

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2019

Bc. Ivo Süß



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

KVANTITATIVNÍ ANALÝZA SCHÉMAT ZÁLOHOVÁNÍ DAT

QUANTITATIVE ANALYSIS OF DATA BACKUP SCHEMES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Ivo Süß

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Karel Burda, CSc.

BRNO 2019



Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**

Ústav telekomunikací

Student: Bc. Ivo Süß

ID: 130709

Ročník: 2

Akademický rok: 2018/19

NÁZEV TÉMATU:

Kvantitativní analýza schémat zálohování dat

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte a popište schémata používaná pro zálohování dat. Dále nastudujte a popište matematické modely těchto schémat. Na tomto základě vytvořte program pro kvantitativní analýzu schémat zálohování dat. S jeho pomocí zjistěte a analyzujte vlastnosti běžně používaných schémat pro různá zatížení. Na základě získaných výsledků zpracujte soubor zásad pro volbu optimálního schématu zálohování dat.

DOPORUČENÁ LITERATURA:

[1] Frisch E.: Handbook of Network and System Administration: System Backup: Methodologies, Algorithms and Efficiency Models. Elsevier, Amsterdam 2008.

[2] Burda K.: Matematický model zálohování a obnovy dat. Elektrověst, 2014, č.1. Dostupné na:
<https://bit.ly/2MuXrC9>

Termín zadání: 1.2.2019

Termín odevzdání: 16.5.2019

Vedoucí práce: doc. Ing. Karel Burda, CSc.

Konzultant:

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Cílem diplomové práce bylo vytvořit program pro kvantitativní analýzu schémat zálohování dat a s jeho pomocí zjistit a analyzovat vlastnosti běžně používaných schémat pro různá zatížení. Na základě získaných výsledků sestavit soubor zásad pro volbu optimálního schématu zálohování dat. Program byl vytvořen pomocí programového prostředí Matlab. Pomocí programu lze zjišťovat u jednotlivých zálohovacích schémat parametry: parametry C (celkový objem záloh) a E (střední objem obnovovacích záloh), velikost záloh jednotlivých dní, vytížení jednotlivých úložišť, cenu úložišť a celkového zálohovacího schématu, velikost celkově zapsaných dat na jednotlivá úložiště za časový úsek. V závěru práce je definováno schéma pro volbu optimálního zálohovacího schématu.

KLÍČOVÁ SLOVA

Děd-Otec-Syn, GFS, Hanojské věže, Inkrementální záloha, Kombinovaná záloha, Matlab, Rotace úložišť, Round robin, Rozdílová záloha, Úplná záloha

ABSTRACT

The aim of master thesis was to create a program for the quantitative analysis of data backup schemes and with its help to identify and analyze the properties of commonly used schemes for different loads. Based on the obtained results, compile a set of principles for choosing the optimal data backup scheme. The program was created by Matlab. It can be used to find out parameters for individual backup schemes: Parameters C (total backup volume) and E (medium volume of recovery backups), backups size of individual days, workload of individual storages, cost of storages and cost of overall backup scheme, size the total amount of data written per storage per time slot. At the end of the thesis is defined a scheme for choosing the optimal backup scheme.

KEYWORDS

Grandfather-Father-Son, GFS, Hanoi tower, Inkremental backup, Combined backup, Matlab, Rotation of storages, Round robin, Diferential backup, Full backup

SÜSS, Ivo. *Kvantitativní analýza schémat zálohování dat*. Brno, 2019, 65 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: doc. Ing. Karel Burda, CSc.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Kvantitativní analýza schémat zálohování dat“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu doc.Ing. Karlu Burdovi, CSc. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

podpis autora

Obsah

1	Úvod	11
2	Zálohování	12
2.1	Úplná záloha	12
2.2	Inkrementální záloha	12
2.3	Rozdílová záloha	13
2.4	Kombinovaná záloha	13
2.5	Porovnání typů záloh	14
3	Matematický popis záloh	15
3.1	Kombinovaná záloha	15
3.2	Úplná záloha	17
3.3	Inkrementální záloha	17
3.4	Rozdílová záloha	18
3.5	Matematické porovnání záloh	19
3.5.1	Výpočet úplné zálohy	19
3.5.2	Výpočet inkrementální zálohy	19
3.5.3	Výpočet rozdílové zálohy	20
3.5.4	Výpočet kombinované zálohy	20
3.5.5	Porovnání výsledků	20
4	Strategie	21
4.1	Round Robin	21
4.2	Rotace 6 úložišť	21
4.3	GFS	22
4.4	Hanojské věže	23
5	Matematický popis strategií	24
5.1	Rotace Round Robin	24
5.2	Rotace 6 úložišť	24
5.2.1	Výpočet pro Inkrementální verzi	24
5.2.2	Výpočet pro Rozdílovou verzi	24
5.3	GFS	24
5.3.1	Výpočet pro Inkrementální verzi	25
5.3.2	Výpočet pro Rozdílovou verzi	25
5.4	Hanojské věže	25
5.4.1	Výpočet pro Úplnou verzi	25
5.4.2	Výpočet pro Rozdílovou verzi	25

6	Zálohovací média	26
6.1	Optický disk	26
6.2	Magnetická páska	26
6.3	SSD disk	27
6.4	Pevný disk HDD	27
6.5	Diskové pole RAID	27
6.5.1	RAID 0	27
6.5.2	RAID 1	28
6.5.3	RAID 2	29
6.5.4	RAID 3	29
6.5.5	RAID 4	30
6.5.6	RAID 5	30
6.5.7	RAID 6	31
6.5.8	RAID 10	31
6.5.9	RAID 50	32
6.5.10	RAID 60	32
7	Program	33
7.1	Uživatelská část	33
7.2	Programová část	34
7.3	Funkce textovy_vypis_nejlevnejsi_rot_x	37
7.4	Funkce textovy_vypis_nejlevnejsi_gfs	38
7.5	Funkce textovy_vypis_nejlevnejsi_hanoi	38
7.6	Funkce hanoi	39
7.7	Funkce rot_x_db	41
7.8	Funkce rot_x_ib a rot_x_fb	44
7.9	Funkce rot_x_kombinovana	44
7.10	Funkce gfs_db a gfs_ib	44
7.11	Funkce graf_rot_x	44
7.12	Funkce graf_gfs	50
7.13	Funkce graf_hanoi	50
8	Naměřená data	51
9	Volba optimálního schématu zálohy	57
10	Závěr	59
	Literatura	60
	Seznam symbolů, veličin a zkratk	62

Seznam příloh	63
A Ukázka výpisu z programu	64
B Obsah přiloženého CD	65

Seznam obrázků

2.1	Schéma úplné zálohy	12
2.2	Schéma inkrementální zálohy	13
2.3	Schéma rozdílové zálohy	13
2.4	Schéma kombinované zálohy	14
2.5	Porovnání typů záloh	14
4.1	Rotace Round Robin	21
4.2	Rotace 6 úložišť	22
4.3	Rotace Grandfather - Father - Son	22
4.4	Rotace Hanojské věže	23
6.1	Schéma zapojení diskového pole RAID 0	28
6.2	Schéma zapojení diskového pole RAID 1	28
6.3	Schéma zapojení diskového pole RAID 2	29
6.4	Schéma zapojení diskového pole RAID 3	29
6.5	Schéma zapojení diskového pole RAID 4	30
6.6	Schéma zapojení diskového pole RAID 5	30
6.7	Schéma zapojení diskového pole RAID 6	31
6.8	Schéma zapojení diskového pole RAID 10	31
6.9	Schéma zapojení diskového pole RAID 50	32
6.10	Schéma zapojení diskového pole RAID 60	32
7.1	Graf Rotace X počet kusů úložišť.	47
7.2	Graf Rotace X celková cena úložišť.	47
7.3	Graf Rotace X velikost denní zálohy úložišť.	48
7.4	Graf Rotace X celkově zapsaná data do úložišť.	48
7.5	Graf Rotace X vytížení úložišť.	49
7.6	Graf Rotace X týdenní vytížení úložišť.	49
8.1	Celkový objem záloh C pro GFS rotaci	51
8.2	Střední objem obnovovacích záloh E pro GFS rotaci	52
8.3	Celkový objem záloh C pro Rotaci X úložišť	52
8.4	Střední objem obnovovacích záloh E pro Rotaci X úložišť	53
8.5	Celkový objem záloh C pro rotaci Hanojské věže	53
8.6	Sřední objem obnovovacích záloh E pro rotaci Hanojské věže	54
9.1	Závislost velikost C a E na typu schématu zálohy, u Rotaci X úložišť, při zátěži do 50%.	57
9.2	Závislost velikosti C a E na typu schématu zálohy, u Rotaci X úložišť, při zátěži nad 50%.	57
9.3	Diagram výběru optimálního zálohovacího schématu	58
A.1	Ukázka výpisu programu	64

Seznam tabulek

3.1	Soupis objemů jednotlivých záloh pro KZ	16
3.2	Soupis objemů obnovovacích záloh pro KZ	16
3.3	Soupis objemů jednotlivých záloh pro IB	18
3.4	Soupis objemů jednotlivých záloh pro DB	18
3.5	Tabulka porovnání typů záloh	20
7.1	Možné hodnoty proměnné dny určující, které dny se provede záloha	34
7.2	Možné hodnoty proměnné typ_zalohy	35
7.3	Možné typy záloh pro jednotlivé rotace úložišť	37
7.4	Ukázkový obsah proměnné rozlozeni_dat s popisem řádků ve funkci hanoi	40
7.5	Ukázkový obsah proměnné pouziti_ulozist s popisem řádků ve funkci hanoi	41
7.6	Ukázkový obsah proměnné matice_tyden s popisem řádků ve funkci rot_x_db	43
7.7	Hodnoty proměnné typ_kombinace a jejich kombinace typu záloh	44
8.1	Naměřená data pro čtyři zátěže u rotace GFS	54
8.2	Naměřená data pro čtyři zátěže u Rotace X úložišť	55
8.3	Naměřená data pro rotaci Hanojské věže	56

1 Úvod

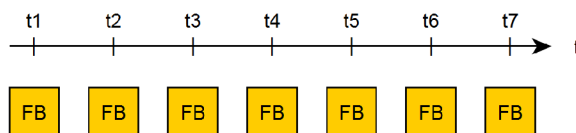
Tato práce se věnuje oblasti zálohování dat. V druhé kapitole jsou popsány typy zálohovacích schémat: úplná záloha, inkrementální záloha, rozdílová záloha a záloha kombinovaná. Následuje jejich porovnání. Třetí kapitola obsahuje matematický popis typů záloh včetně jejich porovnání. Čtvrtá kapitola je věnována popisu strategií zálohování, je zde popsána strategie Round Robin, Strategie 6 úložišť, GFS a strategie Hanojské věže. V páté kapitole se nachází matematický popis výše vypsáných strategií. V šesté kapitole popisují typy záložních médií. Optické disky, magnetické pásky, SDD a HDD disky. V závěru popisují typy RAID diskových polí. Sedmá kapitola obsahuje popis programu pro kvantitativní analýzu a porovnávání typů záloh a strategií. Obsahuje popis funkcí (jednotlivých souborů) daného programu. Osmá kapitola se věnuje porovnání naměřených dat, je zde porovnání zátěže úložišť pro jednotlivé dny všech uvedených zálohovacích schémat. V deváté kapitole je popis výběru optimálního schématu zálohy.

2 Zálohování

V této kapitole jsou popsány typy záloh - úplná, inkrementální, rozdílová a kombinovaná. Informace jsou čerpány z [1][3].

2.1 Úplná záloha

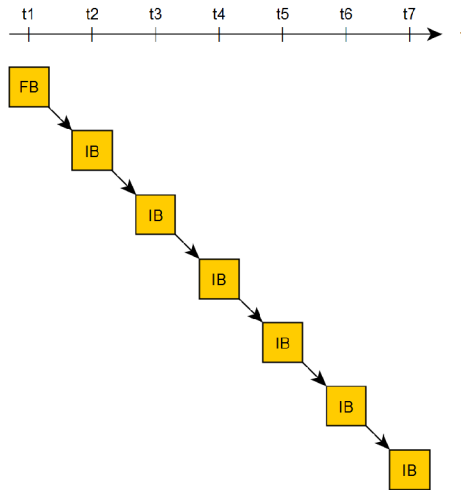
Úplná záloha neboli Full Backup (FB) je typ zálohy, kdy se kopírují všechna data bez ohledu na to, zda-li byla kopírována nebo se od předešlé úplné zálohy změnila. (Obr. 2.1) Tento typ zálohy je jednoduchý, ale také neefektivní na prostor pro zálohy. Velikost jedné úplné zálohy je rovna celkové velikosti zálohovaného prostoru. Při této záloze vzniká plno duplikovaných dat a při velkém objemu zálohovaného místa je tento typ zálohy i časově náročný. Výhoda této zálohy je rychlá možnost obnovy, kdy se jen zkopírují data ze záložního úložiště na hlavní úložiště. Na tomto typu zálohy jsou závislé ostatní typy záloh.



Obr. 2.1: Schéma úplné zálohy

2.2 Inkrementální záloha

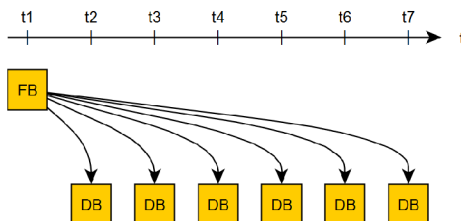
Inkrementální záloha neboli Incremental Backup (IB). Tvoří se nejprve úplnou zálohou a následně se zálohují intervalově pouze data, která se změnila od předešlé zálohy. Po určité době, např. po týdnu, je vhodné udělat opět úplnou zálohu. (Obr. 2.2) Výhodou této metody je, že má nejmenší objem záloh. Naopak nevýhodou je nejsložitější obnova zálohovaných dat, kdy se nejprve nahraje na hlavní úložiště úplná záloha a poté se musí přehrát všemi dalšími zálohami, postupně, od doby úplné zálohy až do požadovaného času. Pokud je jedna ze záloh zničena, nejde provést obnovu.



Obr. 2.2: Schéma inkrementální zálohy

2.3 Rozdílová záloha

Rozdílová záloha neboli Differential Backup (DB). Jde o kompromis mezi úplnou zálohou a inkrementální zálohou. Tvoří se tak, že se nejdříve udělá úplná záloha a potom se periodicky dělají zálohy intervalové, kdy se zálohují jen data, která se změnila od úplné zálohy. (Obr. 2.3) Je výhodná při malé změně zálohovaných dat, protože při velkém objemu měnících se dat může velikost zálohy velmi narůstat. Obnova dat je jednoduchá, nejprve se nahraje na hlavní úložiště úplná záloha, poté se přes ni přehraje záloha, např. požadovaného dne, tím se obnoví data v požadovaném dni.

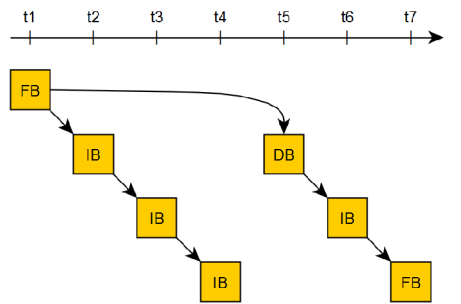


Obr. 2.3: Schéma rozdílové zálohy

2.4 Kombinovaná záloha

Kombinovaná záloha (KZ) spojuje všechny uvedené typy záloh do jednoho schématu, viz obr. 2.4. První záloha je úplná (FB), další tři jsou inkrementální (IB), čtvrtá je

rozdílová (DB) a poslední dvě jsou opět inkrementální (IB).



Obr. 2.4: Schéma kombinované zálohy

2.5 Porovnání typů záloh

Když chceme zálohovat 100 GB disk v našem počítači a za jeden den se nám změní 40 GB dat, tak při použití úplné zálohy by za týden byl objem záloh 700 GB ($7 * 100$ GB). Pokud bychom použili rozdílové zálohování, objem záloh by byl 940 GB (100 GB + 40 GB + 80 GB + ... + 240 GB). Při inkrementální záloze by objem záloh byl pouze 340 GB (100 GB + $6 * 40$ GB). U kombinované zálohy by objem záloh byl 400 GB, viz obr. 2.5.

	FB	DB	IB	KZ
Pondělí	100	100	100	100
Úterý	100	40	40	40
Středa	100	40 40	40	40
Čtvrtek	100	40 40 40	40	40
Pátek	100	40 40 40 40	40	100
Sobota	100	40 40 40 40 40	40	40
Neděle	100	40 40 40 40 40 40	40	40
	700	940	340	400

Obr. 2.5: Porovnání typů záloh

3 Matematický popis záloh

V této kapitole se budeme věnovat matematickému popisu záloh. Vzorce zde uvedené jsou čerpány z literatury [2][1]. Proměnná D značí celkový objem dat v paměťovém úložišti. Proměnná B je objem zálohy. Počet záloh je dán proměnnou M . Předpokládáme, že zálohování probíhá periodicky s odstupem T mezi zálohami. Pravděpodobnost p udává, že mezi po sobě jdoucími zálohami dojde ke změně datové jednotky. Doplňující jednotkou k pravděpodobnosti p je pravděpodobnost q , udávající, že ke změně nedojde.

$$q = 1 - p \quad (3.1)$$

Pravděpodobnost Q_k , že se datová jednotka nezmění po době k záloh ($k * T$) udává vzorec:

$$Q_k = q^k \quad (3.2)$$

Pravděpodobnost P_k , že se datová jednotka po dobu k záloh ($k * T$) změní, je dána rovnicí:

$$P_k = 1 - Q_k = 1 - q^k \quad (3.3)$$

Každé schéma zálohy má dva parametry:

C - Celkový objem záloh, což jsou součty objemů všech záloh. Tento parametr nám říká, jaká musí být celková kapacita úložiště pro všechny zálohy.

$$C = \sum_{i=1}^M (B_i) \quad (3.4)$$

E - Střední objem obnovovacích záloh - je zprůměrovaný součet všech objemů záloh, které se musí zapsat do úložiště při obnově dat do určitého času. Např. u inkrementální zálohy, objem obnovovacích záloh je různý pro každý den.

$$E = \frac{1}{M} \sum_{i=1}^M (E_i) \quad (3.5)$$

3.1 Kombinovaná záloha

Nejprve si ukážeme výpočet u kombinované zálohy, způsob výpočtu je obdobný u ostatních záloh, kde bude jen uveden výsledný vzorec. Dle schématu na obr. 2.4 je v tab. 3.1 vytvořen soupis objemů jednotlivých záloh a v tab. 3.2 soupis objemů obnovovacích záloh. Následně jsou provedeny výpočty celkového objemu záloh C a středního objemu obnovovacích záloh E .

