

Šifrování a GeoGebra

Jakub Fryštacký

Bakalářská práce
2014



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

akademický rok: 2013/2014

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jakub Fryštický**
Osobní číslo: **A11011**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Bezpečnostní technologie, systémy a management**
Forma studia: **prezenční**

Téma práce: **Šifrování a Geogebra**

Zásady pro vypracování:

1. Shrňte stručně historii šifrování dat.
2. Vložte teorii šifrování přenosu číselných posloupností.
3. Seznamte se s programem GeoGebra.
4. Ukažte možnosti práce s maticemi v programu Geogebra.
5. Vytvořte v programu Geogebra nástroj pro šifrování a dešifrování číselných posloupností.
6. Uvedte konkrétní příklady použití nástroje pro šifrování a dešifrování číselných posloupností v programu Geogebra.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **VONDRUŠKA, Pavel. Kryptologie, šifrování a tajná písma: kryptologie. 1. vyd. Praha: Albatros, edice: OKO, 2006, 340 s. ISBN 80-00-01888-8.**
2. **GERGELITSOVÁ, Šárka. Počítač ve výuce nejen geometrie Průvodce GeoGebrou. Generation Europe, o.s., 2012, 256 s. ISBN 978-80-904974-3-6.**
3. **BALEE, Maram, K LAKSHMANA, Rao, Y RAMESH, Kumar. Encryption and Decryption Algorithm using 2-D Matrices. International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 4, 2013, 352-356 s. ISSN 2277 128X.**
4. **VELLAJKANNAN, B.; Dr. MOHAN, V.; GNANARAJ. V. A Note on the Application of Quadratic Forms in Coding Theory with a Note on Security, Int. J. Comp.Tech. Appl, Vol 1 (1), 2010, 78-87 s.**
5. **ZELENKA, Josef. Ochrana dat: kryptologie. Vyd. 1. Hradec Králové: Gaudeamus, 2003, 198 s. ISBN 80-704-1737-4.**

Vedoucí bakalářské práce: **Mgr. Lubomír Sedláček, Ph.D.**

Ústav matematiky

Konzultant: **RNDr. Jana Volná, Ph.D.**

Ústav matematiky

Datum zadání bakalářské práce: **7. března 2014**

Termín odevzdání bakalářské práce: **10. června 2014**

Ve Zlíně dne 7. března 2014



prof. Ing. Vladimír Vašek, CSc.
děkan



doc. Mgr. Milan Adámek, Ph.D.
ředitel ústavu

ABSTRAKT

Bakalářská práce se zabývá praktickou ukázkou šifrování číselných posloupností. Jedná se o vytvoření ukázkových nástrojů pro práci s maticemi a vytvoření nástroje pro šifrování a dešifrování. Programování všech jednotlivých částí je realizováno pomocí programu GeoGebra.

V teoretické části naleznete základní informace ohledně šifrování dat, přenosu číselných posloupností a stručné seznámení s programem GeoGebra. Praktická část je zaměřena na samotnou tvorbu aplikací práce s maticemi a nástroje pro šifrování a dešifrování krok za krokem.

Klíčová slova: šifrování, dešifrování, GeoGebra, JavaScript, matice, číselná posloupnost

ABSTRACT

The bachelor thesis deals with the practical example of encryption numerical sequences. This is a demonstration of creating tools for working with matrices and the creation of tools for encryption and decryption. Programming of the individual components is realized by GeoGebra.

In the theoretical section are provides basic information about data encryption, transmission, numerical sequences and a brief introduction to GeoGebra. The practical part is focused on creating applications work with matrices and tools for encryption and decryption step by step.

Keywords: encryption, decryption, GeoGebra, JavaScript, matrix, numeric sequence

Rád bych poděkoval vedoucímu mé bakalářské práce panu Mgr. Lubomíru Sedláčkovi, Ph.D. a konzultantce paní RNDr. Janě Volné, Ph.D. za jejich odborné vedení, konstruktivní rady, připomínky a konzultace, které mi při tvorbě bakalářské práce poskytovali. Chtěl bych také poděkovat rodině za podporu během studia a přátelům za podporu, pomoc a rady při tvorbě této práce.

„Bezpečnost spočívá výhradně v klíči.“

(Auguste Kerckhoff)

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 TERMINOLOGIE	11
2 HISTORIE ŠIFROVÁNÍ.....	12
2.1 PRVOPOČÁTKY	12
2.1.1 Hieroglyfy	12
2.1.2 Klínové písmo	12
2.1.3 Atbaš.....	13
2.1.4 Skytala.....	13
2.2 STEGANOGRAFIE	14
2.2.1 Polybiův čtverec	15
2.2.2 Caesarova šifra	15
2.2.3 Starověká Indie.....	16
2.2.4 Frekvenční analýza.....	16
2.3 ŠIFROVÁNÍ V ČECHÁCH.....	17
2.3.1 Složitější substituce	17
2.4 ŠIFROVÁNÍ PODLE HESLA.....	18
2.4.1 Vigenèrova šifra	18
2.4.1.1 Rozluštění Vigenèrovy šifry	19
2.4.2 Marie Stuartovna	19
2.4.3 Baconova šifra.....	20
2.4.4 Jeffersonův váleček.....	20
2.4.5 Playfair	20
2.5 KERCKHOFFOVY PRINCIPY	21
2.6 GOLD-BUG	21
2.7 LEGENDA O BEALOVĚ POKLADU	22
2.8 ŠIFROVÁNÍ VEŘEJNOSTI.....	22
2.9 1. SVĚTOVÁ VÁLKA	22
2.9.1 Mata Hari.....	22
2.9.2 Room 40.....	23
2.10 2. SVĚTOVÁ VÁLKA	23
2.10.1 Sigaba, Typex, Purple, Enigma.....	23
2.11 MODERNÍ KRYPTOGRAFIE.....	23
2.11.1 Symetrické šifrování	24
2.11.2 Asymetrické šifrování	24
2.11.3 Kvantová kryptografie.....	24
3 PŘENOS ČÍSELNÝCH POSLOUPNOSTÍ.....	25
3.1 HILLOVA ŠIFRA	25
3.2 TEORIE KODOVÁNÍ	25
3.2.1 Šifrování	25
3.2.2 Dešifrování	26
4 GEOGEBRA.....	27

