝ TEXT:  VIMZE NICNE VIM
POMOCNÝ TEXT:
V     E     N     M
I     N     E
M     I     V
Z     C     I
ŠIFROVÝ TEXT:   VENMI NEMIV ZCI
```

Pokuste se v programovacím prostředí Scratch realizovat uvedenou šifru. Program musí umět převést zobrazit zadané znaky mezinárodní šifrové abecedy jako kostýmy postav v prostředí Scratch a následně je přeskupit, a tak zašifrovat. Vzhledem k nutnosti realizace písmen textu prostřednictvím postav stačí, když bude program schopen zašifrovat text s délkou do 24 písmen.

Funkčnost programu můžete ověřit tím, že zadáte příkladovou větu do systému a výsledky porovnáte.

6. ÚLOHA – ŠIFRA LEONARDA DA VINCI

Itálie. Země, kde v době, kdy v jiných evropských zemích stále trvá středověk, již vzniká renesance. Ta dala vzniknout pojmu renesanční člověk. Když tento pojem vyslovíme, asi každému z nás vyvstane na mysl jméno Leonardo Da Vinci – malíře, architekta, sochaře, hudebníka, přírodovědce, konstruktéra, vynálezce, spisovatele a autora šifry v jedné osobě. Zarazila vás poslední informace? A myslíte, že přeháníme, nebo ne? Pravdou je, že nejde tak úplně o šifru v pravém významu toho slova. Nemáme doklady o tom, že by kdy byla využita za nějakým konkrétním politickým, vojenským, či jiným účelem. Přesto v jeho dochovaných spisech čas od času můžeme narazit na text, který na první pohled nejsme schopni přečíst. Druhý pohled nám však odkrývá poměrně triviální, avšak nesmírně rafinovanou a zajímavou šifru, která v sobě kombinuje prvky substituce a transpozice.

Pokuste se nejprve dešifrovat následující text bez toho, aniž byste byli dopředu seznámeni se způsobem řešení: ⅃ ƎONИAЯDO

Pokud jste text dešifrovali: „Leonardo“, výtečně! Z hlediska transpozice jde o to, že písmena otevřeného textu zde uspořádáme za sebou zprava doleva. Z hlediska substituce zde dochází k zrcadlovému obrácení jednotlivých znaků. Řada soudobých svědectví dokládá, že Leonardo psal převážně levou rukou. Proti zavedené zvyklosti, která vede leváky k tomu, aby psali stejně jako praváci zleva doprava, stojí fakt, že nejsnadnější způsob pro leváky je psát text zprava doleva. Ten vychází z přirozeného pohybu ruky „od středu těla ven“, tedy tak, jak píší praváci. Lze se tedy domnívat, že mistrovi nešlo v první řadě o šifru, ale že si chtěl především usnadnit zápis svých tisícistránkových poznámek.

Abychom se ujistili, že principu šifry rozumíme, ukážeme si její funkčnost na jednoduchém příkladu:

OTEVŘENÝ TEXT: MONA LISA
ŠIFROVÝ TEXT: A⅁I⅃ AИOM

Pokuste se v programovacím prostředí Scratch realizovat uvedenou šifru. Program musí umět převést zobrazit zadané znaky mezinárodní šifrové abecedy, zrcadlově převrácené, jako kostýmy postav v prostředí Scratch a následně je přeskupit, a tak zašifrovat. Vzhledem k nutnosti realizace písmen textu prostřednictvím postav stačí, když bude program schopen zašifrovat text s délkou do 10 písmen.

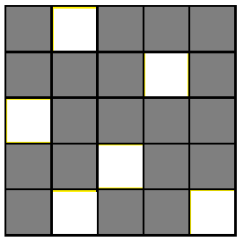
Funkčnost programu můžete ověřit tím, že zadáte příkladovou větu do systému a výsledky porovnáte.

7. ÚLOHA – FLEISSNEROVA MŘÍŽKA

Šifrovací, nebo též transpoziční mřížka. Šifrovací pomůcka, kterou zpopularizoval slavný spisovatel Jules Verne v jednom z mnoha svých dobrodružných románů. Romantickou představu, kdy mřížka slouží čtenáři k napínavému luštění prapodivných záhad, kalí skutečnost, že obdobná mřížka sloužila k vojenské šifrované komunikaci v jednom z nejstrašnějších světových konfliktů v dějinách lidstva – první světové války. Jde o tzv. Fleisserovu mřížku, která pro nás představuje poslední a zároveň nejobtížnější úlohu.

Fleissnerovu mřížku si můžeme představit jako papírovou kartu o čtvercových polích, s tím, že některé z těchto polí vystřihneme a vytvoříme tak průhledná okénka. (viz. obrázek) Poté přiložíme mřížku na papír a do „okének“ začneme postupně psát otevřený text, od shora dolů. Poté, co „okénka“ zaplníme, pootočíme mřížku pravotočivě o 90°. Tento postup celkem
3x zopakujeme. Mřížku poté odejmeme a měl by nám vyjít sled náhodných znaků.

Tabulka č. 4 Fleissnerova mřížka

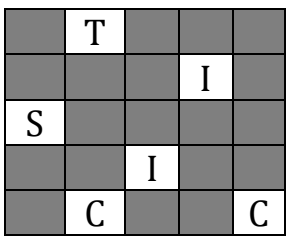


Abychom se ujistili, že principu šifry rozumíme, ukážeme si její funkčnost na jednoduchém příkladu:

OTEVŘENÝ TEXT: TISÍC CEST VEDE K JEDNOMU CÍLI

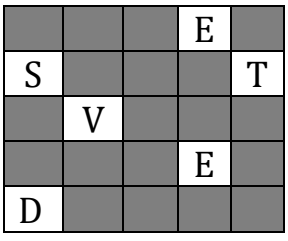
Tabulka č. 5 Fleissnerova mřížka v první poloze

KROK 1:



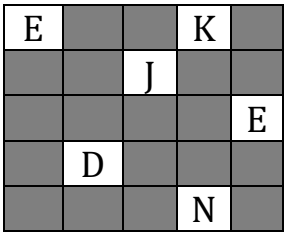
Tabulka č. 6 Fleissnerova mřížka otočená o 90°

KROK 2:



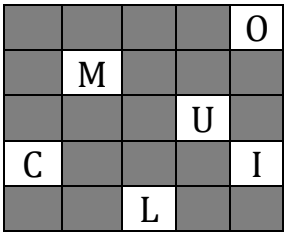
Tabulka č. 7 Fleissnerova mřížka otočená o 180°

KROK 3:



Tabulka č. 8 Fleissnerova mřížka otočená o 270°

KROK 4:



Tabulka č. 9 Rozmístění písmen po odstranění Fleissnerovy mřížky

E	T	E	K	O
S	M	J	I	T
S	V		U	E
C	D	I	E	I
D	C	L	N	C

Po odstranění Fleissenrovy mřížky zůstalo volné prostřední pole tabulky 5 krát 5 znaků, které můžeme využít pro doplnění posledního písmene zprávy, která má přesně 25 znaků. Výsledný šifrový text pak bude sestaven z pětic znaků odpovídajících jednotlivým řádkům tabulky.