Den	Čas	objem zálohy B
Neděle	t_1	D
Pondělí	t_2	$D * (1 - q)$
Úterý	t_3	$D * (1 - q)$
Středa	t_4	$D * (1 - q)$
Čtvrtek	t_5	$D * (1 - q^4)$
Pátek	t_6	$D * (1 - q)$
Sobota	t_7	$D * (1 - q)$

Tab. 3.1: Soupis objemů jednotlivých záloh pro KZ

$$\begin{aligned}
C_{KZ} &= \sum_{i=1}^M (B_i) = D + D * (1 - q) + D * (1 - q) + D * (1 - q) + D * (1 - q) + D * (1 - q^4) + \\
&+ D * (1 - q) + D * (1 - q) = \\
&D * (1 + 1 - q + 1 - q + 1 - q + 1 - q^4 + 1 - q + 1 - q) = \\
&D * (7 - 5 * q - q^4)
\end{aligned} \tag{3.6}$$

Den	Čas	objem obnovovacích záloh E
Neděle	t_1	D
Pondělí	t_2	$D + D * (1 - q)$
Úterý	t_3	$D + D * (1 - q) + D * (1 - q)$
Středa	t_4	$D + D * (1 - q) + D * (1 - q) + D * (1 - q)$
Čtvrtek	t_5	$D + D * (1 - q^4)$
Pátek	t_6	$D + D * (1 - q^4) + D * (1 - q)$
Sobota	t_7	$D + D * (1 - q^4) + D * (1 - q) + D * (1 - q)$

Tab. 3.2: Soupis objemů obnovovacích záloh pro KZ

$$\begin{aligned}
E_{KZ} &= \frac{1}{M} \sum_{i=1}^M (E_i) = \frac{1}{M} (D + D + D * (1 - q) + D + D * (1 - q) + D * (1 - q) + \\
&+ D + D * (1 - q) + D * (1 - q) + D * (1 - q) + D + \\
&+ D * (1 - q^4) + D + D * (1 - q^4) + D * (1 - q) + \\
&+ D + D * (1 - q^4) + D * (1 - q) + D * (1 - q)) = \\
&\frac{1}{M} (D * (1 + 1 - q + 1 + 1 - q + 1 - q + 1 + 1 - q + 1 - q + 1 - \\
&- q + 1 + 1 - q^4 + 1 + 1 - q^4 + 1 - q + 1 + 1 - q^4 + 1 - q + 1 - q)) = \\
&\frac{1}{M} (D * (17 - 9q - 3q^4))
\end{aligned} \tag{3.7}$$

3.2 Úplná záloha

Matematický popis úplné zálohy se dá zapsat jako rovnice $FB(t_i) = D(t_i)$. Což znamená, že objem zálohy v čase t_i je roven celkovému objemu dat v čase t_i . Z čehož vyplývá, že pokud chceme zpočítat celkovou kapacitu záloh, stačí sečíst objem dat D pro všechny zálohy. Vzorce převzaty z [1].

$$C_{FB} = \sum_{i=1}^M D(t_i) = M * D \tag{3.8}$$

$$E_{FB} = \frac{1}{M} \sum_{i=1}^M (E_i) = \frac{M * D}{M} = D \tag{3.9}$$

3.3 Inkrementální záloha

První záloha je záloha úplná (FB), ostatní zálohy jsou intervalové. Intervalová záloha zde ukládá pouze data, která se změnila od předchozí zálohy, tudíž se počítá pravděpodobnost pouze pro změnu po 1 záloze: $P_k = 1 - q^1 = 1 - q$. Dle schématu na obr. 2.2 je tedy celkový objem záloh roven součtu všech dílčích objemů záloh viz tab. 3.3. Vzorce převzaty z [1].

$$B(t_i) = \begin{cases} D, & i = 1, \\ D * (1 - q) & i = 2, 3, \dots, M \end{cases}$$

$$C_{IB} = \sum_{i=1}^M (B_i) = D * [M - (M - 1) * q] \tag{3.10}$$

$$E_{IB} = \frac{1}{M} \sum_{i=1}^M (E_i) = \frac{D}{2} * [(M + 1) - (M - 1) * q] \tag{3.11}$$

Den	Čas	objem zálohy B
Neděle	t_1	D
Pondělí	t_2	$D * (1 - q)$
Úterý	t_3	$D * (1 - q)$
Středa	t_4	$D * (1 - q)$
Čtvrtek	t_5	$D * (1 - q)$
Pátek	t_6	$D * (1 - q)$
Sobota	t_7	$D * (1 - q)$

Tab. 3.3: Soupis objemů jednotlivých záloh pro IB

3.4 Rozdílová záloha

První záloha je úplná záloha (FB) a ostatní zálohy jsou intervalové, referenční zálohou pro intervalové zálohy je úplná záloha (první). Z toho vyplývá, že se počítá pravděpodobnost pro změnu vzorcem $P_k = 1 - q^k$, kde k je pořadí zálohy od první zálohy. Dle schématu na obr. 2.3 je tedy celkový objem záloh roven součtu všech dílčích objemů záloh viz tab. 3.4. Vzorce převzaty z [1].

$$B(t_i) = \begin{cases} D, & i = 1, \\ D * (1 - q^{i-1}) & i = 2, 3, \dots, M \end{cases}$$

Den	Čas	objem zálohy B
Neděle	t_1	D
Pondělí	t_2	$D * (1 - q^1)$
Úterý	t_3	$D * (1 - q^2)$
Středa	t_4	$D * (1 - q^3)$
Čtvrtek	t_5	$D * (1 - q^4)$
Pátek	t_6	$D * (1 - q^5)$
Sobota	t_7	$D * (1 - q^6)$

Tab. 3.4: Soupis objemů jednotlivých záloh pro DB

$$C_{DB} = \sum_{i=1}^M (B_i) = D * \left(M - \frac{q - q^M}{1 - q} \right) \quad (3.12)$$

$$E_{DB} = \frac{1}{M} \sum_{i=1}^M (E_i) = \frac{D}{M} * \left[(2 * M - 1) - \frac{q - q^M}{1 - q} \right] \quad (3.13)$$

3.5 Matematické porovnání záloh

Nyní aplikujeme matematický popis záloh na příklad z předešlé kapitoly obr. 2.5. Máme disk o kapacitě $D = 100GB$, denní změnu dat si označíme proměnou ZD , za den se změní $40GB$ dat, tudíž $ZD = 40GB$, počet dní, který chceme zálohovat, je $M = 7$. Nejdříve vypočítáme pravděpodobnost p , že se data mezi zálohami změní. To vypočítáme trojčlenkou, kdy chceme vědět kolik % z celkové kapacity D je velikost denních datových změn.

$$p = \frac{ZD}{D} = \frac{40}{100} = 0,4 \quad (3.14)$$

Dále vypočítáme pravděpodobnost, že se data mezi zálohami nezmění:

$$q = 1 - p = 1 - 0,4 = 0,6 \quad (3.15)$$

Nyní můžeme přejít na výpočet charakteristických parametrů C a E pro jednotlivé typy záloh.

3.5.1 Výpočet úplné zálohy

$$C_{FB} = \sum_{i=1}^M D(t_i) = M * D = 7 * 100 = \underline{\underline{700 GB}} \quad (3.16)$$

$$E_{FB} = D = \underline{\underline{100 GB}} \quad (3.17)$$

3.5.2 Výpočet inkrementální zálohy

$$\begin{aligned} C_{IB} &= D * [M - (M - 1) * q] = \\ &= 100 * [7 - (7 - 1) * 0,6] = \\ &= 100 * [7 - (6 * 0,6)] = \\ &= 100 * [7 - 3,6] = 100 * 3,4 = \underline{\underline{340 GB}} \end{aligned} \quad (3.18)$$

$$\begin{aligned} E_{IB} &= \frac{D}{2} * [(M + 1) - (M - 1) * q] = \\ &= \frac{100}{2} * [(7 + 1) - (7 - 1) * 0,6] = \\ &= 50 * [8 - (6 * 0,6)] = \\ &= 50 * [8 - 3,6] = 50 * 4,4 = \underline{\underline{220 GB}} \end{aligned} \quad (3.19)$$

3.5.3 Výpočet rozdílové zálohy

$$\begin{aligned}
 C_{DB} &= D * \left(M - \frac{q - q^M}{1 - q} \right) = 100 * \left(7 - \frac{0,6 - 0,6^7}{1 - 0,6} \right) = \\
 &100 * \left(7 - \frac{0,6 - 0,02799}{0,4} \right) = 100 * \left(7 - \frac{0,572}{0,4} \right) = \\
 &100 * (7 - 1,43) = 100 * 5,57 = \underline{\underline{557 GB}}
 \end{aligned} \tag{3.20}$$

$$\begin{aligned}
 E_{DB} &= \frac{D}{M} * \left[(2 * M - 1) - \frac{q - q^M}{1 - q} \right] = \\
 &\frac{100}{7} * \left[(2 * 7 - 1) - \frac{0,6 - 0,6^7}{1 - 0,6} \right] = \\
 &14,286 * \left[(14 - 1) - \frac{0,6 - 0,02799}{0,4} \right] = \\
 &14,286 * \left[13 - \frac{0,572}{0,4} \right] = \\
 &14,286 * [13 - 1,43] = 14,286 * 11,57 = \underline{\underline{165,3 GB}}
 \end{aligned} \tag{3.21}$$

3.5.4 Výpočet kombinované zálohy

$$\begin{aligned}
 C_{KZ} &= D * (7 - 5 * q - q^4) = 100 * (7 - 5 * 0,6 - 0,6^4) = \\
 &100 * (7 - 3 - 0,1296) = 100 * 3,8704 = \underline{\underline{387,04 GB}}
 \end{aligned} \tag{3.22}$$

$$\begin{aligned}
 E_{KZ} &= \frac{1}{M} (D * (17 - 9q - 3q^4)) = \frac{1}{7} (100 * (17 - 9 * 0,6 - 3 * 0,6^4)) = \\
 &\frac{100}{7} (17 - 5,4 - 0,3888) = 14,2857 * 11,2112 = \underline{\underline{160,16 GB}}
 \end{aligned} \tag{3.23}$$

3.5.5 Porovnání výsledků

Záloha	Objem záloh (C)	Objem obnovovacích záloh (E)
Úplná (FB)	700 GB	100 GB
Inkrementální (IB)	340 GB	220 GB
Rozdílová (DB)	557 GB	165,3 GB
Kombinovaná (KZ)	387,04 GB	160,16 GB

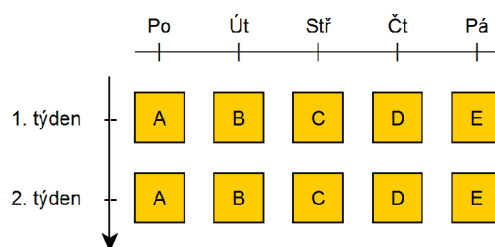
Tab. 3.5: Tabulka porovnání typů záloh

4 Strategie

Strategie zálohování udává pravidla, podle kterých se mění (rotují) záložní úložiště. Tím dochází k přepisu starých již nepotřebných záloh. Rovnoměrné nebo nerovnoměrné opotřebení záložních úložišť udává typ strategie. Tato kapitola čerpá z publikací[1][3][4].

4.1 Round Robin

Rotace úložišť typu Round Robin je nejstarší a nejjednodušší schéma rotace záloh. Používají se pouze úplné zálohy FB, pro každý den je jiné úložiště (obr. 4.1). V pondělí se záloha zapíše na úložiště A, v úterý na úložiště B, . . . Výhoda této rotace je její jednoduchost a stejnoměrné opotřebení úložišť.

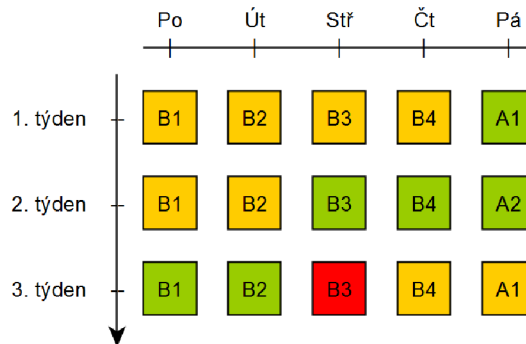


Obr. 4.1: Ukázka rotace Round Robin

4.2 Rotace 6 úložišť

Rotace 6 úložišť používá 2 úložiště (A1 a A2) pro úplné zálohy (FB), které se provádí na přeskáčku každý pátek, a 4 úložiště (B1, B2, B3 a B4) pro rozdílovou (BD) nebo inkrementální zálohu (IB) na zbylé pracovní dny (pondělí až čtvrtek). Minimální počet u tohoto typu rotace je 5 úložišť, na konci týdne (pátek) se provede úplná záloha a další týden od pondělí do čtvrtka se provádějí rozdílové či inkrementální zálohy a v pátek se opět provede záloha úplná. V našem příkladě se 6 úložišti je možno se dostat do dvou předcházejících pátků, pokud bychom chtěli takhle zálohovat celý měsíc, budeme potřebovat úložišť 8 (čtyři na dny pondělí–čtvrtek a další čtyři na pátky v celém měsíci). Při záloze roční, s možností obnovy dat do libovolného pátku v roce, bychom potřebovali úložišť 56. Na obr. 4.2 ve středu 3. týdne (červený čtvereček) se můžeme dostat do dvou předchozích pátků a při použití inkrementální zálohy i do pondělí a úterý 3. týdne. Pokud se používá rozdílové zálohování, můžeme obnovit data i ze středy a čtvrtka 2. týdne (zelené čtverečky).

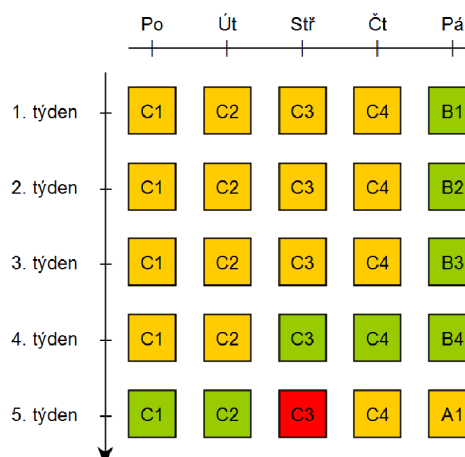
Rotace úložišť je vhodná pro malé firmy, které potřebují zálohovat maximálně pár týdnů starou práci.



Obr. 4.2: Ukázka rotace 6 úložišť

4.3 GFS

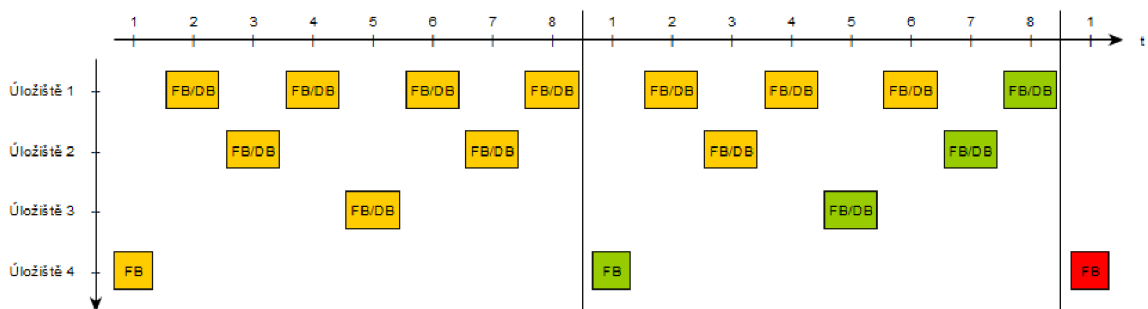
Grandfather, Father, Son, neboli Děd, Otec, Syn, je rotace záloh umožňující při použití 20 úložišť obnovit data až 1 rok stará. GFS se skládá celkem z 20 úložišť. Z 12 úložišť (A1 – A12) – Děd – kam se ukládá úplná záloha (FB) poslední pátek v daném měsíci. Ze 4 úložišť (B1 – B4) – Otec – obsahující úplné zálohy (FB) každý pátek daného měsíce. A ze 4 úložišť (C1 – C4) – Syn – obsahujících inkrementální (IB) nebo rozdílovou zálohu (DB) dnů pondělí až čtvrtek (obr. 4.3). Díky tomu lze obnovit data z konce předešlých 12 měsíců, z předešlých 4 pátků a dle typu zálohy z předešlých dní (princip jak u rotace 6 úložišť).



Obr. 4.3: Ukázka rotace Grandfather - Father - Son

4.4 Hanojské věže

Rotace Hanojské věže používá n úložišť, kde N té úložiště je vždy úplná záloha (FB), ostatní zálohy jsou buď úplné (FB) nebo rozdílové (DB). Inkrementální zálohu (IB) nelze použít z důvodů nepravidelného střídání úložišť. Výhoda této rotace spočívá v nízkém počtu úložišť pro překlenutí dlouhé doby zálohy. Nevýhoda je nesoulad s kalendářem a nerovnoměrné opotřebení úložišť. Příklad se 4 úložišti: úložiště 1 se používá každý 2. den, úložiště 2 každý 4. den, úložiště 3 a 4 každý 8. den a vzájemně se střídají, viz obr. 4.4. Obecněji: úložiště U_i $i < n$, se použije každý 2^i tý den. Úložiště U_n se použije se stejnou periodou jako U_{n-1} . Dle počtu úložišť n můžeme zálohovat data 2^{n-1} dní stará. Při použití 4 úložišť $2^3 = 8$ dní. Při použití 10 úložišť můžeme získat data až 512 dní stará. Pokud chceme obnovit data na konci periody (červený čtvereček), resp. na začátku periody než se vytvořila nová záloha, tak můžeme v našem případě pro 4 úložiště obnovit data 1., 2., 4. a 8. dne v historii (zelené čtverečky).