4.1	PROSTŘEDÍ	27
4.2	OVLÁDÁNÍ.....	28
4.3	GRAFICKÝ VSTUP	28
4.4	PŘÍKAZOVÝ ŘÁDEK	28
4.4.1	Přímý vstup	29
II	PRAKTICKÁ ČÁST	30
5	PRÁCE S MATICEMI	31
5.1	TVORBA MATIC	31
5.2	SOUČET MATIC	32
5.3	SOUČIN MATIC.....	33
5.4	DETERMINANT MATICE.....	34
5.5	INVERZNÍ MATICE.....	36
5.6	TRANSPONOVANÁ MATICE	37
6	NÁSTROJ PRO ŠIFROVÁNÍ TEXTU.....	38
7	NÁSTROJ PRO DEŠIFROVÁNÍ TEXTU	42
8	ZDROJOVÝ KÓD JEDNOTLIVÝCH TLAČÍTEK	45
8.1	TLAČÍTKO DOPLNĚNÍ	45
8.2	TLAČÍTKO MATICE.....	46
8.3	TLAČÍTKO ÚPRAVA KLÍČE	46
8.4	TLAČÍTKO MATICE KLÍČ.....	47
8.5	TLAČÍTKO ŠIFRUJ	47
8.6	TLAČÍTKO ŠIFROVANÝ TEXT	48
8.7	TLAČÍTKO DEŠIFRUJ.....	48
8.8	TLAČÍTKO OTEVŘENÝ TEXT	48
9	VYUŽITÍ APLIKACÍ.....	49
	ZÁVĚR	50
	SEZNAM POUŽITÉ LITERATURY.....	52
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	54
	SEZNAM OBRÁZKŮ	55
	SEZNAM TABULEK.....	57
	SEZNAM PŘÍLOH.....	58

ÚVOD

S šifrováním se každý z nás setkává denně, jen si to mnozí z nás ani neuvědomují. Ať už se jedná o práci s počítačem v podobě přihlašování do osobního nebo pracovního emailu, nebo třeba i přihlášení do internetového bankovníctví. Šifrování také potkáváme při fyzické platbě bezkontaktní platební kartou v obchodech. V těchto a mnoha dalších případech pracujeme s citlivými, důvěrnými osobními daty, které je třeba chránit. A to nejlépe šifrováním.

Téma bakalářské práce jsem si zvolil hlavně proto, že oblast kryptografie mi přijde nesmírně zajímavá a také proto, že jsem si chtěl vyzkoušet práci i s jiným programem než je Matlab nebo Wolfram Mathematica, se kterými jsem se setkal v průběhu studia. Proto jsem vybral spojení šifrování a pro mě zcela nového programu GeoGebra.

Cílem této bakalářské práce je v rámci teoretické části nejdříve na úvod čtenáře lehce zasvětit do základní terminologie, která se týká šifrování, dále jej stručně seznámit s historií šifrování, s teorií šifrování přenosu číselných posloupností a v neposlední řadě jej také stručně seznámit se základním popisem a ovládáním programu GeoGebra. Zde jen stručně, protože ovládání a seznámení s programem GeoGebra je opravdu téma na samostatnou práci. V praktické části je nejdříve cílem vytvořit jednoduché a názorné programy určené pro práci s maticemi v programu GeoGebra. Bude se jednat o jednoduché operace s maticemi jako je sčítání matic, součin matic, výpočet inverzní matice, výpočet transponované matice a výpočet determinantu matice. Vše bude předvedeno velmi jednoduše a přehledně. Druhá část praktické části této bakalářské práce bude tvořena návrhem a realizací nástroje pro šifrování a dešifrování číselných posloupností v GeoGebře. Při šifrování se jako vstupní hodnota bude zadávat text a výstupem bude číselný vektor. Při dešifrování bude vstupem číselný vektor a výstupem text. Klíč bude uživatel zadávat jako text. Důraz je kladen na jednoduchou obsluhu jak při šifrování, tak i při dešifrování. V poslední části praktické části bude zhodnoceno praktické využití všech naprogramovaných výstupů.

I. TEORETICKÁ ČÁST

1 TERMINOLOGIE

Na úvod této práce je třeba si stručně definovat základní pojmy týkající se kryptologie. Kryptologie je tvořena dvěma vědními obory, jsou to kryptologie a kryptoanalýza. Kryptografie - věda o šifrování a dešifrování dat za pomoci matematických metod. Kryptoanalýza - věda zabývající se metodami zjištění původní informace ze zašifrované bez znalosti klíče.

Při šifrování a dešifrování se pak setkáváme s pojmy, jako jsou: otevřený text (zpráva) = text, řetězec znaků nebo písmen, který chceme přenášet. Šifrovaný text (kryptogram) = zašifrovaná zpráva. Klíč: Exkluzivní informace pomocí níž můžeme zašifrovat zprávu. Kryptografický systém = pětice $\{M, C, K, E, D\}$, kde M – konečná množina otevřených textů (prostor textu). C – konečná množina šifer (prostor šifer). K – konečná množina klíčů. E – množina šifrovacích funkcí (pravidel, šifrovacích algoritmů). D – množina dešifrovacích funkcí (pravidel dešifrovacích algoritmů). Ideální kryptografický systém podporuje následující bezpečnostní vlastnosti: utajení, autentizace a integrita dat a nepopíratelnost. „Silné“ šifry jsou takové šifry, které „dokáží“ odolat všem kryptoanalytickým metodám, kromě útoku „hrubou silou“. Slovo „dokáží“ vyjadřuje fakt, že bezpečnost systému je založena na založení silného matematického problému. Obranou proti útoku hrubou silou je obecně zvětšení délky klíče tak, aby se útok stal v reálném čase nerealizovatelný nebo výpočetně neproveditelný. Obecně jsou za „silné“ šifry považovány kvalitní symetrické šifry s délkou klíče nad 70 bitů a u asymetrických šifer ty RSA s délkou klíče nad 700 bitů. Na závěr je třeba si definovat rozdíly mezi kódem a šifrou. U kódu se nahradí slova či fráze jiným slovem, číslem či symbolem. Je třeba volit slova bez logické souvislosti. Příklad: Krycí jména tajných agentů. Výhodou je, že se nejedná o algoritmus. Nevýhodou je, že kód je nepružný a ve větším rozsahu je nutná kódová kniha. U šifry dochází k nahrazení písmen (resp. fragmentů informace) jinými písmeny za daných pravidel. Lze ji popsat pomocí algoritmů a klíče. Je k dispozici neomezená komunikace (je pružná). Další výhodou je, že na rozluštění stačí znalost klíče a algoritmus. Ale velkou nevýhodou je možnost jakéhokoliv útoku na šifru a možnost jejího prolomení. [16, s. 8-26]