ŠIFROVÝ TEXT: ETEKO SMJIT SVIUE CDIEI DCLNC

Pokud by šifrová zpráva měla méně než 25 znaků, doplnili bychom do volných polí nejprve písmeno X (jako ukončovací znak) a pak by následovalo několik náhodně vybraných písmen.

Pokuste se v programovacím prostředí Scratch realizovat uvedenou šifru. Program musí umět převést zobrazit zadané znaky mezinárodní šifrové abecedy jako kostýmy postav v prostředí Scratch a následně je přeskupit, a tak zašifrovat. Vzhledem k nutnosti realizace písmen textu prostřednictvím postav stačí, když bude program schopen zašifrovat text s délkou do 23 písmen. K dispozici máte předpřipravenou mřížku.

Funkčnost programu můžete ověřit tím, že zadáte příkladovou větu do systému a výsledky porovnáte.

BONUSOVÝ ÚKOL:

Vytvořte vlastní mřížku (tzn. s jiným rozložením „okének“) a zašifrujte libovolný text.

B. Dotazník

DOTAZNÍK:

1. Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?

2. Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?

3. V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?

4. V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?

5. Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?

6. Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?

7. Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?

C. Odpovědi respondentů na otázky dotazníku

DOTAZNÍK M6332:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Jsem přesvědčen, že historické šifrování může být zajímavým tématem pro výuku algoritmizace a programování, protože se jedná o úlohy dostatečně složité a náročné na čas a pečlivost provádění, aby se žákům či studentům vyplatilo přemýšlet o tom, jak přenechat rutinní provádění šifrovacího algoritmu počítači, ale současně také o úlohy přiměřeně snadné na to, aby žáci byli schopni algoritmus v lepším případě samostatně převést do konkrétního programovacího jazyka, v horším případě alespoň pochopit fungování hotového vzorového řešení.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Programovací jazyk Scratch je pro výuku základů programování v rámci předmětu Informatika na základní školy, nebo na střední škole netechnického zaměření, vynikající volbou. Zároveň to je v principu univerzální vyšší programovací jazyk, pochopitelně s omezeními společnými všem interpretovaným skriptovacím jazykům, jako jsou rychlost překladu (interpretace příkazů), nebo používání netypových proměnných. Pro žáky, kteří se nechtějí v budoucnosti stát IT odborníky, jsou tyto rysy jazyka spíš výhodou. S použitím jazyka Scratch ve výuce mám osobní dlouhodobou pozitivní zkušenost.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Metodická příručka je dobře uspořádána. Po stručném seznámení s vývojovým prostředím Scratch jsou postupně podrobně popsány jednotlivé úlohy, nechybí historický kontext a vysvětlení postupů „papír a tužka“. Úlohy jsou vhodně seřazeny od jednodušších po složitější. Postupně jsou také zařazovány

náročnější a komplexnější programátorské postupy.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   Příručka pro učitele je pro přehlednost psána na úroveň průměrného žáka. Nástroje pro individuální práci se slabšími žáky, jako jsou návodné otázky, pomocné úlohy apod. si musí čtenář – učitel informatiky doplnit sám. Možná ale tyto otázky zodpoví diplomová práce, ke které se příručka, úlohy i tento dotazník váží.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**
   Úlohy jsou zajímavé a pečlivě jsou zpracována i vzorová řešení. Proto bych se snažil v nějaké podobě využít všechny úlohy. Žákům bych asi zadal k samostatné práce ty jednodušší, což jsou podle mého názoru úlohy na substituční šifry. Složitější, tedy transpoziční šifry, bych zařadil do výuky např. v rámci soutěživých her, spojených s pobytem v přírodě v okolí školy, kdy žáci objeví tajemný dopis a s pomocí počítače a vzorového řešení např. šifrování Fleissnerovou mřížkou dopis šifrují, nebo dešifrují. Originální je šifra Leonarda da Vinci, která navíc ukazuje, jak zdánlivý hendikep (v tomto případě leváctví) se může změnit i ve výhodu.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**
   Složitější šifry se hodí spíše do kroužku programování než do běžné výuky.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**
   Vzorová řešení lze využít k šifrování a dešifrování zpráv pro různé soutěže nejen ve škole, ale např. ve skautských oddílech během roku i na prázdninovém táboře.

DOTAZNÍK M8015:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Domnívám se, že je velmi užitečné zařadit historické šifrování do výuky informatiky. Žákům je téma šifrování velmi blízké, což dokládá i to, že se žáci občas rádi baví tím, že si posílají zašifrované texty mezi sebou. A zapojení historických příběhů a souvislostí se mi vždy osvědčuje při výuce informatiky i matematiky, protože dobře fungují jako aktivizační prvek.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Jazyk Scratch jsem v minulosti při výuce využil a velmi se mi osvědčil jako „startovací" programovací jazyk. Jedná o krásně zpracovaný vizuální programovací jazyk, který pomáhá překonat odpor, kteří mají někteří žáci k syntaxi textových programovacích jazyků.