Obr. 4.4: Ukázka rotace Hanojské věže

5 Matematický popis strategií

V této kapitole se budeme, věnovat matematickému popisu strategií.

5.1 Rotace Round Robin

Rotace Round Robin používá jen úplné zálohy (FB), takže jeho matematický popis je roven popisu úplné zálohy. Počet záloh M je dán počtem dní, do kterých chceme mít možnost obnovit data.

$$C_{rotace\ Round\ Robin} = \sum_{i=1}^M D(t_i) = D * \text{počet dní} \quad (5.1)$$

Střední objem záloh je roven objemu zálohovaných dat.

$$E_{rotace\ Round\ Robin} = D \quad (5.2)$$

5.2 Rotace 6 úložišť

Rotace 6 úložišť se skládá z inkrementální či rozdílové zálohy pro dny pondělí až čtvrtek a dvou úplných záloh pro pátky.

5.2.1 Výpočet pro Inkrementální verzi

$$C_{IB\ rotace\ úložišť} = D * [M - (M - 1) * q] + D * (\text{počet pátků} - 1) \quad (5.3)$$

$$E_{IB\ rotace\ úložišť} = \frac{D}{2} * [(M + 1) - (M - 1) * q] + D \quad (5.4)$$

5.2.2 Výpočet pro Rozdílovou verzi

$$C_{DB\ rotace\ úložišť} = D * \left(M - \frac{q - q^M}{1 - q} \right) + D * (\text{počet pátků} - 1) \quad (5.5)$$

$$E_{DB\ rotace\ úložišť} = \frac{D}{M} * \left[(2 * M - 1) - \frac{q - q^M}{1 - q} \right] + D \quad (5.6)$$

5.3 GFS

Rotace GFS se skládá ze zálohy IB či DB pro všední dny pondělí až čtvrtek. Dále rotaci úložišť pro konce týdnů (pátky) a rotaci úložišť dle počtu měsíců, které chceme mít možnost obnovit.

5.3.1 Výpočet pro Inkrementální verzi

$$C_{IB \text{ rotace } GFS} = D * [M - (M - 1) * q] + D * (\text{počet pátků} - 1) + D * \text{počet měsíců} \quad (5.7)$$

$$E_{IB \text{ rotace } GFS} = \frac{D}{2} * [(M + 1) - (M - 1) * q] + D + D \quad (5.8)$$

5.3.2 Výpočet pro Rozdílovou verzi

$$C_{DB \text{ rotace } GFS} = D * \left(M - \frac{q - q^M}{1 - q} \right) + D * (\text{počet pátků} - 1) + D * \text{počet měsíců} \quad (5.9)$$

$$E_{DB \text{ rotace } GFS} = \frac{D}{M} * \left[(2 * M - 1) - \frac{q - q^M}{1 - q} \right] + D + D \quad (5.10)$$

5.4 Hanojské věže

U strategie Hanojské věže je počet záloh M dán vztahem:

$$M = 2^{N-1} \quad (5.11)$$

Kde N je počet úložišť, v následujících rovnicích je již vzorec dosazen místo proměnné M .

5.4.1 Výpočet pro Úplnou verzi

$$C_{FB \text{ rotace Hanojské věže}} = \sum_{i=1}^{2^{N-1}} D(t_i) \quad (5.12)$$

$$E_{FB \text{ rotace Hanojské věže}} = D \quad (5.13)$$

5.4.2 Výpočet pro Rozdílovou verzi

$$C_{DB \text{ rotace Hanojské věže}} = D * \left(2^{N-1} - \frac{q - q^{2^{N-1}}}{1 - q} \right) \quad (5.14)$$

$$E_{DB \text{ rotace Hanojské věže}} = \frac{D}{2^{N-1}} * \left[(2 * 2^{N-1} - 1) - \frac{q - q^{2^{N-1}}}{1 - q} \right] \quad (5.15)$$

6 Zálohovací média

6.1 Optický disk

Optický disk je plastový disk o průměru 120 nebo 80 mm a 1,2 mm silný. K čtení a zápisu slouží laserový paprsek. Data na disku jsou uložena ve formě prohlubní (pit, log.0) a výstupků (land, log. 1) v datové vrstvě. Pity a landy jsou uloženy ve spirále od středu disku po okraj.[5]

CD je nejstarší, kapacita je 700 MB, dnes se kvůli kapacitě moc nepoužívají. Dalším typem je DVD s kapacitami od 4,7 GB pro jednovrstvé až po 17,1 GB pro oboustranné a dvouvrstvé. Posledním typem je BD, základní kapacita jednovrstvého BD disku je 25 GB až po 100 GB pro 4vrstvé disky.

U výše zmíněných optických disků existují tři varianty disků. ROM je pouze pro čtení, vznikají tzv. lisováním. R umožňují zápis dat. RW značí přepisovatelný disk, je možné ho smazat a znovu zapsat data. Nevýhoda je, že mazání a zápis se provádí s celým diskem. [5][6][8]

U typu DVD existuje i varianta RAM – Nemá data uložena na jedné spirále, ale spirálu má rozdělena na plno malých obdélníků, kterými určuje sektory a tak přistupuje k datům podobně jako pevný disk.[7]

Dále existuje M-Disc (Millennial-Disc) firmy Millennial Inc. Jde o optický disk určený výhradně pro archivaci dat, založený na DVD či BD s mnohem vyšší životností než klasické optické disky. Mechaniky pro zápis musí umět s M-disky pracovat a upravit výkon laseru.[9]

6.2 Magnetická páska

Jde o plastickou (celuliodovou) pásku potaženou magnetickou vrstvou, její historie sahá do roku 1926, kde místo celuloidu byl použit papír. Čtení a zápis obstarává magnetická hlava. Magnetická páska sloužila pro zaznamenávání zvuku. V roce 1951 byla magnetická páska prvně použita pro uchování videa.[12] Ve stejném roce byla prvně použita i pro uchovávání digitálních dat na sálovém počítači UNIVAC 1.[10] V historii bylo několik typů magnetických pásek pro ukládání dat, dnes je nejpoužívanější LTO technologie.

LTO je technologie pro ukládání dat na magnetické pásky, vyvinutá v 90. let firmami HP, IBM a Quantum. V roce 2000 byla představena první generace magnetických pásek LTO gen. 1 s kapacitou 100 GB, při použití komprimace dat 200 GB a maximální přenosové rychlosti 20 MB/s, při komprimaci 40 MB/s. Dnešní verze LTO gen. 8 má kapacitu 12 TB (30 TB při použití komprimace) s přenosovými

rychlostmi až 360 MB/s a 900 MB/s při použití komprimace. Životnost pásek je 15 až 30 let a minimálně 200 přepisů. [11][13]

6.3 SSD disk

SSD disk je elektronické zařízení pro ukládání digitálních dat založené na paměti typu Flash. Jde v podstatě o velký USB flash disk, který je ale připojen ne pomocí USB, ale pomocí interních rozhraní počítače, např. SATA, M.2, PCIe. Oproti pevným diskům nemá žádné mechanické díly, tudíž mezi jeho výhody patří nehlučnost, odolnost proti otřesům. Dále nízká spotřeba elektřiny a velké přenosové rychlosti. Přístupová doba SSD je v řádech μs , u pevných disků HDD je v ms . Nevýhoda SSD disku je větší cena za GB, nemožnost obnovy dat při hardwarové chybě, kdy např. řídicí elektronika přestane fungovat. Další diskutabilní nevýhodou je životnost disku, z důvodů omezeného zápisu na buňku. Většina výrobců u svých výrobků udává parametr TBW, který deklaruje minimální životnost disku.[15]

6.4 Pevný disk HDD

Pevný disk HDD je elektromechanické zařízení k ukládání digitálních dat, které zůstanou zachována i po odpojení elektřiny. Jeho hlavní části tvoří plotny, kde se uchovávají data, motor k otáčení ploten, hlavy pro čtení a zápis, pohon hlav a řídicí elektronika. Dnes se nejčastěji používají disky o velikosti 3,5 palce pro stolní počítače a servery a 2,5 palce pro mobilní počítače. Mezi výhody patří nízká cena za GB, možnost obnovy dat při hardwarové chybě, životnost. Nevýhodami jsou hlučnost, spotřeba elektrické energie, nižší odolnost proti otřesům, nižší přenosové rychlosti oproti SSD diskům.

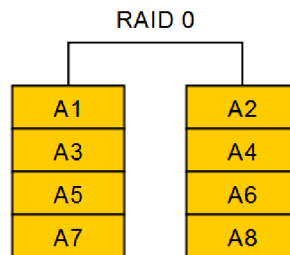
6.5 Diskové pole RAID

Jde o pole více disků, mezi které se rozloží ukládání dat, a při selhání některého z disků lze data obnovit. Pole RAID zvyšuje spolehlivost úložiště, rychlost zápisu nebo čtení dat. Obsluhu zajišťuje RAID řadič. Při popisu RAID polí byly použity materiály [16][17][18][19][3].

6.5.1 RAID 0

Neposkytuje žádnou ochranu při výpadku disku. Data jsou rozdělena na bloky a každý blok je zapsán současně na různé disky, tzv. prokládání. Při výpadku jednoho

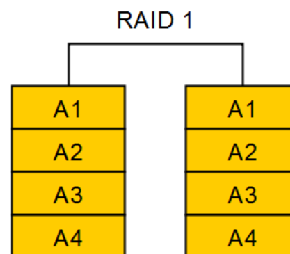
disku se přijde o data. Výhoda tohoto pole je vysoká rychlost čtení a zápisu, protože se pracuje se všemi disky současně. Minimální počet disků jsou 2 kusy. Schéma zapojení je na obr. 6.1.



Obr. 6.1: Schéma zapojení diskového pole RAID 0

6.5.2 RAID 1

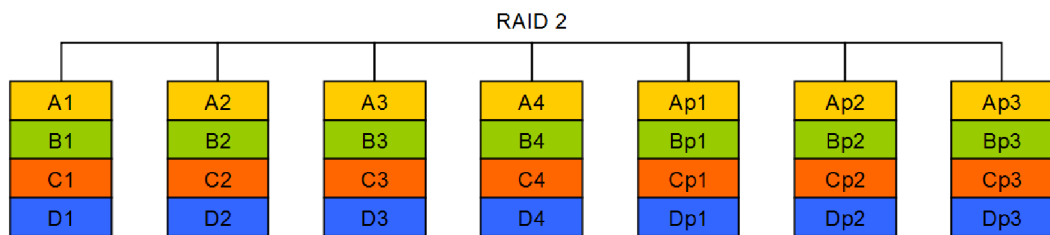
Poskytuje ochranu při výpadku 1 disku. Blok dat se zapisuje na oba disky současně tzv. mirroring neboli zrcadlení. Rychlost zápisu je stejná jako pro jeden disk. Rychlost čtení je až dvojnásobná, protože data můžou být čtena zároveň z obou disků. Minimální počet disků jsou 2 kusy. Schéma zapojení je na obr. 6.2.



Obr. 6.2: Schéma zapojení diskového pole RAID 1

6.5.3 RAID 2

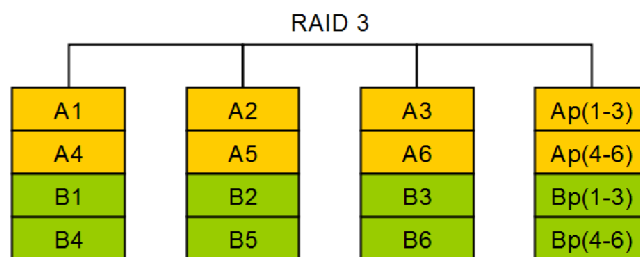
Je založeno na bitovém prokládání, kdy jsou data rozložena na bity a každý bit je uložen na odlišný disk. Parita dat je uložena na dalším disku pomocí Hammingova kódu, díky kterému je možno rozpoznat a opravit chyby při čtení. Vždy je potřeba o jeden disk méně než u zrcadlení, takže pro čtyři datové disky jsou potřeba tři redundantní. Dnes se nepoužívá. Schéma zapojení je na obr. 6.3.



Obr. 6.3: Schéma zapojení diskového pole RAID 2

6.5.4 RAID 3

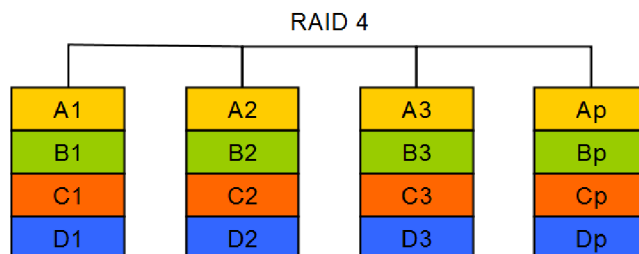
Je založeno na Bytovém prokládání, kdy jsou data rozložena na Byty a každý Byte je uložen na odlišný disk. Parita je uložena zvlášť na jeden disk jako exkluzivní součet (XOR). Při výpadku jednoho disku jde pomocí zbylých disků a paritního disku dopočítat chybějící data. Při výpadku paritního disku jsou data zachována. Schéma zapojení je na obr. 6.4.



Obr. 6.4: Schéma zapojení diskového pole RAID 3

6.5.5 RAID 4

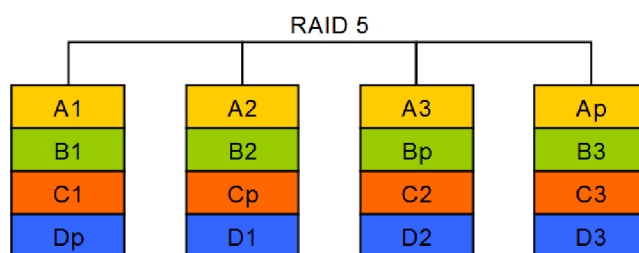
Data jsou prokládána po blocích, parita je zvlášť na disku. U RAID 2 a 3 jeden požadavek na čtení, vyžaduje čtení ze všech datových disků. U RAID 4 jsou data v blocích, takže je možno je číst z různých disků zároveň. Tím pádem je rychlost čtení vyšší než u RAID 2 a 3. Schéma zapojení je na obr. 6.5.



Obr. 6.5: Schéma zapojení diskového pole RAID 4

6.5.6 RAID 5

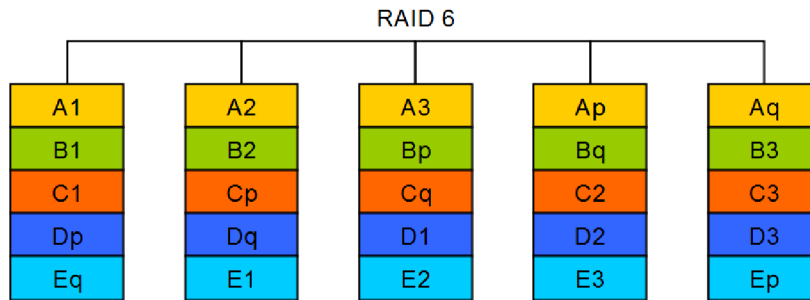
Je založen na blokovém prokládání s paritou, kde parita je prokládána skrz všechny disky. Odolá výpadku 1 z N disků. Počítá paritní blok xorováním datových bloků po bitech. Teoreticky zápis a čtení lze zrychlit $N - 1$ krát, zápis může být zpomalen výpočtem parity. Minimální počet disků jsou 3 kusy. Obnovení RAID 5 pole při výpadku jednoho disku je časově náročné. Schéma zapojení je na obr. 6.6.



Obr. 6.6: Schéma zapojení diskového pole RAID 5

6.5.7 RAID 6

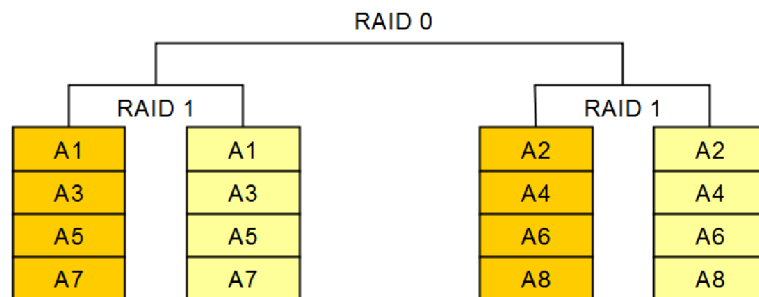
Je založen na blokovém prokládání s dvěma paritami, kde parity jsou prokládány skrz všechny disky. Vypořádá se s výpadkem 2 z N disků. Paritní blok p se vypočítá xorováním po bitech, paritní blok q se vypočítá Reed-Solomonovým korekčním kódem. Oba paritní bloky umožňují vypočítat (obnovit) dva bloky. Rychlost čtení a zápis lze zrychlit $N - 2$ krát. Zápis bývá pomalejší než u RAID 5 kvůli náročnému výpočtu parity q . Min počet disků jsou 4 kusy. Schéma zapojení je na obr. 6.7.