2 HISTORIE ŠIFROVÁNÍ

2.1 Prvopočátky

Za prvními pokusy o šifrování textu je třeba se vypravit několik tisíc nazpět. V této době se nejednalo o klasické šifrování, jak na něj nahlížíme dnes. Než bylo „písmo“ rozluštno, jednalo se o „šifru“.

2.1.1 Hieroglyfy

Egyptské hieroglyfy jsou (obrázkové) písmo, které bylo v Egyptě nejdéle používané a je nejstarší. První se objevují ve formě krátkých hesel na kamenech a střepech. Pocházejí z doby Předdynastické, to je období přibližně 3000 let př. n. l.

Poslední byly nalezeny v chrámových nápisech na ostrově Filé a pocházejí z doby kolem roku 394. [2]

Zprvu sloužily pro záznam textů na různé materiály, později již jen pro náboženské účely, kde se vytesávaly do zdí chrámů. Jednotlivé znaky se vyjadřují svou dokonalou výstižností a někdy i barevností. Jedná se o dokonale propracovaný abecední systém. Psaní bylo dovoleno zleva doprava, ale i naopak. Egypťané běžně používali zhruba kolem 750 znaků, ale doložená existence je zhruba 6000.



Obr. 1. Hieroglyfy [3]

2.1.2 Klínové písmo

V Mezopotámii na přelomu 3. a 4. tisíciletí př. n. l. se vyvíjel národ Sumerů. Používali klínové písmo. Písmo je abstraktní, geometrické a strohé. Původ písma je zcela jistě v obrázkovém písmu. Během své existence zaznamenalo významný vývoj. Písmo tvořilo několik set znaků. Sumerové používali k psaní hliněné tabulky, na které psali písáři

rákosovým pisátkem. Konec byl seříznut do tvaru trojúhelníka. Vznikaly tak drobné vrypy klínového tvaru (odtud odvozeno “klínové písmo“). Později Asyřané klínové písmo značně zjednodušili, používali pouze 300-500 znaků. [4]



Obr. 2. Klínové písmo [4]

2.1.3 Atbaš

Šifra Atbaš, zde již skutečně můžeme mluvit o klasické šifře. Jedná se o monoalfabetickou (jednouchou) šifru, kterou vynalezli a využívali Hebrejci. Zhruba 500 let př. n. l.

Princip šifry prozrazuje už i samotný název, neboť písmena A-T-B-Š jsou postupně prvním (alef), posledním (thav), druhým (bet) a předposledním (šin) písmenem hebrejské abecedy.

Princip šifrování spočívá v tom, že vezmeme písmeno, které chceme zašifrovat. Určíme jeho vzdálenost od počátku abecedy. Toto písmeno pak nahradíme písmenem se stejnou vzdáleností od konce abecedy. Tato šifra je snadná na prolomení. [16, s. 38-39,197]

Tab. 1. Otevřená a šifrovaná abeceda šifry 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

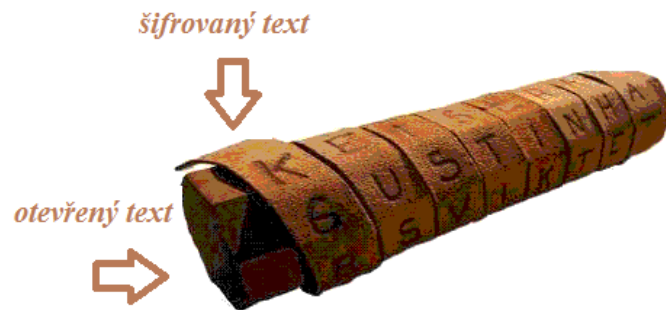
Příklad: otevířený text: SIFROVANI A GEOGEBRA

šifrovaný text: HRUILEZMRZTVLTVYIZ

2.1.4 Skytala

Spartská Skytala je nejstarší kryptologické zařízení určené pro šifrování zpráv. Základem byly dvě hole, jedna pro odesílatele a druhá pro příjemné. Hole měly stejnou šířku. Šířka hole = symetrický klíč). Šifrování zprávy probíhalo tak, že se na první hůl namotal papyrus nebo pergament. Napsala se zpráva. Svitek se sejmul a posel doručil zašifrovanou zprávu.

Příjemce poté na svou hůl namotal zprávu a tu si přečetl. V dnešní době se jedná o transpoziční šifru, jedná se pouze o permutaci míst písmen.



Obr. 3. Spartská Skytala [5]

V dnešním moderním pojetí by dešifrování mohlo probíhat následovně. Mějme následující šifrovanou zprávu.

DAENEHETJILONLECAJIEVISAKSEKVKJKZOAUA AVSTBNCOTEA

Pro dešifrování použijeme matici o n -řádcích, kde n je šířka Skytaly, zde $n = 6$. Poté už jen stačí zapisovat šifrovaný text po sloupcích. Otevřený text lze pak přečíst po řádcích.