   Žáci mají na základní škole často pocit, že se učí o věcech, které jim v životě k ničemu nebudou, které jsou uměle vytvořené jen k tomu, aby je trápily a přinášely jim špatné známky. A textové programovací jazyky by pro ně mohly být další nestravitelnou počítačovou šifrou. Takže využití názornosti a hravosti při programování ve Scratchi je velmi užitečnou formou, jak si hned na začátku programování nezprotivit, ale naopak zamilovat.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Příručka je velmi dobře a přehledně zpracovaná a je napsána poutavou formou. Nejvíce se mi líbil historický příběh každé šifry a domnívám se, že právě historie zařazených šifer by mohla být dobrou motivací pro žáky, aby se šifru pokusili vytvořit v programu Scratch. A více zvídaví žáci se pravděpodobně následně pokusí vytvořit v tomto programu šifru vlastní.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**

   Nezbytným předpokladem k přečtení této příručky je předchozí alespoň základní znalost programu Scratch, protože při popisu programu je příručka velmi stručná. Tato příručka tedy nenahrazuje komplexní návod programu, což autor ve 2. kapitole zmiňuje.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Zatím nevím, které úlohy bych vybral, zaujaly mě v podstatě všechny. Ale vzhledem k časové náročnosti plnění školních vzdělávacích plánů bych vybral jednu substituční a jednu transpoziční šifru.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Domnívám se, že by šla například využít i při výuce matematiky, kdy by se rozvíjela logika a vytvářela šifry ručně bez využití programu. Využila by se především historická část této metodické příručky. A nějaký úkol by mohl být zadán pomocí šifry a žáci by měli odhalit o jakou šifru se jedná.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   V tuto chvíli mě bohužel další využití nenapadá.

DOTAZNÍK M8510:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Šifrování je určitě zajímavé téma, které může žáky na základní škole bavit. S příklady pro programování je ale problém, protože některé žáky nemusí nadchnout skoro nic. Ale zašifrovat zprávu pomocí programu ve skupinkách a pak dané zprávy vyhodnocovat může být zajímavá aktivita pro žáky při hodinách.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Vyučuji na střední škole, dětskými programovacími jazyky se nezabývám, žáky na střední škole by to již tolik nebavilo. Upřednostňuji programovací jazyk C#.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Zajímavé příklady i problematika, která je velice pěkně vysvětlená. Některé žáky to může zaujmout.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   Uvedl bych, jestli je daná problematika určená pro práci např. samostatně, ve skupinkách. I když je řešení velice pěkně popsáno, nevím, zdali si žáci poradí s jeho zadáním v programu pomocí grafických bloků. Jestli by jim k tomu třeba nepomohlo nějaké schéma apod. Možná jim dát i nějaký návod, i když sestavení může být různé, ne vždy jedna cesta v programování vede k cíli.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**
   Pro šikovné žáky klidně všechny, pro ostatní bych zvolil asi první tři, tj. ty snazší. Ale každý žák může postupovat individuálně, případně ve skupince žáků s podobnými znalostmi a schopnostmi.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**
   Šifrování je u nás ve škole zařazeno v předmětu Číslicová technika, která se vyučuje na elektro oborech i ve strojírenství.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**
   Danou problematikou se úplně nezabývám, takže nejsem schopný odpovědět.

DOTAZNÍK 8609:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Moc se mi líbila motivační část příručky. Ač jsem byl původně skeptický, vaše řeč mě opravdu motivovala, takže splnila svůj účel. Při čtení jsem si vzpomněl na dva své žáky 8. ročníku ZŠ, kteří jsou oba zapálení do historie, obzvláště středověkých a novověkých konfliktů a bitev. Oba mají také velký zájem o informatiku, takže vím, že taková kombinace se mezi žáky opravdu objevuje. Historické šifrování jako téma programování by jistě uvítali. Bohužel jsou tito dva výjimeční a jsem stále skeptický k tomu, že by téma historického šifrování motivovalo žáky k programování, obzvláště na základní škole.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Na naší škole programujeme ve Scratchi již několik let. Žáci 5., 6. a 7. ročníku programují ještě v jazyku Kodu. Obě prostředí jsou pro žáky poutavé. Scratch však upřednostňuji, protože tam už je opravdu o programování, zatímco Kodu je spíše takový seznamovací program.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Mě osobně seznámil s dalšími nástroji Scratche, které jsem neznal. Například mě doteď nenapadlo použít jako proměnnou text. Tomu, kdo Scratch alespoň trochu ovládá, rozšíří příručka obzory, jako je rozšířila mně.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   Příručka dle mého skromného názoru není vhodná pro učitele začínající s programováním, jak by se podle prvních stránek mohlo zdát. Ve druhé a třetí kapitole sice seznamuje učitele s prostředím, ale toto seznámení je velice stručné, někdy si vystačí s „postačí se seznámit s…“, což, pokud to není dáno do celkového

kontextu, učitelům sice pomůže s řešením úlohy, nikoliv však s pochopením programovacího jazyka. Pokud je tedy příručka pro učitele-nováčky, seznámení s prostředím není rozhodně dostačující a nevyhovují rozhodně ani úlohy, které jsou pro začátek až moc kombinované. Pokud je tedy příručka tvořena pro pokročilé učitele a žáky, nevěnoval bych pozornost základům, ale prostor ve druhé a třetí kapitole bych věnoval podrobněji blokům a akcím, které se v šifrování vašich úloh často objevují.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Nemám zkušenosti s učením na střední škole nebo základní škole se zaměřením na informatiku, ale na obyčejné základní škole jsou pro informatiku nízké hodinové dotace. Pro programování připadá několik hodin, kde se žáci seznamují s opravdovými základy, nebo se programování řeší formou kroužků či volitelných předmětů. Dovedu si představit, že kdyby měli žáci předmět nebo kroužek programování několikátým rokem, zadané úlohy by zvládli. Jak správně uvádíte, nejspíše pouze první tři úlohy. Na naší škole, kde mají žáci programování v jednom školním roce hodinu týdně, se během 32 výukových hodin nedá k takto kombinované úloze dostat. Žáci ZŠ navíc mají rádi Scratch kvůli jeho hravosti, postavičkám, animacím. Takový typ úlohy by je v prvním roce patrně odradil.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Jak jste sám zmínil, historický exkurz do světa šifer se dá využít v hodinách dějepisu. Nejsem sice vyučujícím dějepisu, ale tipoval bych, že některé šifry by byly pro některé vyučující novou informací a možná i zpestřením hodiny dějepisu.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Jistě mě napadá využití již hotového programu pro ukázky historických šifer v předmětu dějepis. V takovém případě by musel být program ovšem graficky lépe zpracován.