Obr. 6.7: Schéma zapojení diskového pole RAID 6

6.5.8 RAID 10

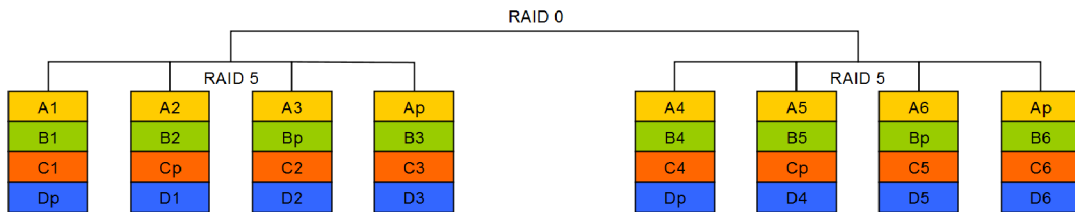
Jde o kombinace RAID 1 a RAID 0. Jde o pole s dvojnásobnou rychlostí zápisu a čtení, které se vypořádá s výpadkem jednoho disku v každém z polí RAID 1. Min počet disků jsou 4 kusy. Schéma zapojení je na obr. 6.8.



Obr. 6.8: Schéma zapojení diskového pole RAID 10

6.5.9 RAID 50

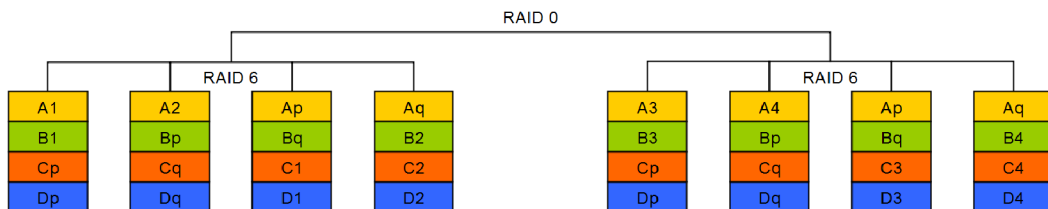
Jde o kombinace RAID 5 a RAID 0. Jde o pole s teoreticky $X * (N - 1)$ krát rychlejším zápisem a čtením, kde X je počet polí RAID 5. Vypořádá se s výpadkem jednoho disku v každém z polí RAID 5. Minimální počet disků je 6 kusů. Schéma zapojení je na obr. 6.9.



Obr. 6.9: Schéma zapojení diskového pole RAID 50 se třemi poli RAID 5

6.5.10 RAID 60

Jde o kombinace RAID 6 a RAID 0. Jde o pole s teoreticky $X * (N - 2)$ krát rychlejším zápisem a čtením, kde X je počet polí RAID 6. Vypořádá se s výpadkem dvou disků v každém z polí RAID 6. Minimální počet disků je 8 kusů. Schéma zapojení je na obr. 6.10.



Obr. 6.10: Schéma zapojení diskového pole RAID 60

7 Program

V této kapitole se popisuje funkce programu. Program je psán v Matlabu a je členěn na dvě části. První část obsahuje uživatelsky nastavitelné proměnné, druhá část dle nastavených proměnných počítá parametry zálohovacích schémat a generuje výpisy a grafy.

7.1 Uživatelská část

```
Data = 1000; %celkový objem záloh (kapacita disku (GB))
Zmena_dat = 200; %objem denní změny (Kolik GB z disku se změní za den)
```

Proměnnou *Data* nastavíme velikost úložiště které chceme zálohovat. Proměnnou *Změna_dat* nastavíme objem denní změny dat.

```
%Jaké chceme používat úložiště
Jen_SSD = 0; %1 = ano 0 = ne
Jen_HDD = 0; %1 = ano 0 = ne
SSD_i_HDD = 1; %1 = ano 0 = ne
```

Další tři proměnné fungují jako přepínač, kterým nastavíme jaké typy úložišť chceme používat: *Jen_SSD*, *Jen_HDD*, *SSD_i_HDD*.

```
%Chceme mít možnost obnovit hodně stará data a nejsme vázáni kalendářem?
Hanoi_zaloha = 0; %1 = ano 0 = ne
%Stačí nám týdení záloha + několik konců týdnů zpětně?
Tydenni_zaloha = 1; %1 = ano , 0 = ne
%Chceme zálohu týdení a s posledními 4 konci týdne a koncem každého měsíce v roce?
Rocni_zaloha = 0; %1 = ano , 0 = ne
```

Dále jsou zde tři otázky, pomocí kterých si vybereme vyhovující typ rotace záložních médií. Opět tři proměnné fungující jako přepínač.

Hanoi_zaloha – Rotace Hanojské věže

Tydenni_zaloha – Rotace X úložišť

Rocni_zaloha – Rotace GFS

```
%HANOI NASTAVENI
n=10; %počet úložišť
% n
% max stáří dat (dní)
% max stáří = 2^(n-1)
```

4	5	6	7	8	9	10	11
8	16	32	64	128	256	512	1024

```
zivotnost_cyklu = 1; %životnost, ekvivalent k počtu roků
```

Dále je blok proměnných pro nastavení rotace Hanojské věže. Zadává se zde počet úložišť *n*, které chceme použít, a životnost cyklu, což je počet opakování rotace.

```
%KALENDÁŘNÍ ZÁLOHY Pro Týdení i Roční zálohy
dny = 1; %které dny chceme zálohovat

roku = 1; %počet roků - životnost
```

Předposlední je blok pro kalendářní zálohy, zde se nastavují *dny*, které chceme zálohovat (Tab.7.1) a počet roků (*roku*), kolik chceme simulovat.

dny	Dny v kterých se provede záloha	
1	pondělí až neděle	celotýdenní provoz
2	pondělí až pátek	o víkendu se data nemění
3	pondělí, středa, pátek, neděle	celotýdenní provoz
4	pondělí, středa, pátek	o víkendu se data nemění

Tab. 7.1: Možné hodnoty proměnné dny určující, které dny se provede záloha

Poslední je blok pro týdenní zálohy, kde se pomocí proměnné *pocet_tydenich_fb* nastavuje počet konců, týdnů které chceme zálohovat.

7.2 Programová část

Jako první se vypíše nastavené hodnoty programu, objem dat, objem denní změny dat, použitá média, typ rotace médií a jejich nastavení. Vypočítá se pravděpodobnost změny dat p a pravděpodobnost q , že se data nezmění. Následně se vypočítají všechny typy záloh pro vybranou rotaci médií. Princip propočítání záloh je obdobný, bude zde popsáno počítání rotace Hanojské věže.

```

if(Hanoi_zaloha)
    if(SSD_i_HDD)
        [num,txt,raw_prvky_z] = xlsread('CENIK_SSD.xlsx');
        polozka_z = length(raw_prvky_z);
        [num,txt,raw_prvky_D] = xlsread('CENIK_HDD.xlsx');
        polozka_D = length(raw_prvky_D);
        fprintf('Úložiště SSD i HDD\n')
    elseif(Jen_HDD)
        [num,txt,raw_prvky_D] = xlsread('CENIK_HDD.xlsx');
        polozka_D = length(raw_prvky_D);
        raw_prvky_z = raw_prvky_D;
        polozka_z = length(raw_prvky_D);
        fprintf('Úložiště HDD\n')
    elseif(Jen_SSD)
        [num,txt,raw_prvky_D] = xlsread('CENIK_SSD.xlsx');
        polozka_D = length(raw_prvky_D);
        raw_prvky_z = raw_prvky_D;
        polozka_z = length(raw_prvky_D);
        fprintf('Úložiště SSD\n')

```

1. Zjistí se, které typy úložiště byly vybrány, a načtou se excelovské ceníky. Je dáno, že na úplné zálohy ve schématu se používají *raw_prvky_D* a na rozdílové a inkrementální se používají *raw_prvky_z*. U rotace Hanojské věže jsou *raw_prvky_D* používány na poslední dvě úložiště, protože se používají se stejnou periodou opakování.

```

typ_zalohy=1;
w=0;
vyhovujici_medium = 0;
for a=1:polozka_D
    if cell2mat(raw_prvky_D(a,2)) >= Data
        vyhovujici_medium = vyhovujici_medium + 1;
        TBW_D = cell2mat(raw_prvky_D(a,4));
        cena_disku_D = cell2mat(raw_prvky_D(a,3));
        nazev_disku_D = a;
        w= w+1;
        v=0;
    end
end

```

2. Nastaví se *typ_zalohy* (Tab.7.2). Proměnnou *w* se nastavuje osa *z* 3D matice s výsledky. Proměnná *vyhovujici_medium* slouží k testování zda v ceníku je použitelné médium dle zadaných požadavků v uživatelské části. Cyklem *for* se prochází ceník s *raw_prvky_D*, vybere se položka – kapacita média, porovná se, zda vyhovuje zadané velikosti úložiště, pokud ano, tak se z ceníku načtou další vlastnosti média – TBW, cena a název ve formě čísla pořadí média v ceníku. Přičte se do proměnné *w* jednička a vynuluje se proměnná *v*, která zastává stejnou funkci jako proměnná *w*, ale pro *raw_prvky_z*.

hodnota	Typ zálohy
1	FB – úplná záloha
2	DB – rozdílová záloha
3	IB – inkrementální záloha
4	KZ – kombinovaná záloha

Tab. 7.2: Možné hodnoty proměnné *typ_zalohy*

```

v=0;
for b=1:polozka_z
    if cell2mat(raw_prvky_z(b,2)) >= Data
        TBW_z = cell2mat(raw_prvky_z(b,4));
        cena_disku_z = cell2mat(raw_prvky_z(b,3));
        nazev_disku_z = b;

        vystup_funkce = hanoi(n, Data, q, typ_zalohy, zivotnost_cyklu, TBW_D,
            cena_disku_D, nazev_disku_D, TBW_z, cena_disku_z,
            nazev_disku_z );

        v=v+1;
        matice_vsech_ulozist_z(:, :, v) = vystup_funkce(:, :);
    end;
end;

```

3. Cyklem *for* se prochází ceník s *raw_prvky_z*, vybere se položka kapacita média, porovná se, zda vyhovuje zadané velikosti úložiště, pokud ano, tak se z ceníku načtou další vlastnosti média – TBW, cena a název ve formě čísla pořadí média v ceníku. U inkrementální zálohy se porovnává, zda vyhovuje zadanému objemu denní změny dat.

4. Načtené vlastnosti médií a uživatelsky zadané požadavky se vloží do funkce (zde funkce *hanoi*), která simuluje rotaci médií, a pošle zpět matici s jednotlivými údaji pro každé úložiště. U ostatních rotací pro každý den (konec týdne či měsíc).

5. Tato matice se uloží do matice *matice_vsech_ulozist_z*.

```
[x x z] = size(matice_vsech_ulozist_z);
celkova_cena_disku = zeros(1,z);
for i = 1:z
    celkova_cena_disku(1,i) = sum(matice_vsech_ulozist_z(7,:,i).*
                                matice_vsech_ulozist_z(8,:,i));
end;
[x y] = min(celkova_cena_disku);
matice_vsech_ulozist(:, :, w) = matice_vsech_ulozist_z(:, :, y);
```

6. Po skončení cyklu *for* s *raw_prvky_z* se z matice *matice_vsech_ulozist_z* vypočítá celková cena úložišť (*celkova_cena_disku*) pro všechny varianty úložišť (*raw_prvky_z*), vybere se nejlevnější a uloží se do matice *matice_vsech_ulozist*.

```
if vyhovujici_medium == 0
    fprintf('Zadanou hodnotou "Data" nevyhovuje žádné medium\n')
    return
end;
[x x z] = size(matice_vsech_ulozist);
celkova_cena_disku = zeros(1,z);
for i = 1:z
    celkova_cena_disku(1,i) = sum(matice_vsech_ulozist(7,:,i).*
                                matice_vsech_ulozist(8,:,i));
end;
[x y] = min(celkova_cena_disku);
textovy_vypis_nejlevnejsi_hanoi(n, Data, q, matice_vsech_ulozist(:, :, y), typ_zalohy)
```

7. Po skončení cyklu *for* s médii *raw_prvky_D* se otestuje zda ceník obsahoval použitelná média, pokud ne, tak skončí program. Jinak se z matice *matice_vsech_ulozist* vypočítá celková cena úložišť, vybere se nejlevnější varianta a vypíše se pomocí funkce (*textovy_vypis_nejlevnejsi_hanoi*), kde se vypíše data pro jednotlivé úložiště (u ostatních rotací pro jednotlivé dny). Matice *matice_vsech_ulozist* obsahuje všechny média z *raw_prvek_D* a k nim nejlevnější média z *raw_prvek_z*.

```
fprintf('Nejlevnější varianta: \n')
fprintf('Disk: %s\n', cell2mat(raw_prvky_z(matice_vsech_ulozist(9,1,y),1)))
fprintf('Kapacita: %d GB\n', cell2mat(raw_prvky_z(matice_vsech_ulozist(9,1,y),2)))
fprintf('TBW: %d GB\n', matice_vsech_ulozist(10,1,y))
fprintf('Cena: %d Kč\n', matice_vsech_ulozist(8,1,y))
fprintf('Disk úložiště %d a %d: %s\n', n, n-1, cell2mat(raw_prvky_D(matice_vsech...
fprintf('Kapacita úložiště %d a %d: %d GB\n', n, n-1, cell2mat(raw_prvky_D(matice_v...
fprintf('TBW úložiště %d: %d a %d GB\n', n, n-1, matice_vsech_ulozist(10, end, y))
fprintf('Cena úložiště %d a %d: %d Kč\n', n, n-1, matice_vsech_ulozist(8, end, y))
graf_hanoi(n, typ_zalohy, matice_vsech_ulozist(:, :, y))
```

8. Vypíše se vlastnosti disků *raw_prvek_z* a *raw_prvek_D* pro tuhle nejlevnější variantu a zavolá se funkce pro vykreslení grafu (*graf_hanoi*).

9. Nastaví, se jiný typ zálohy pomocí proměnné *typ_zalohy* a pokračuje se 2. bodem, dokud se k vybrané rotaci medií neprojdou všechny její varianty (Tab.7.3).

Rotace	Typ zálohy
Hanojské věže	FB, DB
GFS	DB, IB
Rotace X úložišť	FB, DB, IB, KZ

Tab. 7.3: Možné typy záloh pro jednotlivé rotace úložišť

7.3 Funkce `textovy_vypis_nejlevnejsi_rot_x`

Je to funkce pro výpis nejlevnější varianty úložišť pro Rotaci X úložišť.

```
function textovy_vypis_nejlevnejsi_rot_x (typ_zalohy, varianta_zaloh, dny,
    vstupni_matice, pocet_tydenich_fb, D, q)
```

Vstupní proměnné jsou:

typ_zalohy – Určuje pro který typ zálohy se provede výpis – úplná záloha, inkrementální, rozdílová nebo kombinovaná záloha.

varianta_zalohy – Je proměnná, která se používá pouze u kombinované zálohy a určuje kombinaci záloh.

dny – Určuje, které dny byly použity pro zálohování.

Vstupni_matice – Jde o vybranou nejlevnější variantu dle ceny úložišť, obsahuje pro každý den TBW úložiště, cenu úložiště, název v podobě ID čísla, což je číslo řádku v ceníku, počet kusů úložišť, počet kolikrát se na dané úložiště přistupovalo.

Pocet_tydenich_fb – Je počet konců týdne, které jsou požadovány zálohovat (např. 4, když chceme týdenní zálohu a zároveň poslední 4 konce týdnů).

D – Celková velikost úložiště.

q – Určuje pravděpodobnost, že se data nezmění.

Funkce má dvě vrstvy dané větvením typu switch. V první vrstvě se dle proměnné *typ_zalohy* vybere daná větev, ve které se následně dle proměnné *dny* vybere v druhé vrstvě daný požadovaný výstup funkce – výpis parametrů rotace úložišť.

```
switch(typ_zalohy)
.
.
.
case 2 %IB
    switch(dny)
        case 1 %po-ne
            objem_zalohy = [D*(1-q) , D*(1-q) , D*(1-q), D*(1-q) , D*(1-q) , D*(1-q), D];
            E_zalohy = [D+(1*(D*(1-q))) , D+(2*(D*(1-q))) , D+(3*(D*(1-q))) , ...
            dnu_v_tydnu_bez_posledniho_dne = 6;
            cena_disku = vstupni_matice(4,:).*vstupni_matice(1,:);
            velikost_zalohy = objem_zalohy .* vstupni_matice(3,1:7);
            fprintf('\n\nPo-Ne Rotace X IB: \n');
            fprintf('Po disk ks: %d počet přístupů: %d Objem zálohy: %s GB ...
            fprintf('Út disk ks: %d počet přístupů: %d Objem zálohy: %s GB ...
            fprintf('Stř disk ks: %d počet přístupů: %d Objem zálohy: %s GB ...
            fprintf('Čt disk ks: %d počet přístupů: %d Objem zálohy: %s GB ...
```

```

fprintf('Pá disk ks: %d počet přístupů: %d Objem zálohy: %s GB ...
fprintf('So disk ks: %d počet přístupů: %d Objem zálohy: %s GB ...
for tyden = 1:pocet_tydenich_fb
    fprintf('Ne-%d disk ks: %d počet přístupů: %d Objem zálohy: %s GB ...
end;
fprintf('C: %.2f GB   E: %.2f GB\n', ...
fprintf('Celková cena úložišť: %d Kč\n\n', sum(cena_disku))

```

objem_zaloh – Obsahuje objemy záloh pro každý den.

E_zalohy – Zde jsou nadefinovány objemy obnovovacích záloh pro každý den.

dnu_v_tydnu_bez_posledniho_dne – Definuje počet dní v týdnu bez posledního dne, protože poslední den je týdenní úplná záloha, kterých může být více. Tato proměnná je pomocná proměnná pro práci s proměnou *vstupni_matice*, protože v proměnné *vstupni_matice* jsou data jednotlivých dnů uložena za sebou. Např. pro schéma zálohy Pondělí až Neděle se čtyřmi zálohami konce týdne jsou v matici data uložena ve sloupcích: Pondělí, ..., Sobota, Neděle 1, Neděle 2, ..., Neděle 4.

cena_disku – Do této proměnné se vypočítá celková cena úložišť pro jednotlivé dny, kdy ze vstupní matice *vstupni_matice* se vyberou položky cena úložiště a počet kusů.