D	E	N	I	K	J	A	N
A	T	L	E	S	K	A	C
E	J	E	V	E	Z	V	O
N	I	C	I	K	O	S	T
E	L	A	S	V	A	T	E
H	O	J	A	K	U	B	A

Na příkladu je tedy jasné, že je správně zvolená šířka Skytaly. Šířku jde metodou pokusu a omylu velmi rychle zjistit. Proto je tato šifra snadno prolomitelná. [5]

2.2 Steganografie

Slovo steganografie pochází z řečtiny - ze slov steganos (schovaný) a graphein (psát). Je to věda o utajení komunikace prostřednictvím ukrytí zprávy. Do této oblasti patří např. neviditelné inkousty. Steganografie je proto často kombinuje s kryptologií.

První zdokumentovaný případ použití steganografie je z Řecka, kdy Řek Demaratus žijící v Súsách poslal varování o perských přípravách na invazi do Řecka vyryté do voskové psací tabulky, z níž nejprve seškrábal vosk a po vyškrábání zprávy do dřevěného podkladu ji voskem opětovně zakryl. V jiném případě, který se odehrál jen několik desítek let po této události, byla otrokovi oholena hlava, zpráva napsána na jeho holou lebku a on byl vyslán s poselstvím na cestu poté, co mu vlasy dorostly. [6] [16, s. 126-128, 197-198]

2.2.1 Polybiův čtverec

Autorem je řecký spisovatel a historik Polybius (přibližně 203 – 120 př. n. l.). Jedná se o substituční šifru spojenou se steganografií. Každé písmeno se nahrazuje dvojicí čísel – číslem řady a číslem sloupce z Polybiova čtverce (5x5) viz Tab. 2. Místo písmen se pak v šifrovaném textu používají číslice. Pokud písmena nebyla zapsaná do čtverce abecedně nýbrž náhodně, nejedná se o šifru. Polybiův čtverec je základem mnoha dalších šifrovacích systémů. [16, s. 42-44, 201]

Tab. 2. Polybiův čtverec

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

Například písmeno R se šifruje jako 42, O jako 34. Šifrovaná zpráva je pak řada čísel.

2.2.2 Caesarova šifra

Caesarova šifra je jeden z nejznámějších šifrovacích systémů. Jedná se o monoalfabetickou substituční šifru. Vznikla kolem roku 50 př. n. l. Pojmenovaná je podle svého autora Julia Caesara (údajně jí využíval i pro dopisování s Kleopatrou). Byla to na onu dobu nerozluštitelná šifra, až do doby, kdy jí prozradil Cicero. [16, s. 45-46]

Princip spočívá v posunutí písmena o tři místa dále v abecedě.

Tab. 3. Otevřená a šifrovaná abeceda Caesarovy šifry

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

Příklad: otevřený text: SIFROVANI A GEOGEBRA

 šifrovaný text: VLIUTYDQLDJHRJHEUD

2.2.3 Starověká Indie

V Indii bylo šifrování oblíbené. Ve známé Kámasútře, kterou napsal indický učenec Vátsyáyana, se mezi 64 uměními, které si má osvojit každé děvče, pokud chce dokonale ovládat umění milovat, na 43. místě uvádí: „Vyznat se v tajných písmech a tajných znacích“.

V Indii má nepřímo původ samotné slovo šifra. Šlo o slovo s úplně jiným významem. Při přebírání číslovek Arabů od starověkých Indů přeložili Arabové i sanskritské označení sunya pro nulu. Toto slovo znamenající prázdný přeložili Arabové doslovně jako as-sifr. Při překládání děl Arabů do latiny se používala slova zephirium, cifra anebo figura nihil.

Později ve francouzštině z toho vzniklo chiffre. Stále se však jednalo o označení pro nulu. Ve 13. stol. se nula nazývala „kroužek“ (kroužek podobný otazníku) nebo „cifra“. Toto označení můžeme ojediněle nalézt ještě v 18. století. V 15. a 16. století označení cifra postupně zdomácnělo jako označení všech číslic. V této době měla nula zvláštní postavení v zápise čísel. Nakonec, na tu dobu i pravděpodobně tajuplná úloha nuly při zápise čísel, vedla k takovému obsahu slova šifra, jak ho známe dnes. [7] [16, s. 204-206]

2.2.4 Frekvenční analýza

První zmínka o frekvenční analýze pochází z 9. století, kdy arabský matematik Al Kindi popsal princip frekvenční analýzy.

Ve všech jazycích má každé písmeno svou charakteristickou četnost. Pokud se tedy frekvence využití písmena v šifrovaném textu blíží frekvenci jiného písmena obecně využívaného v dané řeči, jde pravděpodobně o jedno a totéž písmeno, resp. o jeho ekvivalent v rámci otevřeného textu. Základní představu o monoalfanumerických šifrách je tedy možné získat pomocí frekvenční analýzy – neaplikujeme ji však slepě. Příklad: From Zanzibar to Zambia and Zaire, ozone zones make zebras run zany zigzags. → Kryptoanalytik by měl být i tak trochu lingvista.

Postup dešifrování je následující. Nejprve se pokoušíme odhalit samohlásky (tvoří cca. 40% textu). Hledáme častá slova, která se vyskytují v daném jazyce. Dále zkusíme delfskou metodu – hádat celá slova, odhadnout obsah zprávy.

Šifrovaný text:	MRNBNA CNGC	RBC WRLQC VNQA	PNQNRV
Otevřený text:	DIESER TEXT	IST NICHT MEHR	GEHEIM
Volný překlad:	TENTO TEXT UŽ NENÍ TAJEMSTVÍM		

Tato metoda je efektivní pokud máme podezření, že se jedná o substituční šifru. Pokud se nejedná o posun čísel (obdoba Caesarovy šifry) je tato metoda neúčinná. Pokud se však jedná o posun, je lepší použít frekvenční analýzu, než zkoušet všechny možnosti. Šifru můžeme ztížit tím, že do otevřeného textu přidáme kódová slova nebo písmena, která po zašifrování mají za úkol klamat kryptoanalytika. Nemají s původním textem nic společného. [5]

2.3 Šifrování v Čechách

Prvním člověkem, který v Čechách šifroval svou tvorbu, byl mistr Jan Hus. Šifroval své Listy z Kostnice. Používal jednoduchou substituci. Nahrazoval pouze samohlásky a to písmeny, která následují po nich.