DOTAZNÍK M8706:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Myslím, že je to vhodné téma k motivaci ve výuce programování, hned z několika důvodů: vysoká míra algoritmizace při řešení úloh, různé způsoby šifrování dat, práce s daty, různé způsoby řešení úloh.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Ano, pro žáky základních škol je to vhodný programovací jazyk, který je snadno dostupný. Vývojové prostředí je přehledné a usnadňuje žákům práci při tvorbě programů. Blokové programování je rozšířené u edukačních robotů, kteří se používají na základních školách. Žáci tak snadno mohou přecházet ze jednoho prostředí do druhého.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Samotné téma šifrování dat s historickým exkurzem je hezké motivační téma, které má přesah do dalších předmětů. Možnosti využití vnímám u aktivních a nadaných žáků, dále v projektové výuce.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   Pokud žáci v programovacím jazyku Scratch dosud nepracují, bude pro ně začátek dosti náročný. Doporučil bych nejprve více jednodušších úloh pro seznámení s jazykem a pochopení základních principů. Dále jako velký problém vidím nedostatek času ve výuce informatiky na ZŠ pro použití této příručky a samotné výuky šifrování.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

Jedná se o sadu úloh, které jsou velmi zajímavé. Při použití ve výuce, bych úlohy vybíral podle časových možností a podle reakcí žáků. Začal bych postupně, jak jsou uvedené a následně bych případně přidával další úlohy. Potřeboval bych úlohy vyzkoušet přímo ve výuce.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Volnočasové kroužky pro žáky se zaměřením na informatiku, letní tábory, turistické oddíly, únikové hry.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Stejné, jako u předchozí otázky.

DOTAZNÍK M8908:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Zařazení šifrování do výuky programování je velmi závislé na celkovém konceptu výuky a do jisté míry také na vztahu vyučujícího k danému tématu. Vzhledem k charakteru úloh musí být vyučující schopen bezpečně žákům vysvětlit způsob šifrování, neboť i samotné šifrování „klasickou" formou může mnoha žákům činit potíže. Stejně tak musí být vyučující připraven reagovat na problémy, které může programování šifer žákům činit a na tyto problémy reagovat například návodnými úlohami, čehož nebude schopen bez dobré znalosti problematiky. Má-li vyučující tyto předpoklady, poté se mi jeví téma šifrování jako velmi vhodné nejen z hlediska motivace, ale i z hlediska rozsahu použitých příkazů daného programovacího jazyka.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   S volbou programovacího jazyka Scratch souhlasím a sám s ním mám velmi dobré zkušenosti při výuce programování na základní škole.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Metodickou příručku hodnotím kladně z hlediska její struktury. Učitel je zde totiž zpočátku seznámen s programovacím jazykem Scratch a domnívám se, že i učitel, který se s tímto jazykem nikdy nesetkal, se na základě uvedených informací dokáže velmi rychle ve Scratchi zorientovat. Kladně také hodnotím popis a řazení samotných úloh – od nejjednodušších po složitější. Od jednoduchých substitučních šifer přechází autor k šifře Leonarda da Vinci, kde se propojuje práce s textovými řetězci a grafikou, a poté přechází k nejsložitějším transpozičním šifrám.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**

Příručka by mohla obsahovat i stručný komentář autorských řešení úloh, která jsou součástí příručky. Autor by se také mohl zamyslet nad tím, co by žákům mohlo při řešení činit problémy a v rámci metodické příručky navrhnout způsob, jak těmto komplikacím předcházet (například formou návodných otázek nebo úloh…).

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Do výuky bych zařadil úlohy založené na principu substitučních šifer. Na těchto úlohách lze s žáky velmi snadno procvičit práci s cykly a práci s textovými řetězci. Jak jsem již zmiňoval, zařadil bych šifru Leonarda da Vinci, protože by byla pro žáky snadno pochopitelná a zároveň by se pomocí ní naučili propojovat práci s textovými řetězci a počítačovou grafikou.

   Transpoziční šifry Skytalé a Fleissnerova mřížka bych použil spíše jako úlohy pro nadanější žáky, protože jejich naprogramování (délka šifrovaného textu, práce s kostýmy a pohyb samotných postav/písmen…) je pro mnohé žáky velmi obtížné.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   S ohledem na již zmíněnou obtížnost transpozičních šifer se mi jeví její použití vhodné také pro kroužky programování nebo volitelnou výuku informatiky.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Hotové řešení úloh, na kterých může učitel ukázat princip šifrování, nikoliv samotný program, by se jistě mohlo hodit ve škole v rámci projektového dne nebo naopak v rámci různých volnočasových aktivit, kde by pro žáky některá z šifer mohla být nachystána.

DOTAZNÍK M9303:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**

   Zařadit šifrování do úvodu výuky programování je, dle mého názoru, vhodné. Žáci si mohou mezi sebou předávat zprávy a následně je dešifrovat. Rozvíjí se zde logika a originální řešení problémů. Žáci mohou v rámci hodiny vymyslet také svůj vlastní způsob šifrování.

   Velkou výhodou je mezipředmětový vztah s dějepisem a také českým jazykem. Obecně toto téma nabízí velké možnosti v rámci mezipředmětových vztahů, protože ho lze využít i v anglickém jazyce – žáci mohou sestavovat šifry v angličtině, a tak si protrénovat cizí jazyk. Výsledky šifry poté mohou i vyhláskovat.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**

   Jednoznačně souhlasím. Scratch se formou jednoduchého kódování hodí již na první stupeň. Žáci se snadno a přehledně naučí základy a vytvoří si vhodné způsoby algoritmického přemýšlení. Tyto základy se jim budou hodit při případné volbě dalšího programovacího jazyka.