Dále se vypočítá celkový objem dat zapsaných do úložiště pro každý den a uloží se do proměnné *velikost_zalohy*. Výpočet se provede pomocí proměnné *objem_zalohy*, a z proměnné *vstupni_matice* se vybere počet přístupů pro dané úložiště. Následně se všechna data přehledně vypíše, pro každý den zvlášť. Nakonec se vypíše pro dané schéma celkový objem záloh *C* a střední objem obnovovacích záloh *E* a celková cena všech úložišť.

7.4 Funkce `textovy_vypis_nejlevnejsi_gfs`

Funkce `textovy_vypis_nejlevnejsi_gfs` funguje na stejném principu jako `textovy_vypis_nejlevnejsi_rot_x`, ale je zde pár rozdílů.

```

function textovy_vypis_nejlevnejsi_gfs(typ_zalohy, dny, vstupni_matice, D, q)

mesic = {'Leden' 'Únor' 'Březen' 'Duben' 'Květen' 'Červen' 'Červenec' 'Srpen' ...
tydnu4 = 4;

```

Vstupní parametry jsou jen *typ_zalohy*, *dny*, *vstupni_matice*, *D* a *q*. Navíc je definována proměnná *mesic*, kde jsou uložena jména měsíců. Proměnná *tydnu4* je pomocná proměnná při práci s proměnou *vstupni_matice*. Výpis je pouze pro hodnoty 2 a 3 (Inkrementální a Rozdílová záloha) proměnné *typ_zalohy*.

7.5 Funkce `textovy_vypis_nejlevnejsi_hanoi`

Je to funkce pro výpis varianty úložišť pro rotaci Hanojské věže.

```
function textovy_vypis_nejlevnejsi_hanoi(n, D, q, vstupni_matice, typ_zalohy)
```

Vstupními proměnnými jsou:

n – Počet úložišť.

D – Celková velikost úložiště

q – Pravděpodobnost, že se data nezmění.

vstupni_matice

typ_zalohy – Může nabývat hodnot 1 nebo 3 (úplná nebo rozdílová záloha).

Rotace úložišť typu Hanojské věže je definovaná pouze pro úplnou nebo rozdílovou zálohu.

```
switch (typ_zalohy)
case 1 %FB
syms k;
C_FB = symsum(D, k, 1, 2^(n-1));
E_FB = D;
cena_disku = vstupni_matice(7,:).*vstupni_matice(8,:);
fprintf('\n\nHanoi %d úložišť FB: \n', n);
for i = 1:n
    fprintf('Úložiště %d disk ks: %d počet přístupů: %d Celkově zapsaná ...
end;
fprintf('C: %.2f GB    E: %.2f GB\n', C_FB, E_FB)
fprintf('Celková cena úložišť: %d Kč\n\n', sum(cena_disku))
```

Dle vstupních proměnných D , q a n se vypočítá celkový objem záloh C a střední objem obnovovacích záloh E (Znázorněné proměnnými C_FB a E_FB). Ze *vstupni_matice* se vypočítá celková cena úložišť (*cena_disku*) a následně se vše vypíše.

7.6 Funkce hanoi

```
function matice_vseho = hanoi( n, D, q, typ_zalohy, zivotnost_cyklu,
TBW_D, cena_disku_D, nazev_disku_D, TBW_z, cena_disku_z, nazev_disku_z)
```

Dle vstupní proměnné n , která udává počet úložišť, se vygeneruje matice *pouziti_ulozist* a načtou se do ní informace o záložních médiích (TBW, ceny, názvy).

```
for i=1:n-1
    pouziti_ulozist(1,i) = 2^i;
    if i>1
        pouziti_ulozist(2,i) = pouziti_ulozist(1,i) +1 - pouziti_ulozist(1,i-1);
    else
        pouziti_ulozist(2,i) = 2;
    end;
end;
```

Pak se pomocí cyklu `for` do prvního řádku uloží mocnina čísla dvě, které značí skok, po kterém se bude médium přepisovat. Do druhého řádku se uloží číslo – počátek použití daného úložiště.

```
for i=1:n-1
    pouziti_ulozist(3,i) = pouziti_ulozist(1,n-1)/pouziti_ulozist(1,i);
end;
```

```
pouziti_ulozist(:,n)=pouziti_ulozist(:,n-1);
pouziti_ulozist(2,n)=1;
```

Do třetího řádku se uloží počet použití média při jedné rotaci. Zkopíruje se předposlední sloupec do posledního sloupce, protože poslední dvě úložiště mají stejnou periodu opakování.

```
switch(typ_zalohy)
  case 1
    rozlozeni_dat = zeros(1,2^(n-1));
    y = length (rozlozeni_dat);
    for i=1:y
      rozlozeni_dat(1,i) = D;
    end;
  case 3
    rozlozeni_dat = zeros(1,2^(n-1));
    y = length (rozlozeni_dat);
    for i=1:y
      if i==1
        rozlozeni_dat(1,i) = D;
      else
        rozlozeni_dat(1,i) = D*(1-q^i);
      end
    end;
end;
```

Dle typu zálohy (úplná nebo rozdílová) se vygeneruje matice (tab.7.4) s velikostí zálohy pro každý den (*rozlozeni_dat*).

Objem zálohy	1000	360	488	590,4	672,3	737,9	790,3	...
--------------	------	-----	-----	-------	-------	-------	-------	-----

Tab. 7.4: Ukázkový obsah proměnné *rozlozeni_dat* s popisem řádků ve funkci *hanoi*

```
delka =length(rozlozeni_dat);
if zivotnost_cyklu == 1
  for u=1:n-1
    start= pouziti_ulozist(2,u);
    dil=pouziti_ulozist(1,u);
    for den=start:dil:delka
      if pouziti_ulozist(6,u) <= 0
        pouziti_ulozist(7,u) = pouziti_ulozist(7,u) + 1;
        pouziti_ulozist(6,u) = TBW_z;
      end;
      pouziti_ulozist(6,u) = pouziti_ulozist(6,u) - rozlozeni_dat(den);
      pouziti_ulozist(4,u)=pouziti_ulozist(4,u) + rozlozeni_dat(den);
    end;
    pouziti_ulozist(5,u) = round((pouziti_ulozist(4,u)/pouziti_ulozist(3,u)),2);
  end;
```

Podle životnosti cyklu se provede simulace jednou nebo vícekrát. Zjistí se počet dní k záloze. Načte se počátek použití úložiště (*start*), skok, po kterém se dané úložiště používá (*dil*) a pomocí cyklu `for den=start:dil:delka` se projedou všechny dny použití daného úložiště. Zjistí se, zda jeho TBW je menší nebo rovno nule, pokud ano, tak se přičte jednička do pole s počtem kusů daného úložiště a obnoví se

TBW hodnota. Dále se z pole TBW odečte objem záloh pro daný den, přičte se objem zálohy daného dne do čtvrtého řádku matice, který ukládá celkový objem zapsaných dat. Následně se provede výpočet průměrně zapsaných dat a uloží se do pátého řádku. Výsledná matice (tab.7.5) *pouziti_ulozist* se odešle jako výsledek této funkce.

Dil	2	4	8	16	32	...
Start	2	3	5	9	17	...
počet použití	256	128	64	32	16	...
celkově zapsaná data	256000	128000	64000	32000	16000	...
denní záloha	1000	1000	1000	1000	1000	...
zbývající TBW disku	104000	232000	296000	328000	344000	...
kusů disku	1	1	1	1	1	...
cena disku	3290	3290	3290	3290	3290	...
index názvu disku	46	46	46	46	46	...
TBW disku	360000	360000	360000	360000	360000	...

Tab. 7.5: Ukázkový obsah proměnné *pouziti_ulozist* s popisem řádků ve funkci *hanoi*

7.7 Funkce *rot_x_db*

Jde o funkci, která počítá parametry Rotace X úložišť pro rozdílovou zálohu.

```
function matice_vseho = rot_x_db( D, q, dny, pocet_tydenich_fb, roku, TBW_D,
    cena_disku_D, nazev_disku_D, TBW_z, cena_disku_z, nazev_disku_z, vypis_on_off)
```

Vstupní proměnné jsou:

D – Celková velikost úložiště.

q – Pravděpodobnost, že se data nezmění.

dny – Určuje které dny byly použity: Pondělí až Neděle; Pondělí až Pátek; Pouze dny Pondělí, Středa, Pátek a Neděle; nebo jen Pondělí, Středa a Pátek.

pocet_tydenich_fb – Je počet konců týdne, které jsou požadovány zálohovat (např. 4, když chceme týdenní zálohu a zároveň poslední 4 konce týdnů).

roku – Udává, na kolik let se má dané schéma simulovat (např. 20 let).

TBW_D a *TBW_z* – Obsahují výrobcem udávanou životnost v zapsaných TB. Jde o parametr používaný u SSD disků, u pevných disků HDD je v ceníku tento parametr nastaven vysoko.

cena_disku_D a *cena_disku_z* – Obsahují cenu jednoho úložiště.

nazev_disku_D a *nazev_disku_z* – Obsahují index (pořadí) úložiště v ceníku.

vypis_on_off – Umožňuje zapnout či vypnout výpis dat pro konkrétní sadu vstupních úložišť.

```

TBW_z= TBW_z*1000;
TBW_D= TBW_D*1000;
switch (dny)
    case 1 %po-ne
        dnu_v_tydnu_bez_posledniho_dne = 6;
        matice_tyden = zeros(6,dnu_v_tydnu_bez_posledniho_dne+pocet_tydenich_fb);
        matice_tyden (1,:) = 1;
        matice_tyden (2,1:dnu_v_tydnu_bez_posledniho_dne) = TBW_z;
        matice_tyden (4,1:dnu_v_tydnu_bez_posledniho_dne) = cena_disku_z;
        matice_tyden (5,1:dnu_v_tydnu_bez_posledniho_dne) = nazev_disku_z;
        matice_tyden (6,1:dnu_v_tydnu_bez_posledniho_dne) = TBW_z;
        matice_tyden (2,dnu_v_tydnu_bez_posledniho_dne+1:end) = TBW_D;
        matice_tyden (4,dnu_v_tydnu_bez_posledniho_dne+1:end) = cena_disku_D;
        matice_tyden (5,dnu_v_tydnu_bez_posledniho_dne+1:end) = nazev_disku_D;
        matice_tyden (6,dnu_v_tydnu_bez_posledniho_dne+1:end) = TBW_D;

```

Po načtení vstupních proměnných se proměnné TWB_D a TWB_z převedou z TB na GB. Dle vstupní proměnné dny se vybere požadovaná skladba záložních dnů. Vytvoří se matice $matice_tyden$ do které se zapíše vstupní data – TBW, ceny a názvy úložišť. Matice $matice_tyden$ má v tomto příkladě rozměr 6 řádků na 6+ počet neděl sloupců (co sloupec to jeden den).

```

for r=1:(52*roku)
    %po
    if matice_tyden(2,1) <= 0
        matice_tyden(1,1) = matice_tyden(1,1) + 1;
        matice_tyden(2,1) = TBW_z;
    end;
    matice_tyden(2,1) = matice_tyden(2,1) - (D*(1-q^1));
    matice_tyden(3,1)= matice_tyden(3,1) + 1;
    .
    .
    .
end;

roky = round(52/pocet_tydenich_fb);
rfor = roku*roky;
for r=1:rfor
    for tyden = 1:pocet_tydenich_fb
        %konec týdne
        if matice_tyden(2,dnu_v_tydnu_bez_posledniho_dne+tyden) <= 0
            matice_tyden(1,dnu_v_tydnu_bez_posledniho_dne+tyden) = matice_tyden(1,...
            matice_tyden(2,dnu_v_tydnu_bez_posledniho_dne+tyden) = TBW_z;
        end;
        matice_tyden(2,dnu_v_tydnu_bez_posledniho_dne+tyden) = matice_tyden(2,...
        matice_tyden(3,dnu_v_tydnu_bez_posledniho_dne+tyden) = matice_tyden(3,...
    end;
end;

```

Po načtení vstupních proměnných se spustí cyklus `for`, který proběhne $52 * \text{počet let}$, které chceme simulovat. Zjistí se, zda hodnota v poli TBW, úložiště daného dne, je nulová či záporná. Pokud je, do pole počtu kusů úložišť pro daný den se přičte jednička a pole TBW se znovu nastaví na danou hodnotu dle parametru úložiště aktuálně načtenou z ceníku. Pak se hodnota TBW pro dané úložiště daného dne sníží o objem denní zálohy a do pole počtu přístupů se přičte jednička. Tímto

způsobem se zpracují všechny dny, kromě posledního dne – konce týdne. Protože rok má 52 týdnů, vydělí se tohle číslo počtem požadovaných záloh konce týdnů (*pocet_tydenich_fb*), tím se vypočítá, kolikrát se jednotlivé úložiště konce týdne použije během roku. Následně se vynásobí počtem let, které chceme simulovat. Tohle je základ pro cyklus `for` ve kterém se nachází další cyklus `for`, který počítá s proměnnou *počet_tydenich_fb*, která udává počet konců týdnu, které chceme zálohovat. Následně se provede to stejné, co se všedními dny – otestuje se a popřípadě nastaví TBW, a zvýší se počet kusů úložiště, odečte se velikost denní zálohy a zvýší se počet přístupů.

```

if(vypis_on_off)
  fprintf('\n\nPo-Ne Rotace X DB (roků: %d):\n', roku);
  fprintf('Pořadí v ceníku disku pro Po-So: %d\n', nazev_disku_z);
  fprintf('Pořadí v ceníku disku pro Ne: %d\n', nazev_disku_D);
  fprintf('Po disk ks: %d počet přístupů: %d\n', matice_tyden(1,1), matice_tyden(3,1));
  fprintf('Út disk ks: %d počet přístupů: %d\n', matice_tyden(1,2), matice_tyden(3,2));
  fprintf('Stř disk ks: %d počet přístupů: %d\n', matice_tyden(1,3), matice_tyden(3,3));
  fprintf('Čt disk ks: %d počet přístupů: %d\n', matice_tyden(1,4), matice_tyden(3,4));
  fprintf('Pá disk ks: %d počet přístupů: %d\n', matice_tyden(1,5), matice_tyden(3,5));
  fprintf('So disk ks: %d počet přístupů: %d\n', matice_tyden(1,6), matice_tyden(3,6));
  for tyden = 1:pocet_tydenich_fb
    fprintf('Ne%d disk ks: %d počet přístupů: %d\n' ...
    end;
end;

```

Pokud má proměnná *vypis_on_off* hodnotu jedna, provede se výpis, kde je vypsán index aktuálně načtených úložišť a počet médií a přístupů pro každý den. Nakonec se výsledná matice (Tab.7.6) *matice_tyden* odešle jako výsledek této funkce.

Kusů disku	1	1	1	1	1	...
Zbývající TBW disku	349600	341280	334624	329299,19	325039,36	...
počet přístupů k disku	52	52	52	52	52	...
cena disku	3290	3290	3290	3290	3290	...
index názvu disku	46	46	46	46	46	...
TBW disku	360000	360000	360000	360000	360000	...

Tab. 7.6: Ukázkový obsah proměnné matice *matice_tyden* s popisem řádků ve funkci `rot_x_db`

7.8 Funkce `rot_x_ib` a `rot_x_fb`

Funkce `rot_x_ib` a `rot_x_fb` fungují stejně jako funkce `rot_x_db` jen jsou změněny objemy záloh pro každý den dle příslušného typu zálohy (Inkrementální nebo Úplná záloha).

7.9 Funkce `rot_x_kombinovana`

Funkce `rot_x_kombinovana` funguje podobně jako funkce `rot_x_db`, ale místo proměnné `dny` je zde použita vstupní proměnná `typ_kombinace`, která určuje typ kombinované zálohy (Tab.7.7). Funkce `rot_x_kombinovana` je naprogramována pouze pro hodnotu jedna proměnné `dny` tj. záložní dny Pondělí až Neděle.

Hodnota	po	út	stř	čt	pá	so	ne
1	DB	IB	IB	DB	IB	IB	FB
2	DB	IB	DB	IB	DB	IB	FB
3	DB	IB	IB	IB	DB	IB	FB
4	DB	IB	DB	IB	IB	IB	FB

Tab. 7.7: Hodnoty proměnné `typ_kombinace` a jejich kombinace typu záloh

7.10 Funkce `gfs_db` a `gfs_ib`

Funkce `gfs_db` a `gfs_ib` jsou obdobné funkcím `rot_x_fb`, `rot_x_db` a `rot_x_ib`. S tím rozdílem, že je zde cyklus `for` pro měsíční zálohy, ve kterém je další cyklus `for` pro týdenní zálohy.

7.11 Funkce `graf_rot_x`

Jde o funkci pro vykreslování grafů pro Rotaci X úložišť.

```
function graf_rot_x(typ_zalohy, varianta_zaloh, dny, vstupni_matice,
pocet_tydenich_fb, D, q)
```

Vstupní proměnné:

`typ_zalohy` – Určuje, pro který typ zálohy se provede výpis – Úplná záloha, Inkrementální, Rozdílová nebo Kombinovaná záloha.

`varianta_zalohy` – Je proměnná, která se používá pouze u kombinované zálohy a určuje kombinaci záloh.

`dny` – Určuje, které dny byly použity: Pondělí až Neděle; Pondělí až Pátek; Pouze

dny Pondělí, Středa, Pátek a Neděle; nebo jen Pondělí, Středa a Pátek.

Vstupni_matice – Jde o vybranou nejlevnější variantu dle ceny úložišť, obsahuje pro každý den TBW úložiště, cenu úložiště, název v podobě ID čísla, což je číslo řádku v ceníku, počet kusů úložišť, počet kolikrát se na dané úložiště přistupovalo.

Pocet_tydenich_fb – Je počet konců týdne, které jsou požadovány zálohovat (např. 4, když chceme týdenní zálohu a zároveň poslední 4 konce týdnů).