Příklad:	otevřený text:	ABECEDA
	šifrovaný text:	BBFCFDB

Při dešifrování nastává v tomto příkladu problém, že dešifrování není jednoznačné. BB můžeme dešifrovat jako AA, nebo jako AB nebo BA. Není jasné, které písmeno je v otevřeném textu samohlásky. Dešifrování není opravdu jednoznačné. [16, s. 213]

2.3.1 Složitější substitute

S nebezpečím válečných i politických konfliktů, rostla i potřeba šifrování. V roce 1467 sestavil Leon Battista Alberti zařízení, které šifrovalo složitěji, než bylo doposud známé. Zařízení (viz. Obr. 4.) se skládá ze dvou kruhů. Velkého vnějšího, který je pevný, a malého vnitřního, který je pohyblivý. Při šifrování se v prvním případě na obrázku šifruje A jako g. Při šifrování jiné části zprávy se vnitřní kruh pootočí a A se zašifruje např. jako o. Princip tedy spočívá v otáčení vnitřního kruhu při různých částech zprávy. Různé substitute pro různé části zprávy. Použití frekvenční analýzy v tomto případě není možné, protože každé písmeno je substituováno pokaždé jiným písmenem. Můžeme zde mluvit o prvním návrhu polyalfabetické substitute. [16, s. 213-215]



Obr. 4. Šifrovací zařízení [1]

2.4 Šifrování podle hesla

Roku 1581 vynalezl Johannes Trithemius tabulku zvanou tabula recta. Jedná se o čtvercovou tabulku, kdy první řádek je abeceda a každý další řádek je posunutý o jeden znak. Červená abeceda na Obr. 5 je pouze pomocná pro lepší orientaci při hledání.

	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
A	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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

Obr. 5. Tabula recta [8]

2.4.1 Vigenèrova šifra

Roku 1586 bývalý francouzský diplomat Blaise de Vigenère publikoval práci Traktát o šifrách, ve které demonstroval slabiny monoalfabetických šifer a do detailu popsal nový druh šifry. Šifra nese jeho jméno. Jedná se o polyalfabetickou (mnohoabecední šifru). Základní myšlenkou je používat střídavě různá monoalfabetická šifrování, čímž se kryptogram stává odolným vůči klasické frekvenční analýze. Jsou potřeba (pro šifrování i dešifrování) dvě základní věci: Klíč a Vigenèrův čtverec. Vigenèrův čtverec není ve skutečnosti nic jiného než, již zmíněná Tabula recta. [16, s. 61-68]

Příklad: otevřený text:

klíč:

šifrovaný text:

S	I	F	R	O	V	A	N	I	T	E	X	T	U
G	E	O	G	E	B	R	A	G	E	O	G	E	B
Y	M	T	X	S	W	R	N	O	X	S	D	X	V

Pod OT napíšeme opakovaně klíč o stejné délce jako OT. Pro šifrování použijeme Vigénèrovu tabulku. Např. S + G = Y. Při dešifrování postupujeme obráceně. Písmeno z klíče odpovídá danému řádku, na něm si najdeme písmeno z ŠT a sloupec nám prozradí jaké je písmeno z OT.

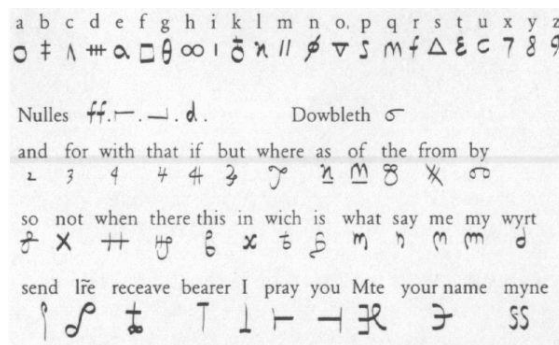
2.4.1.1 Rozluštění Vigénèrovy šifry

Přes 300 let se mnozí učenci domnívali, že Vigénèrova šifra je nerozluštitelná. Tento fakt platil až do roku 1854, kdy jí prolomil Charles Babbage. Ale své řešení nezdokumentoval. Proto se za autora řešení považuje Friedrich Kasiski.

Od té doby už nebyla bezpečná. Ale už jen fakt, že odolávala přes 300 let, je opravdu pozoruhodný.

2.4.2 Marie Stuartovna

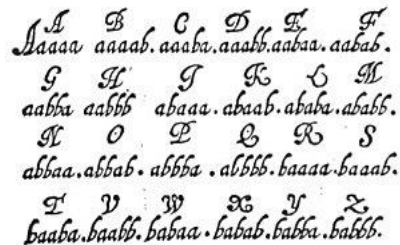
Zde se nacházíme v Anglii v 16. století. V této době, kdy se pořád válčí, je nesmírně důležité šifrovat zprávy, aby zpráva nepadla do nepovolaných rukou. Příběh Marie Stuartovny je velmi známý. Byla popravena kvůli nedokonalému šifrování zpráv. Z velezrady jí usvědčily vlastní dopisy, které posílala z vězení, kde ji držela anglická královna. Z rozluštěných dopisů vyplývá, že plánovala se svými komplici atentát na anglickou královnu. Marie Stuartovna byla popravena roku 1587 na hradě Fotheringay. [16, s. 229]



Obr. 6. Jednoduchá šifra Marie Stuartovny [9]

2.4.3 Baconova šifra

Autorem této další šifry byl Francis Bacon. Šifra je dalším pokrokem své doby. Řečeno jazykem dnešní doby se jedná o pětibitový binární kód. Každé šifrované písmeno je nahrazeno pětici znaků ve dvou různých fontech. Různé fonty kódovaly různá písmena.