   V oblasti dětského programování samozřejmě existují další programovací jazyky, které lze využít (Baltík, Robot Karel ad.), ale Scratch mi osobně vyhovuje nejvíce. Má velkou „fanouškovskou" základnu a je tedy mnohem snazší najít různé materiály a metodiky. Scratch je také zcela zdarma (na rozdíl například od Baltíka) a není tedy vůbec nákladné ho ve výuce používat.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**

   Příručka je napsaná přehledně a její největší výhodou je, že je „blbuvzdorná". Myšleno je to tak, že učitel nepotřebuje žádné zvláštní vědomosti, ani schopnosti, aby ji mohl v praxi využít. Velkou výhodou je, že autor již vytvořil vhodné

podklady i přímo do programu Scratch. Učitel tedy nemusí v prostředí vytvářet žádné algoritmy, pouze otevře soubor a může pracovat. Jak sem již zmínil, má také velká pozitiva v mezipředmětových vztazích.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**

   Přímo do příručky bych vložil odkaz, na kterém lze stáhnout .sb3 soubory s řešením jednotlivých šifer v prostředí Scratch. K jednotlivým úlohám bych přidal jejich přibližnou časovou náročnost.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Ve výuce lze snadno využít všechny z přiložených úloh. Vzhledem k mezipředmětovým vztahům (v tomto ohledu dějepis) bych pravděpodobně na prvních pozicích vybral šifry Caesara, Jana Husa a Leonarda da Vinci. Nápaditostí se mi nejvíce líbí úloha Skytalé. Šifry se mi líbí všechny, ale pokud bych musel nějakou vynechat, byla by to pravděpodobně jednu ze substitučních. Jejich princip je poměrně velmi podobný. Výhodou však je, že takové šifrování si klidně může po svém zkusit vytvořit i některý ze žáků.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Jak jsem již zmiňoval, dokážu si snadno představit využití těchto úloh i v hodinách dějepisu a anglického jazyka. Čtvrtá úloha by se pravděpodobně dala využít i v hodinách matematiky. Druhá úloha by se velmi dobře dala využít v hodině dějepisu (téma Julius Caesar, nebo obecně o Římu). Učitel by například mohl žákům předat nejznámější citáty od Caesara v zašifrované formě.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Popravdě mě v tuto chvíli žádné jiné vhodné využití ve školním prostředí nenapadá. Toto řešení vytvořené v prostředí Scratche se hodí právě na výuku programování. Možná pokud by někdo chtěl šifrovat nějakou zprávu, mohl by tyto

již vytvořené podklady snadno využít online a nemusel by zprávy měnit ručně po jednotlivých písmenech na papíru.

DOTAZNÍK M9401:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**

   Zařazení by záviselo na ŠVP a na tematickém plánu školy. Dle mých zkušeností, lze tuto problematiku zařadit, ovšem pouze s malou hodinovou dotací. Tuto problematiku bych zařadil do úvodní části studia Informačních technologií. V případě odborné školy zabývající se výpočetní technikou / matematikou by samozřejmě měl být uzpůsoben větší prostor.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**

   Z mého pohledu závisí stupni, kde je Scratch používán. Dle mého názoru ho lze použít na druhém i třetím stupni. Spíše bych se, ale klonil ke druhému stupni. U třetího stupně by záleželo na typu SŠ.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**

   Rozvíjení myšlení, hledání řešení, dobrá motivace – převedení myšlenkových pochodů do automatické podoby za pomocí techniky – provázanost se smyslem a vývojem Výpočetní technologie. Využití Scratche – jednoduché a dostupné prostředí.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**

   Ocenil bych pracovní listy či jiný materiál, který by posloužil hlavně žákům. Dále bych zvýšil kvalitu obrázků.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

Dokázal bych použít všechny šifry, zamýšlel bych se pouze nad jejich složitostí, či využitím. Například šifru Skytalé by byla dobrá na porovnání právě mezi šifrováním pomocí pomůcky a pomocí počítače, zde by bylo i dobře demonstrovatelná zásada algoritmů – univerzálnost.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Poznatky z této příručky lze poměrně dobře využít i v rámci akcí školy jako jsou různé výlety, adaptační kurzy apod. Aktivity na principu únikových her při teambuildingu. Dokážu si představit takovéto úkoly i během suplovaných či nestandartních vyuč. hodin. Kdy žáci soutěžní formou rozvíjejí své myšlení.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Bohužel mě momentálně nic nenapadá. Možná lze kódy využit pro inspiraci, ale ne v jiném odvětví než ve výuce informatiky. Asi pouze na soutěžích v šifrování apod.

DOTAZNÍK M9502:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**

   Jakákoliv forma motivace spočívající v tom ukazovat na konkrétních příkladech využití algoritmu / programu v praxi je velice vhodná a žáci si tak lépe dokáží představit k čemu je a kde se používá takové programování. Použití tématu historických šifer bude motivovat především ty žáky, které mají nějaký kladný vztah k historii, méně pak ty, které nemají.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**

   Scratch podle mě není nejvhodnějším programovacím jazykem pro výuku programování, jelikož se u něho žáci stejně musí naučit všechny skupiny bloků, k čemu se používají a kde je mají hledat ve webovém rozhraní, tak aby je mohli vložit do scénáře. Bylo by tedy vhodnější žaky rovnou učit např. v jazyku Python ve spojení s Google Colab (webové IDE), který má dostatečně jednoduchou syntaxi a ve spojení s Google Colab by mohla být dosažena i určitá forma vizualizace a interakce při použití notebooků, kde by se míchal kód společně s výukovým textem. Zároveň pak mohou tento jazyk využít v praxi nebo při dalším studiu na střední nebo vysoké škole.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**

   Díky této metodické příručce mohou učitelé snadno zapojit úlohy v ní obsažené do své výuky. Tyto úlohy, které jsou seřazené podle obtížnosti si pak učitel může zapojit do svých hodin, dle úrovni žáků anebo si je libovolně upravit díky již předpřipraveným šablonám (vzorovým řešením), např. přidat úkol na dešifrování zprávy. Mezi další výhody této metodiky patří to, že se žáci při řešení úloh nebudou učit pouze programovat, ale dozví se i něco o historii a základech šifrování.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**

Jak už jsem zmiňoval výše, jako hlavní nevýhodu této metodiky je zvolení programovacího jazyka Scratch, který žáci nemohou využít buďto v praktickém využití nebo v jejich dalším studiu a musí se pak učit zbytečně další jazyk, i když už by mohli získat základ jiného hojně používaného jazyka.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Do výuky informatiky bych zařadil všechny úlohy, jelikož se díky nim žáci seznámí se základními šiframi a zároveň si na nich zlepší svou dovednost algoritmizace, práce s polem a textem, která jim bude v případném dalším studiu přínosná. Množství takovýchto úloh se pak dá navýšit i např pro střední školy o další historické šifrovací či jiné algoritmy.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Žádné jiné využití této metodické příručky mě nenapadá.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Žádné jiné využití autorských vzorových řešení úloh mě nenapadá.