D – Celková velikost úložiště.

q – Určuje pravděpodobnost, že se data nezmění.

```
switch(typ_zalohy)
.
.
.
case 3 %DB
    switch(dny)
        case 1 %po-ne
            objem_zalohy = [D*(1-q^1) , D*(1-q^2) , D*(1-q^3) , D*(1-q^4) , D*(1-q^5) , ...
            nazev_dne = {'Pondělí (' 'Úterý (' 'Středa (' 'Čtvrtek (' 'Pátek (' ...
            nazev_grafu = 'Po-Ne Rotace X DB';
            dnu_v_tydnu_bez_posledniho_dne = 6;
            cislo_grafu = [1311 1312 1313 1314 1315 1316];
```

Dle proměnné *typ_zalohy* a proměnné *dny* se dostaneme do větve, ve které se nastaví proměnné pro daný typ zálohy a danou posloupnost zálohovaných dní. Nastaví se proměnné:

objem_zalohy – obsahující objem zálohy pro každý den

nazev_dne – jde o proměnnou obsahující název zálohovaných dnů, použita je pro popis osy grafu.

nazev_grafu – obsahuje název grafu.

cislo_grafu – obsahuje unikátní číslo pro každý graf, aby se při vykreslování výsledků více typů zálohy pro jednu skupinu dní grafy navzájem nepřekrývaly. Jde o čtyřmístné číslo, kde první číslice značí typ rotace, druhá číslice je typ zálohy, třetí číslice je skladba dní a poslední číslice je pořadí grafu.

```
ks_disku = vstupni_matice(1,:);
kc_disku = vstupni_matice(4,:).*vstupni_matice(1,:);

for i = 1:(dnu_v_tydnu_bez_posledniho_dne+pocet_tydenich_fb)
    pocet_dnu(i) = i;
    if (i >= dnu_v_tydnu_bez_posledniho_dne+1)
        if (dny == 1) || (dny == 3)
            nazev_dne(i) = {strcat('Neděle ',strcat(int2str(i-dnu_v_tydnu_bez_posledniho_dne),' ('))});
        else
            nazev_dne(i) = {strcat('Pátek ',strcat(int2str(i-dnu_v_tydnu_bez_posledniho_dne),' ('))});
        end;
        objem_zalohy(i) = [D];
    end;
end;
```

Po nastavení proměnných specifických pro každý typ zálohy se nastaví obecné společné proměnné. Proměnná *ks_disku* obsahuje počet disků pro každý den. Pro-

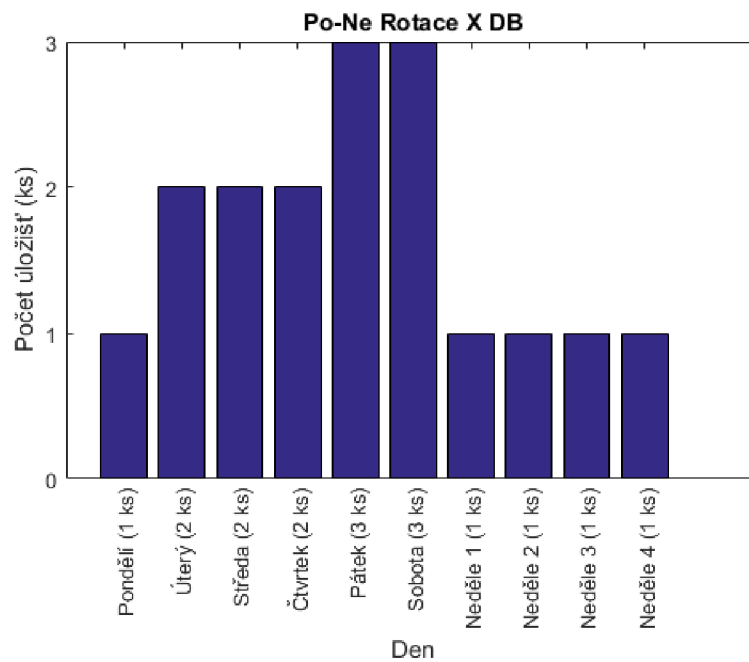
měnná *kc_disku* obsahuje celkovou cenu za úložiště pro každý den.

Dále je zde cyklus `for`, který má počet opakování roven počtu záložních dnů plus počtu zálohovaných konců týdne. Nastaví se v něm proměnná *pocet_dnu*, která slouží pro definování osy *x* v grafu. Dále je zde podmínka, pokud hodnota opakování cyklus je větší nebo rovna hodnotě konce týdne, tak se dle proměnné *dny* nastaví název dne s číselným indexem pro konce týdnů (Např. Neděle 1, Neděle 2,...)

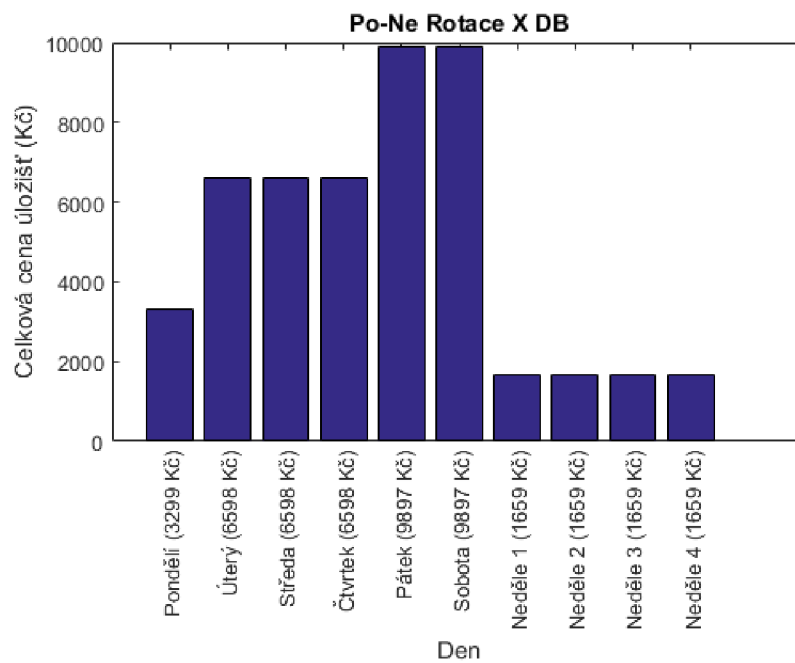
```
maximum_y_osy = max(ks_disku);
kusu_disku = strsplit(num2str(ks_disku));
popis_dne = strcat(strcat(nazev_dne,kusu_disku),' ks');

figure(cislo_grafu(1))           %okno grafu
bar(pocet_dnu, ks_disku) %vykreslí sloupcový graf
ax = gca;
ax.XTickLabel = popis_dne;      %tickům přiřadí názvy dnů
ax.YAxis.Exponent = 0;         %y osa v celých číslech
ax.YTick = (0:maximum_y_osy);  %nastavení ticku grafu po celých číslech
ax.YLim = [0, maximum_y_osy];  %nastavení limit grafu 0 až max počet úložišť
ax.XTickLabelRotation = 90;    %otočí popsek x osy o 90
title(nazev_grafu) %název grafu
xlabel('Den'); % popis osy x
ylabel('Počet úložišť (ks)'); % popis osy y
```

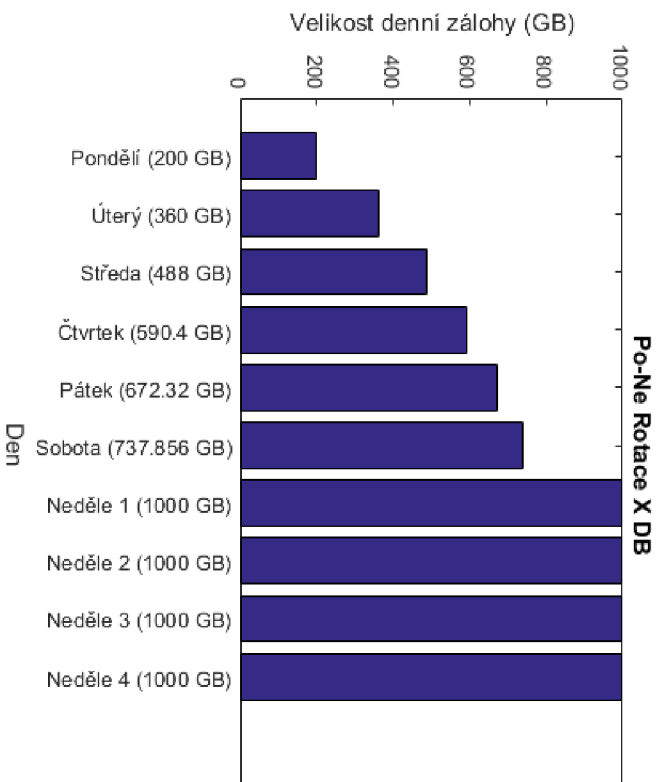
Následně se u každého grafu nastaví *popis_dne*, který obsahuje název každého dne, hodnotu a jednotku pro každý den, kterou graf zobrazuje. Graf se vykreslí a proměnnou *popis_dne*, ve funkci *ax.XTickLabel* se nastaví název bodů na ose *x*. Dále se popis osy *x* otočí o 90° funkcí *ax.XTickLabelRotation*, nastaví se popis osy *x*, *y* a název grafu. První graf (obr.7.1) zobrazuje počet kusů úložišť pro každý den, druhý graf (obr.7.2) celkovou cenu úložišť, třetí graf (obr.7.3) zobrazuje velikost denní zálohy, čtvrtý graf (obr.7.4) celkový objem zapsaných dat. Předposlední graf (obr.7.5) zobrazuje procentuální vytížení všech úložišť. Poslední graf (obr.7.6) ukazuje vytížení úložišť pouze v jednom týdnu (s jednou zálohou konce týdne).



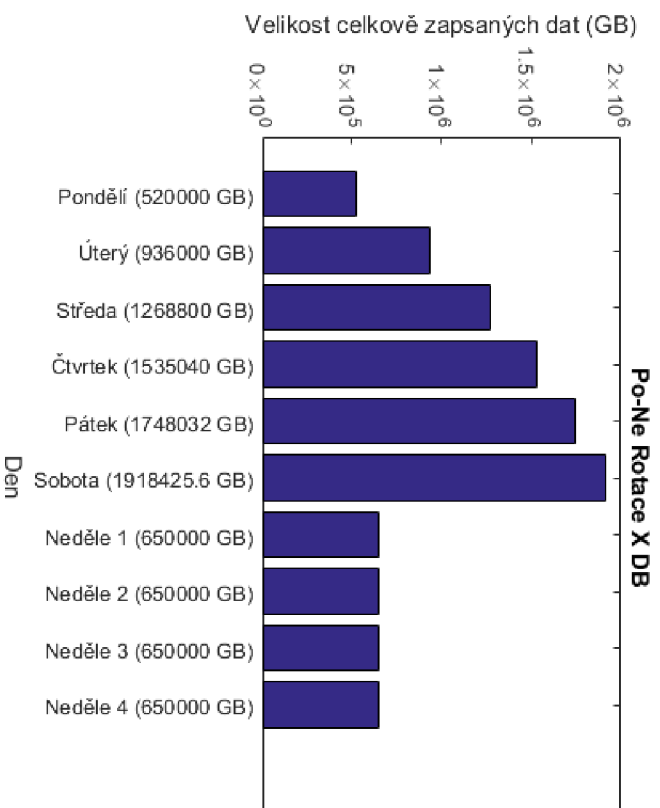
Obr. 7.1: Graf Rotace X počet kusů úložíšť.



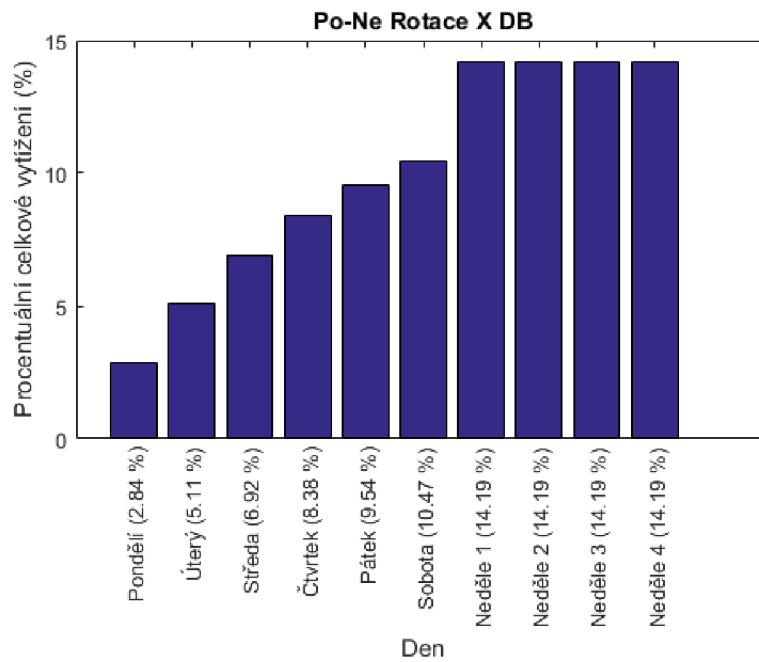
Obr. 7.2: Graf Rotace X celková cena úložíšť.



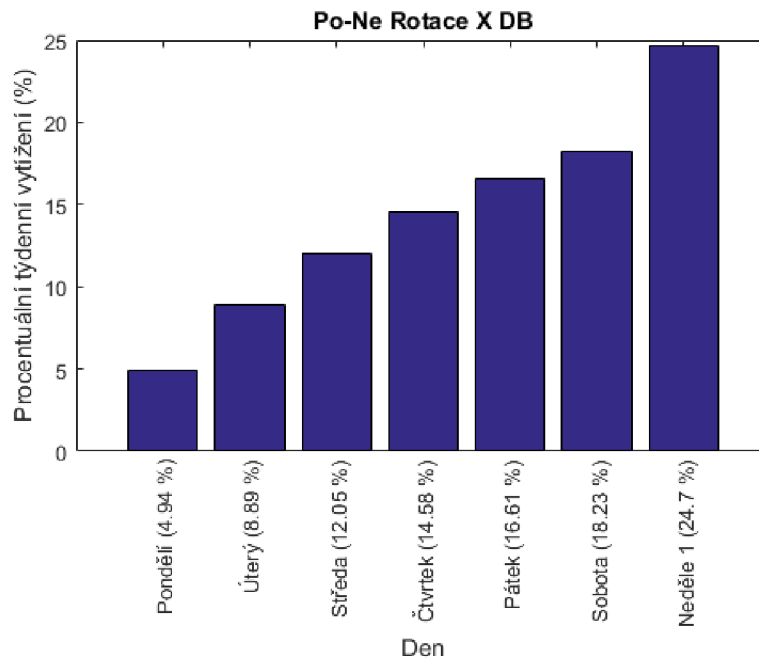
Obr. 7.3: Graf Rotace X velikost denní zálohy úložišť.



Obr. 7.4: Graf Rotace X celkově zapsaná data do úložišť.



Obr. 7.5: Graf Rotace X vytížení úložišť.



Obr. 7.6: Graf Rotace X týdenní vytížení úložišť.

7.12 Funkce graf_gfs

Funkce *graf_gfs* funguje obdobně jako funkce *graf_rot_x*. Poslední graf zobrazuje vytížení úložišť ve formě všech záložních dnů (Např. Pondělí až Sobota), jednoho konce týdne a jednoho měsíce.

7.13 Funkce graf_hanoi

Funkce *graf_hanoi* vykresluje grafy pro rotaci záloh typu Hanojské věže.

```
function graf_hanoi (n, typ_zalohy, vstupni_matice)

switch(typ_zalohy)
  case 1 %FB
    nazev_grafu = strcat(strcat('Hanoi FB ',int2str(n)), ' úložišť');
    cislo_grafu = [3131 3132 3133 3134];
  case 3 %DB
    nazev_grafu = strcat(strcat('Hanoi DB ',int2str(n)), ' úložišť');
    cislo_grafu = [3331 3332 3333 3334];
  otherwise
    end;

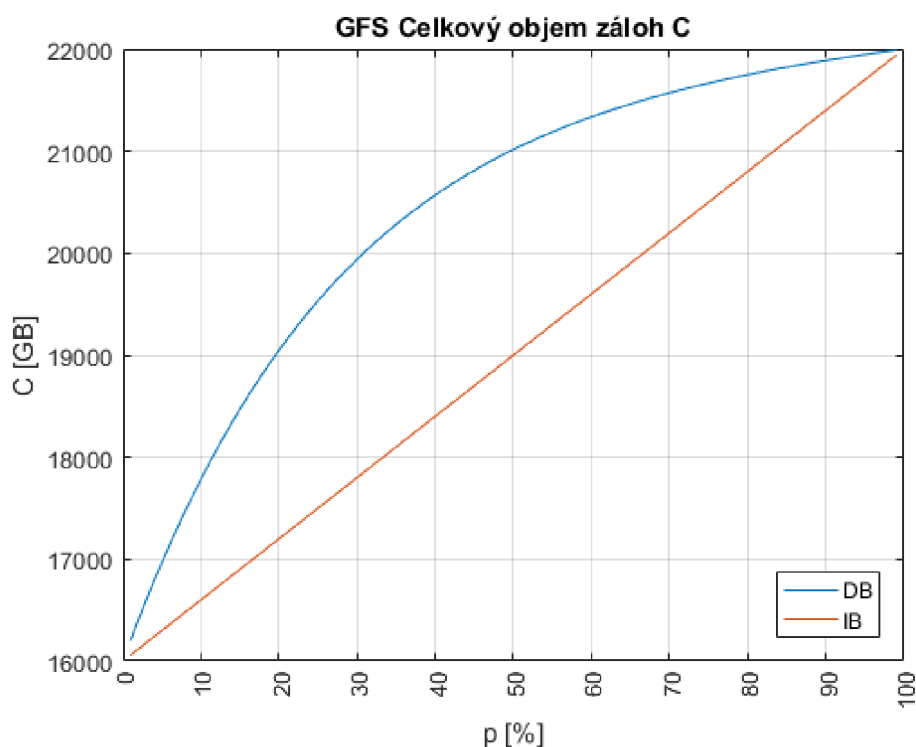
pocet_ulozist = zeros(1,n);
for i=1:n
  poradi_uloziste(i) = {'U '};
  poradi_uloziste(i)= strcat(strcat(poradi_uloziste(i),int2str(i)), ' (') ;
  pocet_ulozist(i) = i;
end;

ks_disku = vstupni_matice(7,:);
kc_disku = vstupni_matice(7,:).*vstupni_matice(8,:);
```