Obr. 7. Baconova šifra [1]

2.4.4 Jeffersonův váleček

Přesuňme se do 19. století, ve kterém byl vynalezen a široce používán telegraf. Zprávy se daly jednoduše přenášet na velkou vzdálenost, ale za cenu ztráty soukromí – nebyly vůbec šifrovány. Význam šifrování tak opět vzrůstá.

Za americké občanské války (1861-1865) používá Unie zařízení zvané Jeffersonův váleček. Váleček se skládá z disků, na každém disku je napsaná abeceda v jiném pořadí. Odesílatel zprávy vybere pořadí válečků, nastaví zprávu, zvolí posun a jako zašifrovanou zprávu zapíše řádek vzdálený o posun od původní zprávy. Příjemce zprávy dostane zašifrovaný text, pořadí válečků a posun. [16, s. 241]



Obr. 8. Jeffersonův váleček [1]

2.4.5 Playfair

Playfair byla používaná ve Vietnamské válce. Substituční metoda s pomocí 2 kódových slov. Kódují se dvojice písmen křížem přes tabulku 4x (5x5).

Text: UTOK NA SEVERNI BRANU

Hesla: BRAVO ALPHA

Šifra: YPCOPBSICIELOBABUR

Nejprve vytvoříme tabulky. Do levé horní a pravé spodní zapíšeme hesla. Opakující se znaky hesla se nepiší. Tabulky doplníme podle abecedy s vynecháním znaků hesla. Poté vezmeme první dva znaky zprávy a určíme čtverec s použitím prvního znaku jako souřadnic v levé horní tabulce a druhého znaku jako souřadnic v pravé dolní tabulce. Následně sepíšeme další dva znaky z rohů čtverce v pořadí pravý horní a levý spodní. Substitucí pomocí tabulky dostaneme následující záměnu UT -> YP. Playfair se používala ještě za druhé světové války.

[10]

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

2.5 Kerckhoffovy principy

Důležitou osobností kryptografie 19. století byl August Kerckhoff, který přišel s několika principy bezpečné kryptografie. Například prosazoval, že bezpečnost šifry nemůže být založena na utajení jejího principu. Správná šifra by měla být prakticky nerozluštitelná i při znalosti jejího principu. Dalším z principů je, že klíč šifry musí být snadno sdělitelný bez použití záznamu apod. Obě dvě pravidla jsou dnes i ve 21. století stále aktuální a používají se. [1]

2.6 Gold-Bug

Toto šifrování není jen důležité, ale také zábavné. Vyskytuje se v legendách, příbězích, proniká do široké veřejnosti.

Za vznikem šifrování stojí masivní úspěch povídky Gold-Bug (zlatý brouk), jejímž autorem je Edgar Allan Poe. Úspěch povídky je založen na centrálním kryptogramu. Dochází k popularizaci šifrování. [1]

2.7 Legenda o Bealově pokladu

Ve zkratce: Počátek 19. století, Beale, hostinec, bedna, tři šifry a poklad. Pan Beale svěřil bednu panu hostinskému Morrisovi a poté zmizel. Pan Morris v bedně našel tři šifry udávající polohu pokladu. Nevěděl co s nimi a tak je svěřil svému příteli, který jejich luštěním strávil řadu let. Podařilo se mu ale vyluštit jen jednu z nich, která udává pouze přibližnou polohu pokladu. Všechny tři šifry zveřejnil v novinách, ale ty nebyly doposud rozlušťeny a ani poklad nebyl nalezen. Odhad ceny pokladu v roce 2010 je 65 milionů dolarů. Pravděpodobně se jedná o hoax (podvod, mystifikace, žert). [11, s. 8-19]

2.8 Šifrování veřejnosti

Šifrování proniká do široké veřejnosti. Vynalézaví Angličané vymysleli, jak si posílat dopisy zadarmo. Na rozdíl od dopisů bylo totiž posílání novin zadarmo, stačilo špendlíkem vypíchat písmena a skrytá zpráva byla na světě. Jednalo se o velmi vynalézavou metodu steganografie. [1]

2.9 1. světová válka

První světová válka vychovala i prvního z velikánů kryptologie dvacátého století - Williama Frederica Friedmana. Jeho čtyřsvazkové dílo "Základy kryptoanalýzy" z roku 1923 se stalo opravdovou biblí všech kryptologů první poloviny dvacátého století. [15, s. 24-26]

Podobně jako v 19. století měl zásadní vliv na šifrování telegram, ve 20. století to bylo rádio. Komunikace se zlepšila a hlavně zrychlila. Během první světové války sehrálo šifrování velkou roli. Používají se složitější substituce, ale také se používá šifra Playfair. Stále více populární jsou kódové knihy. Ale jejich velkou nevýhodou je, že při ztrátě nebo odcizení kódové knihy, jsou kódy ztraceny. Pro rozluštění zprávy pouze stačilo tuto knihu vlastnit. Nejednalo se zde o žádný šifrovací algoritmus. [16, s. 264-269]

2.9.1 Mata Hari

Šifrování sehrává roli v intrikách a aférách. Mata Hari, známá holandská tanečnice a kurtizána, byla nařčena ze špionáže pro Německo. Někteří historici ale pochybují o její vině,

protože zprávy, které údajně posílala, byly v kódu, o kterém Němci věděli, že jej Francouzi dokáží luštit. Nakonec Matu Hari usvědčil neviditelný inkoust nalezený v jejím pokoji. Nepomohlo ji ani tvrzení, že je to součást jejího make-upu. [1]

2.9.2 Room 40

Room 40 byla úspěšná kryptografická jednotka anglického námořnictva fungující během první světové války. Její zásluhou se povedlo zapojit USA do války. Němci se totiž chystali začít ponorkovou válku v Atlantiku. K odpoutání pozornosti Američanů chtěli využít Mexiko a přesvědčit jej, aby zaútočilo na Spojené státy. Angličané ale telegram pro Mexiko zachytili, rozluštili a ještě to zařídili tak, aby to vypadalo, že zprávu zachytili až rozluštěnou v Mexiku a Němci nepřišli na to, že umí luštit jejich kódy. [16, s. 265-269]