DOTAZNÍK Z9103:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Myslím, že cokoliv „tajného“ je pro děti zajímavé a většinu žáků bude zajímat, jakými způsoby mohou zašifrovat svá psaníčka. Historické šifrování by mohlo být příjemným zpestřením výuky informatiky.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Základní princip fungování jazyka Scratch žáci pochopí poměrně rychle, přijde mi vhodné zařadit tento jazyk pro výuku programování na základních školách.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   V příručce jsou přehledně popsány jednotlivé bloky a jejich fungování, tuto část bych využila pro seznámení žáků s prostředím Scratch. Kladně hodnotím také návaznost jednotlivých úkolů, zajímavosti z historie a tip u šifry Skytalé.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   V příručce jako taková nedostatky nevidím, pouze se obávám zařazení tohoto tématu do běžných hodin informatiky, pro některé žáky by programování těchto šifer mohlo být obtížné. Možná by bylo dobré uvést doporučený věk žáků.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**
   Vše by záleželo na schopnostech žáků, začala bych s jednoduššími šiframi, ale nemohu říct, že bych nějakou šifru vyloženě nezařadila.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

Část příručky o šifrách by šla určitě využít i v hodinách dějepisu nebo matematiky, toto téma je pro žáky zajímavé.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**
   Kromě běžných hodin informatiky nebo informatického kroužku mne jiný způsob využití nenapadá.

DOTAZNÍK Z9105:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Téma historického šifrování je vhodné zařadit do výuky. Dá se na něm ukázat mezipředmětové vztahy, propojení informatiky, dějepisu a matematiky.
   Historické šifrování je zajímavé téma a mohlo by v žácích probudit zájem o vymyšlení a naprogramování si vlastní šifry.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Programovací jazyk Scratch je vhodný pro výuku programování. Používám tento programovací jazyk od primy po oktávu. Dají se v něm vytvořit programy různé obtížnosti. Scratch je žákům bližší svým prostředím na rozdíl od jiných programovacích jazyků. Ve Scratch si žáci mohou naprogramovat svoji vlastní jednoduchou počítačovou hru a to je myslím, pro ně ta hlavní motivace pro tento jazyk.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Klady této příručky vidím v propojení více předmětů. Žáci si mohou šifry vyzkoušet vytvářet při dějepise na papír a poté v rámci hodin informatiky si šifry vytvořit v programovacím jazyce Scratch. Dále za kladné považuji zvolení programovacího jazyka, který je pro žáky velmi intuitivní.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   I když by žáci byli seznámeni předem s programovacím jazykem, ne všichni žáci by byli schopni vytvořit nebo pochopit vytváření dané šifry v programovacím jazyce.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Na začátek bych určitě zařadila Caesarovu šifru a šifru Jana Husa, které je jednoduchá a žáci je snadno pochopí. Zajímavá je i šifra Leonarda Da Vinci Fleissnerova mřížku bych použila spíše pro žáky vyšších ročníků a pro zdatnější žáky. Zdá se mi pro žáky hůře pochopitelná a složitější na rozdíl od ostatních šifer.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Příručka by se také dala využít na kroužcích informatiky nebo také na seminářích z informatiky pro žáky. Těch se účastní více žáků, kteří se chtějí informatice věnovat. Dále by se část příručky dala využít v hodinách dějepisu. V příručce jsou hezky popsány jednotlivé šifry a jejich použití při šifrování.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Vzorové řešení by se dala také použít výuce v dějepisu, kdy se žáci s těmito šiframi seznámí. A učitel dějepisu může ukázat, jak funguje zašifrování textu jednotlivými šiframi.