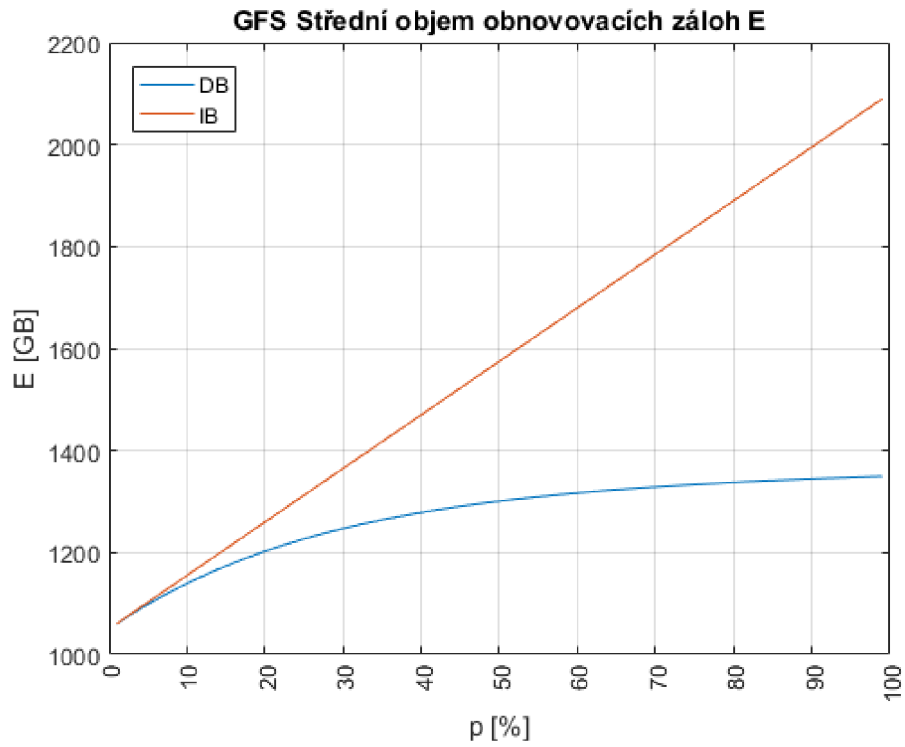
Funkce *graf_hanoi* si dle vstupní proměnné *typ_zalohy* nastaví název grafu (*nazev_grafu*) a číslování grafů (*cislo_grafu*). Dále si nastaví proměnnou *pocet_ulozist* dle vstupní proměnné *n*, protože rotace Hanoi není soudobá s kalendářem, tak na ose x není název dní, ale název úložišť, z kterých se skládá. Cyklem *for* se nastaví do proměnné *poradi_uloziste* názvy jednotlivých úložišť a nastaví proměnnou *pocet_ulozist*. Dále si ze *vstupni_matice* načte počet kusů záložních médií a jejich celkovou cenu (proměnné *ks_disku* a *kc_disku*). Následně se vykreslí pouze grafy s počtem kusů záložních médií, celkovou cenu jednotlivých úložišť, velikostí celkově zapsaných dat a graf s procentuálním vytížením jednotlivých úložišť.

8 Naměřená data

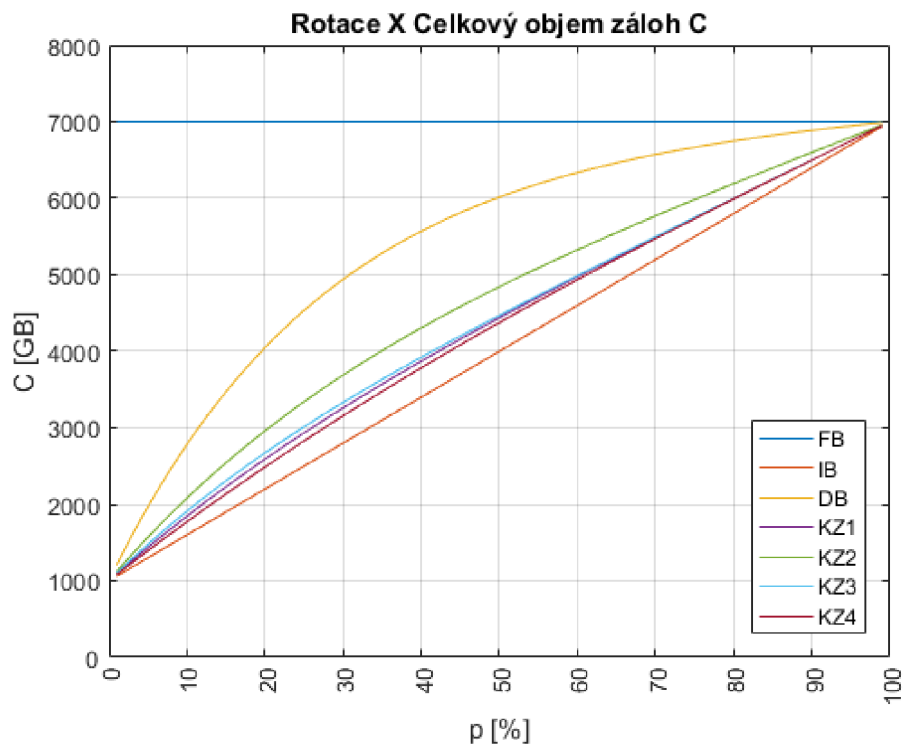
Pomocí souboru `mereni.m` se vytvořily grafy pro celkový objem záloh C a střední objem obnovovacích záloh E pro jednotlivé druhy rotací úložišť. (Obr. 8.1 – Obr. 8.6). V tabulace (Tab.8.1) jsou uvedené celkové objemy záloh C , střední objemy obnovovacích záloh E a procentuální vytížení jednotlivých úložišť pro rotaci GFS při čtyřech velikostech zátěže (z): 5%, 20%, 50% a 85%. V tabulace (Tab.8.2) jsou uvedena naměřená data pro Rotaci X úložišť. Tabulka (Tab.8.3) obsahuje celkové objemy záloh C , střední objemy obnovovacích záloh E a procentuální vytížení jednotlivých úložišť pro rotaci Hanojské věže.



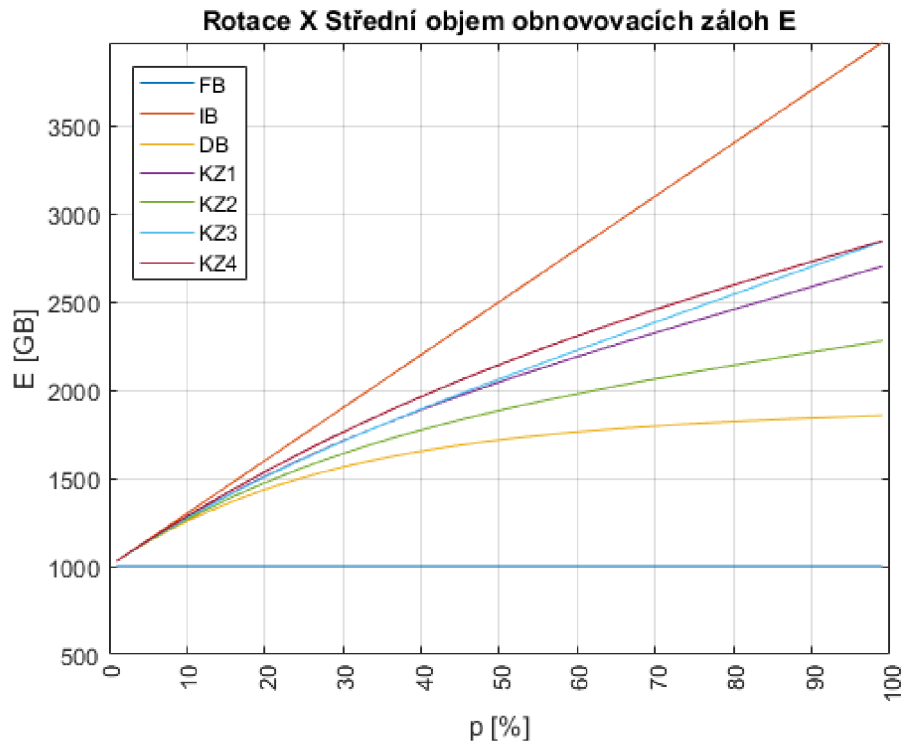
Obr. 8.1: Celkový objem záloh C pro GFS rotaci



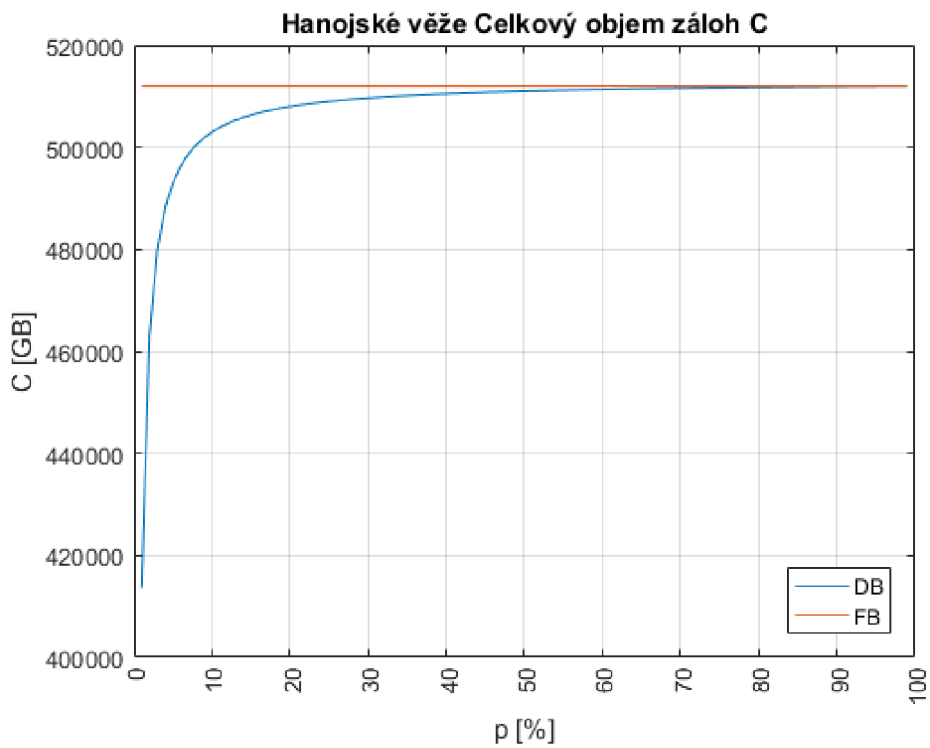
Obr. 8.2: Střední objem obnovovacích záloh E pro GFS rotaci



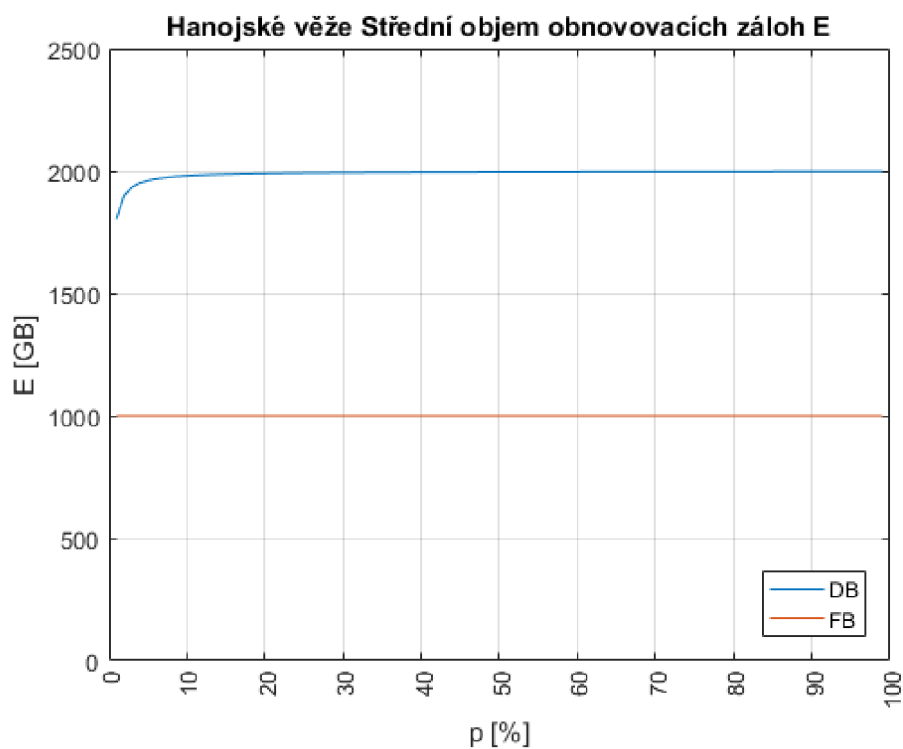
Obr. 8.3: Celkový objem záloh C pro Rotaci X úložišť



Obr. 8.4: Sřední objem obnovovacích záloh E pro Rotaci X úložišť



Obr. 8.5: Celkový objem záloh C pro rotaci Hanojské věže



Obr. 8.6: Střední objem obnovovacích záloh E pro rotaci Hanojské věže

GFS IB										
z (%)	C (GB)	E (GB)	Po (%)	Út (%)	Stř (%)	Čt (%)	Pá (%)	So (%)	Ne (%)	Měs (%)
5	16300	1002,27	2,17	2,17	2,17	2,17	2,17	2,17	43,48	43,48
20	17200	1145,45	6,25	6,25	6,25	6,25	6,25	6,25	31,25	31,25
50	19000	1431,82	10	10	10	10	10	10	20	20
85	21100	1765,91	11,97	11,97	11,97	11,97	11,97	11,97	14,08	14,08

GFS DB										
z (%)	C (GB)	E (GB)	Po (%)	Út (%)	Stř (%)	Čt (%)	Pá (%)	So (%)	Ne (%)	Měs (%)
5	16966,75	998,49	1,69	3,29	4,81	6,25	7,63	8,93	33,71	33,71
20	19048,58	1093,12	3,96	7,13	9,67	11,69	13,52	14,62	19,81	19,81
50	21015,63	1182,53	7,13	10,69	12,47	13,36	13,81	14,03	14,25	14,25
85	21823,53	1219,25	10,86	12,49	12,74	12,78	12,78	12,78	12,78	12,78

Tab. 8.1: Naměřená data pro čtyři zátěže u rotace GFS

z (%)	RotX	C (GB)	E (GB)	Po (%)	Út (%)	Stř (%)	Čt (%)	Pá (%)	So (%)	Ne (%)
-	FB	7000	1000	14,29	14,29	14,29	14,29	14,29	14,29	14,29
5	IB	1300	1150	3,85	3,85	3,85	3,85	3,85	3,85	76,92
20	IB	2200	1600	9,09	9,09	9,09	9,09	9,09	9,09	45,45
50	IB	4000	2500	12,5	12,5	12,5	12,5	12,5	12,5	25
85	IB	6100	3550	13,93	13,93	13,93	13,93	13,93	13,93	16,39
5	DB	1966,75	1138,11	2,54	4,96	7,25	9,43	11,5	13,47	50,85
20	DB	4048,58	1435,51	4,94	8,89	12,05	14,58	16,61	18,23	24,7
50	DB	6015,63	1716,52	8,31	12,47	14,55	15,58	16,1	16,36	16,62
85	DB	6823,53	1831,93	12,46	14,33	14,61	14,65	14,65	14,66	14,66
5	KB1	1435,49	1143,78	3,48	3,48	3,48	12,92	3,48	3,48	69,66
20	KB1	2590,4	1510,17	7,72	7,72	7,72	22,79	7,72	7,72	38,6
50	KB1	4437,5	2044,64	11,27	11,27	11,27	21,13	11,27	11,27	22,54
85	KB1	6249,49	2521,21	13,6	13,6	13,6	15,99	13,6	13,6	16
5	KB2	1568,84	1141,1	3,19	3,19	9,09	3,19	14,42	3,19	63,74
20	KB2	2960,32	1474,38	6,76	6,76	16,48	6,76	22,71	6,76	33,78
50	KB2	4843,75	1883,93	10,32	10,32	18,06	10,32	20	10,32	20,65
85	KB2	6396,55	2177,59	13,29	13,29	15,58	13,29	15,63	13,29	15,63
5	KB3	1476,22	1143,21	3,39	3,39	3,39	3,39	15,32	3,39	67,74
20	KB3	2672,32	1506,38	7,48	7,48	7,48	7,48	25,16	7,48	37,42
50	KB3	4468,75	2062,5	11,19	11,19	11,19	11,19	21,68	11,19	22,38
85	KB3	6249,92	2621,41	13,6	13,6	13,6	13,6	16	13,6	16
5	KB4	1392,63	1145,79	3,59	3,59	3,59	3,59	3,59	3,59	71,81
20	KB4	2488	1536	8,04	8,04	19,61	8,04	8,04	8,04	40,19
50	KB4	4375	2142,86	11,43	11,43	20	11,43	11,43	11,43	22,86
85	KB4	6246,63	2662,36	13,61	13,61	15,95	13,61	13,61	13,61	16,01

Tab. 8.2: Naměřená data pro čtyři zátěže u Rotace X úložišť

FB											
z	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	
5	50	25	12,5	6,25	3,13	1,56	0,78	0,39	0,2	0,2	(%)
85	50	25	12,5	6,25	3,13	1,56	0,78	0,39	0,2	0,2	(%)
C (GB)	512000										
E (GB)	1000										
DB											
z	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	
5	49,95	24,98	12,49	6,25	3,13	1,56	0,78	0,39	0,2	0,2	(%)
C (GB)	493000										
E (GB)	1960,94										
z	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	
85	50	25	12,5	6,25	3,13	1,56	0,78	0,39	0,2	0,2	(%)
C (GB)	511823,53										
E (GB)	1997,7										

Tab. 8.3: Naměřená data pro rotaci Hanojské věže

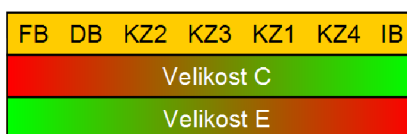
9 Volba optimálního schématu zálohy

Na základě naměřených dat z předešlé kapitoly se stanovilo schéma volby optimálního schématu zálohy. (Obr.9.3) Schéma se sestává ze tří hlavních otázek:

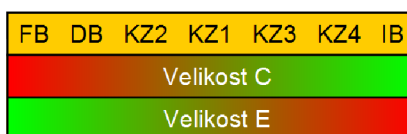
1. Požadujeme co nejzazší zálohu bez ohledu na kalendářní cyklus? Pokud ano, vybere se rotace Hanojské věže. Při zátěži $p < 10 \%$ je optimálnější rozdílové záložní schéma (DB). Při zátěži $p > 10 \%$ je optimálnější schéma úplných záloh (FB).