2.10 2. světová válka

V době druhé světové války dochází k velkému posunu od kryptologů-lingvistů ke kryptologům-matematikům. V této době také dochází k mechanizaci šifrování, pomalu se vytrácí klasika šifrování tužkou na papír. Ale třeba v šifře Playfair, která se stále ještě používala za druhé světové války, to neplatí. Zde je papír a tužka nepostradatelným pomocníkem. [16, s. 272-277]

2.10.1 Sigaba, Typex, Purple, Enigma

Co se týče mechanizace šifrování za druhé světové války, asi každého napadne ENIGMA. Rozluštění jejího principu znamenalo klíčový zvrat války. Ale i jiné země měly šifrovací stroje, jejichž jména zůstala v pozadí. Tyto stroje nejsou v historii tak známé, jako již zmíněná Enigma. Američané měli stroj Sigaba, v Anglii to byl Typex, v Japonsku Purple. [16, s. 272]

2.11 Moderní kryptografie

V dnešní moderní době se k bezpečné komunikaci používají výhradně elektronická média. Již žádná tužka a papír. Každá zpráva se před šifrováním vždy převádí do binární podoby (sled 0 a 1). Tak jako elektronická média výrazně pomohla kryptogramům, na druhou stranu poskytla důmyslnější nástroje útočníkům. Základem moderních šifer jsou polyalfabetické šifry pracující s různou délkou klíče. Základní dělení v moderní kryptografii: symetrické šifrování a asymetrické šifrování. [15, s. 24-26]

2.11.1 Symetrické šifrování

Založeno na principu jednoho klíče, kterým lze otevřený text jak zašifrovat tak i dešifrovat. Značnou výhodou je nízká výpočetní náročnost. Největší nevýhodou je, že pro tajnou komunikaci je nutné si bezpečným kanálem předat klíč. Při komunikaci více osob najednou je potřeba $\frac{n \cdot (n-1)}{2}$ klíčů, kde n je počet osob. Mezi symetrické metody kryptografie patří např.: DES, 3DES, AES, IDEA,... [15, s. 24-26]

2.11.2 Asymetrické šifrování

Skupina kryptografických metod, ve kterých se pro šifrování a dešifrování používají odlišné klíče – veřejný klíč a soukromý klíč. Veřejný klíč je komukoliv dostupný a lze s ním zašifrovat zprávu pro určitého uživatele. Soukromý klíč se používá pro dešifrování zprávy a je vlastníkem pečlivě chráněn. Tím tedy odpadá problém distribuce klíčů. Nepoužívá se pouze při komunikaci, ale i pro podepisování zpráv. Šifrování je založeno na tzv. jednocestných funkcích, tedy operacích, které lze snadno provést pouze v jednom směru (ze vstupu lze snadno spočítat výstup, ale z výstupu je obtížné získat vstup). Velkou nevýhodou je však rychlost (1000x pomalejší než symetrické metody). [15, s. 24-26]

2.11.3 Kvantová kryptografie

Metoda pro bezpečný přenos informací. Bezpečnost garantují fundamentální zákony kvantové fyziky. Řeší problém bezpečné distribuce klíčů. Umožňuje spolehlivou detekci odposlechu. Využívá Vernamovu šifru (zcela náhodný klíč stejné délky jako zpráva). Základem je přenos sekvence pomocí stavu částic fotonu. [15, s. 24-26]

3 PŘENOS ČÍSELNÝCH POSLOUPNOSTÍ

Ochrana digitálních dat hraje důležitou roli v datové komunikaci prostřednictvím internetu. Rozhodně je třeba chránit data před neoprávněným přístupem. Data mohou existovat v mnoha formách, jako jsou obrázky, tabulky, ikony, text, video, audio, barvy atd. Dosud bylo navrženo mnoho metod a algoritmů na světě. Kryptografie hraje důležitou roli při kódování a dekódování informací. [13]

3.1 Hillova šifra

Jedná se o matematickou šifru. Znaky abecedy převedeme na čísla 0 – 25. $A=1$, $B=2$, $C=3$, atd. Klíčem je náhodně zvolená matice A stupně n (nesmí být singulární, determinant se nesmí rovnat nule). Text rozdělíme do bloků o délce n a převedeme na číselné vektory v . Šifrování probíhá tak, že každý blok zašifrujeme tak, že jej jako vektor vynásobíme s maticí. Provedeme modulo 26 a převedeme zpět z číselné formy do textové. U záporných čísel přičítáme 26, abychom dostali výsledek z rozsahu 0 - 25.

$$ŠT = (A \cdot v) \text{ mod } 26$$

Při dešifrování jednotlivé bloky šifrovaného textu násobíme s inverzní maticí A . Poté opět provedeme modulo 26 a převedeme zpět z číselné formy do textové. K záporným číslům přičítáme 26, abychom dostali výsledek z rozsahu 0 – 25. [13]

$$OT = (A^{-1} \cdot ŠT) \text{ mod } 26$$

3.2 Teorie kódování

3.2.1 Šifrování

Je dán OT: I LOVE VINODHINI. Ke každému znaku přiřadím číslo. Nejjednodušší způsob je za mezery vložit 0. Dále pak $A=1$, $B=-1$, $C=2$, $D=-2$,... nebo druhý způsob jako v Hillově šifře. Pro ukázkou zvolím první způsob.