DOTAZNÍK Z9302:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**

   Téma šifrování je podle mě obecně důležité, aby bylo zařazeno do výuky informatiky, a to především proto, aby žáci pochopili princip komunikace v počítači, kdy jsou jednotlivé informace převedeny na 1 a 0. Toto si představit je pro některé žáky velmi složité. Spojit tedy toto téma s tématem šifrování může žákům pomoci pochopit daný princip. Navíc, pokud jsou vybrány takové šifry, které jsou spojovány s některou pro žáky známou historickou postavou nebo událostí, stává se toto téma pro ně zajímavější. Zaujmout žáky bývá totiž jedním z hlavních problémů současné pedagogiky.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**

   S volbou zařazení programovacího jazyka Scratch pro výuku programování v hodinách informatiky na základních školách rozhodně souhlasím. Především díky grafické nápaditosti, která žáky na první pohled zaujme. I přes to, že se učí novým věcem a rozvíjí logické myšlení, mají pocit, že si hrají. Pro žáky na základních školách je výuka programování ve vizuálních programovacích jazycích snazší, než v textových. Z mého pohledu by ale zařazení programovacího jazyka Scratch mělo předcházet využití nějaké programovací hry k výuce základních algoritmů a pochopení principu algoritmizace. Když mají tyto základy žáci osvojené, je pro ně jednodušší v jazyce Scratch jednotlivé programy sestavovat.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**

   I když jako učitel informatiky jsem principy většiny využitých šifer znala, tak pro mě byl novinkou historický kontext některých z nich. Myslím si, že by podobný názor měli i jiní učitelé. Dané šifry jsou známé, ale příběh, který se k nim pojí už ne tolik. Přitom příběh je právě to, čím žáky zaujmeme, vtáhneme je do probíraného tématu. Navíc příručka nabízí právě to, k čemu se snaží směřovat

informatika v posledních letech. I přes to, že je stále důležitá teorie, klade se velký důraz na výuku programování. Tato příručka propojuje obě tyto linie, čímž splňuje cíle současné výuky informatiky.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**

   Obsah metodické příručky spoléhá na to, že učitel informatiky dokonale ovládá základy programování. Bohužel i v této době je plno učitelů informatiky, především starší generace, kteří programování dosud neučili. Nyní je však trendy ve výuce k tomu "nutí" a některým z nich to přináší značné problémy. Myslím si, že tito učitelé by uvítali, kdyby byly jednotlivé příklady v příručce více rozebrány, včetně řešení jednotlivých příkladů, využití některých bloků, apod.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Do výuky bych pravděpodobně nezařadila poslední (Fleissnerovu) šifru. Rozhodně ne jako samostatnou práci pro žáky. Princip vzniku je totiž mírně odlišný od ostatních třech popisovaných transpozičních šifer. Je potřeba využití mřížky jako další postavy, a tedy i vzniku nových bloků, jejichž vznik a použití by většině žáků dělalo problémy. Rozhodně by se ale touto šifrou dalo celé téma uzavřít. A to tím způsobem, že by žáci daný program sestavili nejdříve společně s učitelem a poté by se dal využít bonusový úkol z příručky jako samostatná práce žáků na závěr.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Jak je v příručce popsáno, tak pojetí výuky nejen na základních školách čím dál tím více klade důraz na využití mezipředmětových vztahů. Ve školách se již častěji setkáváme s projekty, kdy učitelé různých předmětů v tom samém období využívají jedno a to samé téma. Pokud dáme stranou tyto mezipředmětové vztahy, dala by se příručka použít ve výuce takových předmětů, které jsou úzce spojené s informatikou. Například počítačová grafika nebo výtvarná výchova, jejíž výuka se v této době také často přesouvá na počítače. V rámci výuky těchto předmětů se dá žákům ukázat vykreslování objektů, animace a pohyb předmětů. Čímž se ale opět

dostáváme k mezipředmětovým vztahům. Bez znalostí z hodin informatiky a dějepisu by se žáci při tvorbě programu neobešli.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Odpověď na tuto otázku zůstává stejná jako na tu předchozí. Nejčastěji se na základních školách úlohy tohoto typu budou využívat ve výuce předmětů informatika, programování nebo jiných podobných předmětech, které jsou úzce spjaté právě s informatikou, algoritmizací a programováním.

DOTAZNÍK Z9402:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Podle mého názoru je vhodné zařadit téma historické šifrování do výuky informatiky. Žáci se dozvědí, jak to fungovalo v historii a na základě toho se mohou dostat až do šifrování dnešní doby.
   Jak bylo řečeno v příručce, nejenže tímto tématem spojíme dva předměty, ale zároveň můžeme posílit zájem žáků, kteří dosud projevovali pasivní přístup ať už u jednoho či druhého předmětu.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Scratch, jako programovací jazyk pro žáky druhého stupně, mi přijde ideální. Z důvodu vizualizace mohou žáci lépe porozumět programování. Myslím, že díky vizualizaci se mohou lépe orientovat v krocích, které programují, a i metodou pokus omyl vyřešit zadané úkoly.
   Pokud by však bylo zařazeno programování ve všech ročnících druhého stupně ZŠ, postupně bych opouštěla Scratch a pokusila se tento program nahradit složitější variantou.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   V metodické příručce je vše přehledně vysvětleno. Myslím, že pokud zrovna učitel informatiky nemá další aprobaci historii, i tak by se od tohoto mohl inspirovat pro svoji výuku a zařadit to do svého plánu.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   Neshledávám žádné nedostatky.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   Všechny úlohy mi přijdou srozumitelné a pochopitelné. Co bych zařadila, by záleželo zrovna na tom, co by žáci probírali v historii. Složitější úlohy bych zařadila až ve vyšších ročnících.

   Myslím, že by bylo vhodné úplně na úvod zařadit i zpětné šifrování. Kde například jedna půlka třídy by měla za úkol zašifrovat nějakou zprávu a druhá půlka třídy zprávu rozšifrovat.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Metodická příručka, vysvětlující práci v programovacím jazyce Scratch se určitě dá použít i v dalších předmětech.

   Žáci v nižších stupních to mohou případně využít v předmětu výtvarná výchova, kde si mohou vytvořit jednotlivé příkazy a pokyny, které například pak mohou využít v předmětu tělesná výchova, kde na základě jednoduchých pokynů budou imitovat vizualizaci programovacího jazyka.

   Dále by se tato příručka dala použít i v cizím jazyce, nejlépe anglickém. Pokud by se žáci naučili jednotlivé příkazy přeložit do angličtiny, měli by jistě dobrý základ pro další, už složitější programovací jazyk.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   V tomto směru mě napadají jen předměty zaměřené na historii.

DOTAZNÍK Z9501:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Je to velice zajímavé odvětví informatiky, u kterého lze propojit mezipředmětově informatiku s dějepisem. Motivací k šifrování lze studenty inspirovat ke studiu programování, aby byli schopní šifru naprogramovat.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   Ano, souhlasím. Scratch je skvělý programovací jazyk, který i starším studentům na středních školách názorně ukáže základy programování. Nemusejí se nijak trápit zapamatováním si složitých zápisů jednotlivých algoritmů, vše již pouze sestavují k sobě a nastaví požadované parametry. Takže pro výuku základů programování naprosto ideální program pro všechny věkové kategorie.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Největší předností je samozřejmě možnost čerpat z již vytvořené sady úloh a správných řešení, které autor učitelům nabízí k dispozici. Učiteli tak ušetří poměrně dost času s vymýšlením úloh, a hlavně později s opravou těchto úloh.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   Možným problémem by mohlo být nedostatečné vysvětlení úvodu do algoritmizace a šifrování ze strany vyučujícího učitele. Dále také možnost nezájmu studentů, kterým není blízké ani jedno odvětví.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**
   Do výuky bych nejraději zařadila na úplný začátek úlohu č. 2 Caesarovu šifru, protože je to typická úvodní lehká úloha, která je rychlá nejenom na pochopení,

ale také na vytvoření, a přesto praktická a účinná. Dále bych do výuky zařadila úlohu č. 6 Šifra Leonarda da Vinci, protože si myslím, že je bude bavit dešifrovat šifru nejenom pomocí programu, ale také pouze svou vlastní hlavou.