2. Potřebujeme jen týdenní zálohu a několik konců předešlých týdnů? Pokud ano, vybere se Rotace X úložišť. Dle objemu dat pro zálohu, cenu úložišť a zátěže si vybereme variantu schématu rotace, zda chceme rychlou obnovu dat, ale velikost záloh bude velká (Úplná záloha FB). Nebo chceme minimální velikost záloh, ale obnova dat bude trvat dlouhou dobu (Inkrementální záloha IB). Viz srovnávací obrázek (Obr.9.1) pro zátěž do 50 % a srovnávací obrázek (Obr.9.2) pro zátěž nad 50 %.

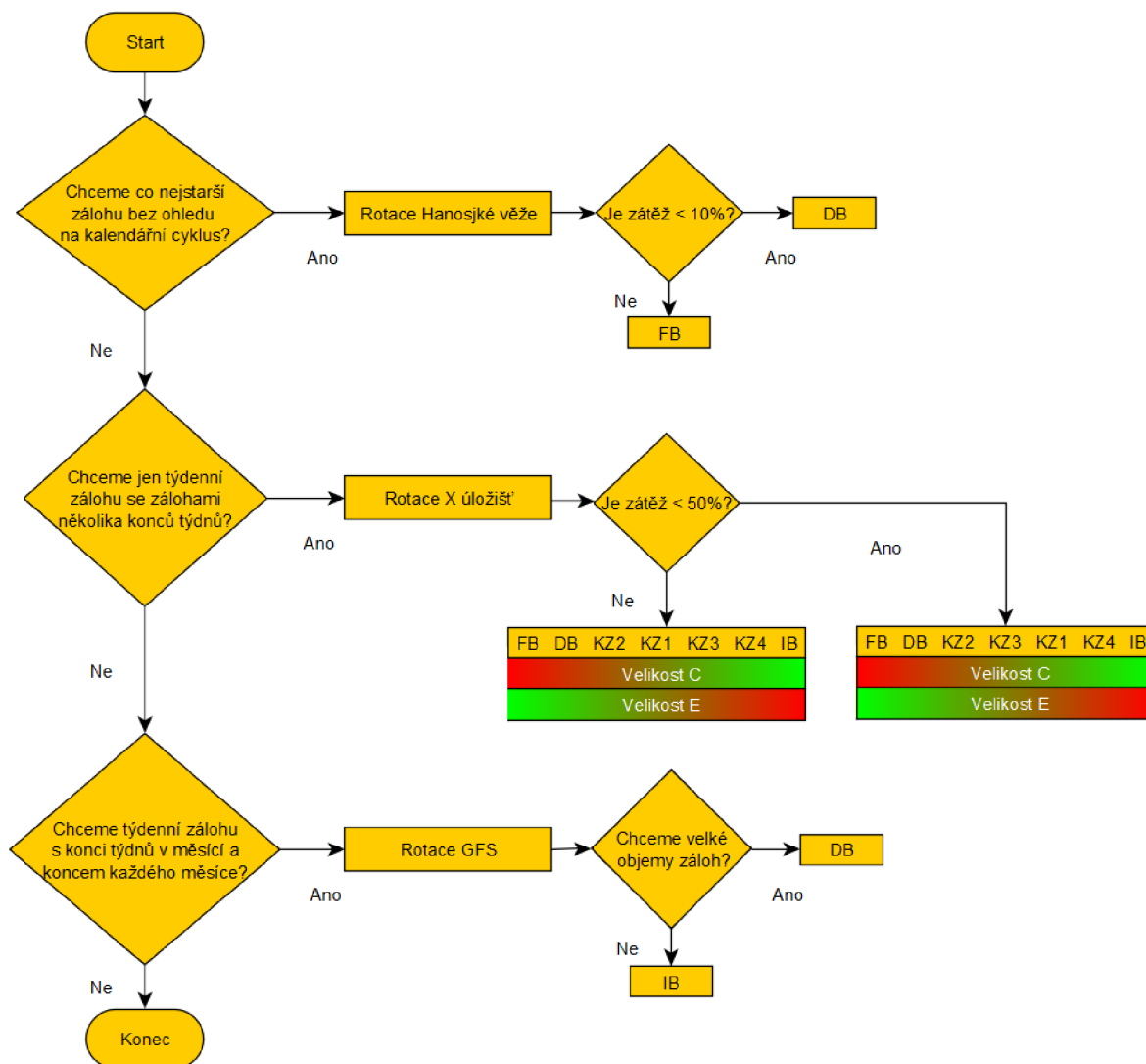
3. Chceme týdenní zálohu s konci předešlých čtyř týdnů a koncem každého měsíce v roce? Pokud ano, vybere se rotace GFS. Dle objemu dat pro zálohu, ceny úložišť si vybereme, zda chceme rychlou obnovu dat, ale velký objem záloh – Rozdílové schéma (DB) rotace GFS. Nebo chceme pomalou obnovu dat, ale menší objemy záloh – Inkrementální schéma (IB) rotace GFS.



Obr. 9.1: Závislost velikosti C a E na typu schématu zálohy, u Rotaci X úložišť, při zátěži do 50%.



Obr. 9.2: Závislost velikosti C a E na typu schématu zálohy, u Rotaci X úložišť, při zátěži nad 50%.



Obr. 9.3: Diagram výběru optimálního zálohovacího schématu

10 Závěr

Cílem diplomové práce bylo vytvořit program pro kvantitativní analýzu schémat zálohování dat a s jeho pomocí zjistit a analyzovat vlastnosti běžně používaných schémat pro různá zatížení. Na základě získaných výsledků sestavit soubor zásad pro volbu optimálního schématu zálohování dat.

V této práci jsem popsal typy záloh, úplnou, inkrementální, rozdílovou a kombinovanou zálohu. Dále jsem popsal strategie zálohování: Round Robin, 6 úložišť, GFS a strategii Hanojské věže. Každý uvedený typ zálohy a strategii jsem matematicky popsal parametry C (Celkový objem záloh) a E (Střední objem obnovovacích záloh). V šesté kapitole jsem popsal typy záložních medií. Optické disky, magnetické pásky, SSD a HDD disky. Na závěr jsem popsal typy RAID diskových polí. V sedmé kapitole jsem popsal funkci jednotlivých funkcí programu vytvořeného v prostředí Matlab. V předposlední kapitole jsou sepsána naměřená data, změřená vytvořeným programem, pro zátěže 5 %, 20 %, 50 % a 85 %, pro všechny uvedená schémata záloh. Jejich parametry C (Celkový objem záloh) a E (Střední objem obnovovacích záloh) a vytížení jednotlivých úložišť. V poslední kapitole popisují princip volby optimálního schématu zálohování dat.

Program by mohl být rozšířen o práci s RAID poli. Dále by se mohl upravit, aby se pro každý den vybralo optimální úložiště. V případě rozdílové zálohy a SSD disků by mohl každý den mít jinou kapacitu SSD disku. Pokud by se ke každému disku pořídily i informace o rychlosti čtení a zápisu (např. programem CrystalDiskMark nebo podobným), mohl by se počítat i čas potřebný pro vytvoření zálohy a čas strávený obnovou dat.

Literatura

- [1] BURDA K. *Bezpečnost informačních systémů. VUT v Brně, Brno 2013 (cs)* [online]. [cit. 3. 11. 2018]. Dostupné z URL:
<https://www.vutbr.cz/www_base/priloha.php?dpid=78973>.
- [2] BURDA K. *Matematický model zálohování a obnovy dat. Elektrotechnika, 2014, č.1.* [online]. [cit. 3. 11. 2018]. Dostupné z URL:
<<https://bit.ly/2MuXrC9>>.
- [3] *Přednáška MBIS 10e.pdf.* [cit. 3. 11. 2018].
- [4] *An introduction to data backup tape rotation schemes* [online]. [cit. 3. 11. 2018]. Dostupné z URL:
<<https://searchdatabackup.techtarget.com/tip/An-introduction-to-data-backup-tape-rotation-schemes>>.
- [5] *Wikipedie: Otevřená encyklopedie: Optický disk c2018* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://bit.ly/2LjHz6Q>>.
- [6] *Wikipedie: Otevřená encyklopedie: DVD c2018* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://en.wikipedia.org/w/index.php?title=DVD&oldid=873348465>>.
- [7] *Wikipedie: Otevřená encyklopedie: DVD-RAM c2018* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://en.wikipedia.org/w/index.php?title=DVD-RAM&oldid=839505386>>.
- [8] *Wikipedie: Otevřená encyklopedie: Blue-ray c2018* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://en.wikipedia.org/w/index.php?title=Blu-ray&oldid=873348690>>.
- [9] *Optická média žijí: archivační M-Disk zapíšete na DVD vypalovače* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://bit.ly/2Es1bFh>>.
- [10] *Wikipedie: Otevřená encyklopedie: Magnetická páska c2018* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://bit.ly/2SMIyil>>.

- [11] *Wikipedie: Otevřená encyklopedie: Linear Tape-Open c2018* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<https://en.wikipedia.org/w/index.php?title=Linear_Tape-Open&oldid=860735661>.
- [12] *Wikipedie: Otevřená encyklopedie: Video c2018* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://en.wikipedia.org/w/index.php?title=Video&oldid=872428077>>.
- [13] *What is LTO technology* [online]. [cit. 13. 12. 2018]. Dostupné z URL:
<<https://www.lto.org/technology/what-is-lto-technology/>>.
- [14] *Wikipedie: Otevřená encyklopedie: Pevný disk c2019* [online]. [cit. 13. 05. 2019]. Dostupné z URL:
<https://cs.wikipedia.org/w/index.php?title=Pevný_disk&oldid=17236947>.
- [15] *Wikipedie: Otevřená encyklopedie: Solid-state drive c2019* [online]. [cit. 13. 05. 2019]. Dostupné z URL:
<https://cs.wikipedia.org/w/index.php?title=Solid-state_drive&oldid=17053820>.
- [16] *RAID (redundant array of independent disks)* [online]. [cit. 9. 12. 2018]. Dostupné z URL:
<<https://searchstorage.techtarget.com/definition/RAID>>.
- [17] *Wikipedie: Otevřená encyklopedie: RAID c2018* [online]. [cit. 9. 12. 2018]. Dostupné z URL:
<<https://en.wikipedia.org/w/index.php?title=RAID&oldid=872025308>>.
- [18] *Wikipedie: Otevřená encyklopedie: Standard RAID levels c2018* [online]. [cit. 9. 12. 2018]. Dostupné z URL:
<https://en.wikipedia.org/w/index.php?title=Standard_RAID_levels&oldid=872876666>.
- [19] *Wikipedie: Otevřená encyklopedie: Nested RAID levels c2018* [online]. [cit. 9. 12. 2018]. Dostupné z URL:
<https://en.wikipedia.org/w/index.php?title=Nested_RAID_levels&oldid=859357813>.

Seznam symbolů, veličin a zkratek

BD	Blue-ray Disc – typ optického media
CD	Compact Disc – typ optického media
DB	Diferential Backup – rozdílová záloha
DVD	Digital Video Disc / Digital Versatile Disc – typ optického media
FB	Full Backup – úplná záloha
GB	GigaByte – jednotka digitální informace
HDD	Hard Disc Drive – typ paměťového media
HP	Hewlett-Packard – název technologické společnosti
IB	Incremental Backup – inkrementální záloha
IBM	International Business Machines – název technologické společnosti
KZ	Kombinovaná Záloha
LTO	Linear Tape Open – typ magnetické pásky
M.2	typ interního slotu počítače
MB	MegaByte – jednotka digitální informace
M-disc	Millennial Disc – podtyp optického media založeného na DVD či BD.
mm	milimetr – jednotka délky
ms	milisekunda – jednotka času
PCIe	Peripheral Component Interconnect Express – typ interní sběrnice v počítači
RAID	Redundant Array of Independent Disks – pole pevných disků tvořících úložiště
RAM	Random Access Memory – paměť s náhodným přístupem
ROM	Read Only Memory – paměť pouze pro čtení
RW	Read Write – u optických disků značí možnost čtení i zápisu
SATA	Serial Advanced Technology Attachment – typ interní sběrnice v počítači
SSD	Solid State Disk – typ paměťového media
TB	TeraByte – jednotka digitální informace
TBW	TeraByte Written – počet zapsatelných TeraBytů, udává životnost SSD disku
μs	mikrosekunda – jednotka času
USB	Universal Serial Bus – typ externí sběrnice v počítači

Seznam příloh

A Ukázka výpisu z programu	64
B Obsah přiloženého CD	65

A Ukázka výpisu z programu

```
Celkový objem dat: 1000      Objem denní změny dat: 200
Typ zálohy: Týdenní (Rotace X úložišť) Záložní dny: Po-Ne Životnost roků: 50 Počet zálohovaných konců týdnu: 4
Úložiště SSD i HDD

Po-Ne Rotace X IB:
Po disk ks: 3 počet přístupů: 2600 Objem zálohy: 200 GB Celkově zapsaná data: 520000 GB Celková cena: 2517 Kč
Út disk ks: 3 počet přístupů: 2600 Objem zálohy: 200 GB Celkově zapsaná data: 520000 GB Celková cena: 2517 Kč
Stř disk ks: 3 počet přístupů: 2600 Objem zálohy: 200 GB Celkově zapsaná data: 520000 GB Celková cena: 2517 Kč
Čt disk ks: 3 počet přístupů: 2600 Objem zálohy: 200 GB Celkově zapsaná data: 520000 GB Celková cena: 2517 Kč
Pá disk ks: 3 počet přístupů: 2600 Objem zálohy: 200 GB Celkově zapsaná data: 520000 GB Celková cena: 2517 Kč
So disk ks: 3 počet přístupů: 2600 Objem zálohy: 200 GB Celkově zapsaná data: 520000 GB Celková cena: 2517 Kč
Ne-1 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
Ne-2 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
Ne-3 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
Ne-4 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
C: 2200.00 GB   E: 1600.00 GB
Celková cena úložišť: 21738 Kč

Nejlevnější varianta:
Disk: Patriot Burst
Kapacita: 240 GB
TBW: 180000 GB
Cena: 839 Kč
Disk FB: Seagate BarraCuda 2TB
Kapacita FB: 1862 GB
TBW FB: 1000000000000 GB
Cena FB: 1659 Kč

Po-Ne Rotace X DB:
Po disk ks: 1 počet přístupů: 2600 Objem zálohy: 200 GB Celkově zapsaná data: 520000 GB Celková cena: 3299 Kč
Út disk ks: 2 počet přístupů: 2600 Objem zálohy: 360 GB Celkově zapsaná data: 936000 GB Celková cena: 6598 Kč
Stř disk ks: 2 počet přístupů: 2600 Objem zálohy: 488 GB Celkově zapsaná data: 1268800 GB Celková cena: 6598 Kč
Čt disk ks: 2 počet přístupů: 2600 Objem zálohy: 590.4 GB Celkově zapsaná data: 1535040 GB Celková cena: 6598 Kč
Pá disk ks: 3 počet přístupů: 2600 Objem zálohy: 672.32 GB Celkově zapsaná data: 1748032 GB Celková cena: 9897 Kč
So disk ks: 3 počet přístupů: 2600 Objem zálohy: 737.856 GB Celkově zapsaná data: 1918425.6 GB Celková cena: 9897 Kč
Ne-1 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
Ne-2 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
Ne-3 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
Ne-4 disk ks: 1 počet přístupů: 650 Objem zálohy: 1000 GB Celkově zapsaná data: 650000 GB Celková cena: 1659 Kč
C: 4048.58 GB   E: 1435.51 GB
Celková cena úložišť: 49523 Kč

Nejlevnější varianta:
Disk: ADATA Ultimate SU800
Kapacita: 1000 GB
TBW: 800000 GB
Cena: 3299 Kč
Disk FB: Seagate BarraCuda 2TB
Kapacita FB: 1862 GB
TBW FB: 1000000000000 GB
Cena FB: 1659 Kč
```

Obr. A.1: Ukázka výpisu programu

B Obsah příloženého CD

```
/ ..... kořenový adresář příloženého CD
├── matlab ..... všechny soubory programu
│   ├── a_program.m ..... hlavní soubor
│   ├── CENIK_HDD.xlsx
│   ├── CENIK_SSD.xlsx
│   ├── gfs_db.m
│   ├── gfs_ib.m
│   ├── graf_gfs.m
│   ├── graf_hanoi.m
│   ├── graf_rot_x.m
│   ├── hanoi.m
│   ├── mereni.m ..... samostatný soubor pro vykreslení grafů do 8. kapitoly
│   ├── rot_x_db.m
│   ├── rot_x_fb.m
│   ├── rot_x_ib.m
│   ├── rot_x_kombinovana.m
│   ├── textovy_vypis_nejlevnejsi_gfs.m
│   ├── textovy_vypis_nejlevnejsi_hanoi.m
│   ├── textovy_vypis_nejlevnejsi_rot_x.m
└── diplomova_prace.pdf
```