Nezařadila bych například úlohu č. 5 Skytalé, která mi přijde velice složitá na sestavení.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**
   Myslím, že by bylo vhodné je zařadit v začátcích programátorského kroužku nebo robotického kroužku.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**
   Například je vložit do online kurzu, kde by uživatelé mohli studovat samostatně. Samozřejmě by k tomu zapotřebí vhodné prostředí a další podpůrný materiál, díky kterému by poté byl kurz vhodný pro samostudium a který by uživatele provázel po celou dobu.

DOTAZNÍK Z9502:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**
   Plně se ztotožňuji s tím, co je psáno v úvodu metodické příručky. Pro humanitně zaměřené studenty je určitě lepší, když si i v tomto předmětu, který jim třeba nic moc neříká, najdou něco, co je zajímá a kde si na vlastní kůži mohou vyzkoušet, jak takové šifrování funguje a fungovalo.
   Stejně tak „ajťáci" se takto dozví alespoň pár informací z historie. Já osobně jsem o některých šifrách neměla ani tušení a jsem ráda, že jsem si o nich mohla něco přečíst.

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**
   S volbou tohoto jazyka rozhodně souhlasím. Sama jsem měla tu možnost vyučovat Scratch úplné začátečníky. Obrovskou výhodou tohoto programovacího jazyka je způsob tvorby algoritmů. Pro začátečníky bývá často obtížné zapamatovat si přesné příkazy a dodržovat bezchybnou syntaxi při tvorbě programu. Proto je sestavování algoritmu pomocí skládání dílků (=příkazy v jazyce Sratch) tou nejlepší možností, jak se naučit strukturu algoritmu a jak všemu správně porozumět. Scratch je zároveň stále se vyvíjející platformou. Rozdíl ve funkčnosti verze 1.4, ve které jsem se učila já na střední škole, a ve verzi 3, ve které jsem pracovala se svými studenty byl znatelný.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**
   Líbí se mi, že u každé úlohy je i historický původ dané šifry. Věřím, že studenty tento úvod dokáže správně namotivovat a vtáhnout do problematiky. Zároveň je v příručce více úloh od podobné obtížnosti, takže mají studenti možnost, po prvotním výkladu od vyučujícího, zkoušet úlohy sami.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**
   Zcela upřímně mohu říct, že nedostatky jsem žádné nenašla. Samozřejmě nejlepším způsobem, jak zjistit nedostatky či rezervy, je vyzkoušet příručku v praxi. Bohužel tuto možnost nyní nemám. Po pročtení a vyzkoušení uvedených úloh jsem však nenašla nic, s čím bych byla nespokojená.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**
   První tři úlohy bych do výuky zařadila určitě. Jsou jednoduché na pochopení, a přesto dokáží při programování potrápit. Tyto úlohy jsou vhodné jak pro začátečníky z řad základní školy, tak i pro středoškolské studenty. Sedmá úloha o Fleissnerově mřížce může být zpočátku pro studenty trochu nepřehledná či složitá, ale po názorné ukázce na příkladu, je šifra určitě zaujme svým provedenímm které je odlišné od ostatních a také funkčností.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**
   Logické by bylo využít příručku v hodině dějepisu při výkladu o šifrách. Akorát programování bych zde zmínila pouze okrajově, jako možnost. Případně by bylo možné použít již naprogramované úlohy jako ukázku, jak šifry fungují. Žáci by tak mohli lépe pochopit princip šifer.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**
   Dle mého názoru by se úlohy daly využít i matematice, kde by napomohly rozvíjet u žáků logické myšlení, které je tolik potřebné a mnohdy podceňované. Poté samozřejmě v různých zájmových kroužcích, zaměřených na techniku.

DOTAZNÍK Z9603:

1. **Proč podle vás je nebo není vhodné zařadit téma historického šifrování jako motivace k výuce programování do výuky informatiky?**

   Záleží na provedení. Při propojení těchto dvou témat hrozí sklouznutí k "oslím můstkům" a k nelogickému propojování. Naopak využití historických souvislostí, důvodů, proč byla která technologie potřeba, může být ku prospěchu věci (příklad - Alan Turing a rozluštění Enigmy + 2. světová válka)

2. **Souhlasíte s volbou programovacího jazyka Scratch jako vhodného pro výuku programování, nebo byste upřednostnili jiný programovací jazyk? Pokud jiný, který by to měl být?**

   Nemám dobré zkušenosti s "obrazovými" programovacími jazyky. I když může být textový programovací jazyk složitější k proniknutí, jemné nuance a ošetření a pochopení toho, co se vlastně v programu děje podle obrazový programovací jazyk neobsáhne. Na druhou stranu, několik úvodních cvičení s rychlým přechodem k např. Jazyku Pascal by mohly pomoci ilustrovat činnost.

3. **V čem vidíte klady a potenciál této metodické příručky v rámci zařazení do výuky informatiky?**

   Propojení s historickými šiframi je zajímavé.

4. **V čem vidíte nedostatky a rezervy této metodické příručky v rámci zařazení do výuky informatiky?**

   Nevidím zde žádné zásadní nedostatky.

5. **Které úlohy, ze sedmi předložených, byste do výuky zařadili a které nezařadili a proč?**

   První tři úlohy jsou velmi podobné. Podle mě by stačilo vybrat jednu z nich nebo pracovat s jejich podobností. Úloha šifra Leonarda da Vinciho je podle mě poměrně složitá, protože nepracuje s textem jako s textem, ale zpracovává

jednotlivé symboly. Zařadila bych tuto úlohu jako volitelnou nebo jako rozšíření pro zájemce. Ostatní úlohy bych do výuky zařadila.

6. **Napadá vás nějaké jiné vhodné využití této metodické příručky než využití ve výuce informatiky?**

   Ne.

7. **Napadá vás nějaké jiné vhodné využití autorských vzorových řešení úloh než využití ve výuce informatiky?**

   Tzv. šifrovačky.