I	L	O	V	E	V	I	N	O	D	H	I	N	I		
5	0	-6	8	-11	3	0	-11	5	-7	8	-2	-4	5	-7	5

Čísla zapíši do matice M 6x3. Na zbylá místa dopíšu nuly.

$$M = \begin{pmatrix} 5 & 0 & -4 \\ 0 & -11 & 5 \\ -6 & 5 & -7 \\ 8 & -7 & 5 \\ -11 & 8 & 0 \\ 3 & -2 & 0 \end{pmatrix}$$

Vytvořím libovolnou regulární matici A 3x3 a z ní vypočítám inverzní matici A^{-1} .

$$A = \begin{pmatrix} -1 & 5 & -1 \\ -2 & 11 & 7 \\ 1 & -5 & 2 \end{pmatrix}, A^{-1} = \begin{pmatrix} 57 & -5 & 46 \\ 11 & -1 & 9 \\ -1 & 0 & -1 \end{pmatrix}$$

Matici X poté získám vynásobením dvou matic M a A

$$X = M \cdot A = \begin{pmatrix} 5 & 0 & -4 \\ 0 & -11 & 5 \\ -6 & 5 & -7 \\ 8 & -7 & 5 \\ -11 & 8 & 0 \\ 3 & -2 & 0 \end{pmatrix} \begin{pmatrix} -1 & 5 & -1 \\ -2 & 11 & 7 \\ 1 & -5 & 2 \end{pmatrix} = \begin{pmatrix} -9 & 45 & -13 \\ 27 & -146 & -67 \\ -11 & 60 & 27 \\ 11 & -62 & -47 \\ -5 & 33 & 67 \\ 1 & -7 & -17 \end{pmatrix}$$

ŠT je pak posloupnost čísel z matice X -9,27,-11,11,-5,...,-67,27,-47,67,-17.

Zde je jasné, že čísla nelze převést zpátky na písmena. [14]

3.2.2 Dešifrování

Dešifrování probíhá obdobně. Mám šifrovanou posloupnost čísel -9,27,-11,11,-5,...,-67,27,-47,67,-17. To přepíšu do matice, kterou vynásobím s maticí A^{-1} . Poté dostanu matici M . [14]

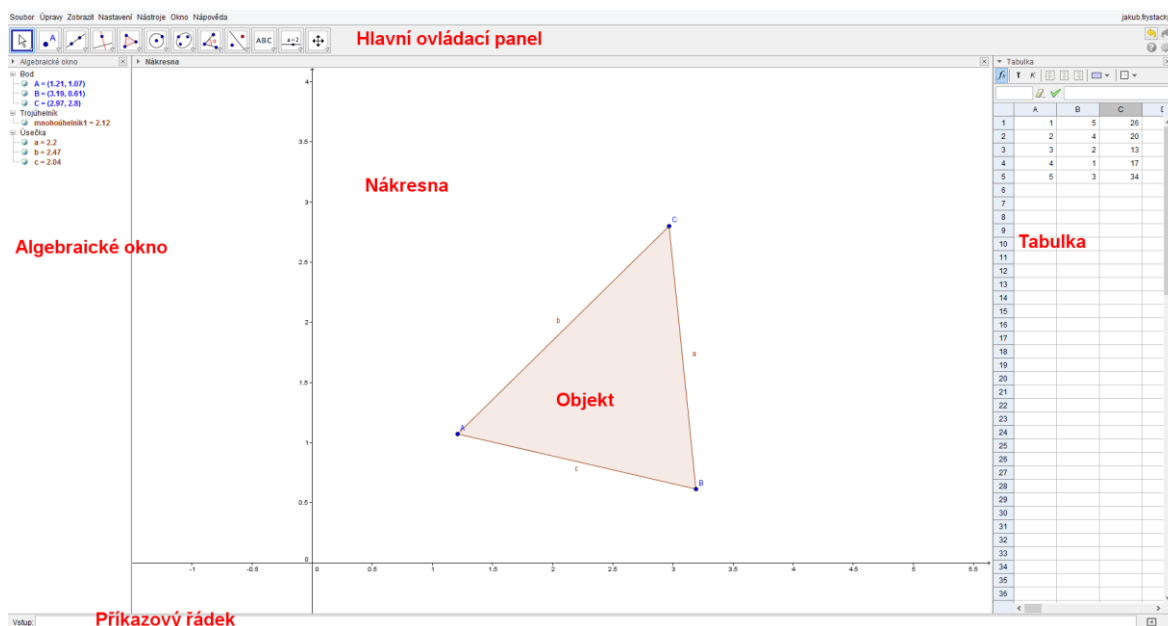
$$M = X \cdot A^{-1} = \begin{pmatrix} -9 & 45 & -13 \\ 27 & -146 & -67 \\ -11 & 60 & 27 \\ 11 & -62 & -47 \\ -5 & 33 & 67 \\ 1 & -7 & -17 \end{pmatrix} \cdot \begin{pmatrix} 57 & -5 & 46 \\ 11 & -1 & 9 \\ -1 & 0 & -1 \end{pmatrix} = \begin{pmatrix} 5 & 0 & -4 \\ 0 & -11 & 5 \\ -6 & 5 & -7 \\ 8 & -7 & 5 \\ -11 & 8 & 0 \\ 3 & -2 & 0 \end{pmatrix}$$

4 GEOGEBRA

Světově nejoblíbenější dynamický matematický systém. Učit se. Vyučovat. Sdílet. Tak jednoduše lze stručně popsat GeoGebra. GeoGebra (www.geogebra.org) je zdarma dostupný dynamický matematický software určený pro všechny úrovně vzdělávání, který propojuje geometrii, algebru, tabulky, grafy, statistiku a kalkulus v jednom snadno použitelném balíčku. Interaktivní materiály určené pro výuku, hodnocení nebo analýzu problémů vytvořené v systému GeoGebra mohou být kýmkoliv sdíleny a užívány na www.geogebra.org. GeoGebra je světově nejoblíbenější dynamický matematický software, který jako vzdělávací software získal řadu ocenění, a který podporuje STEM vzdělávání a inovace ve výuce po celém světě. [12, s. 11]

4.1 Prostředí

S vytvářenými objekty se dá pracovat v okně Nákresna nebo v Algebraickém okně. Také lze spolupracovat s tabulkou, do které můžeme zadávat data. Příkazy je možné kromě nástrojů v grafickém okně také zadávat z Příkazového řádku. Kterýkoliv objekt můžeme skrýt nebo zobrazit. Nákresnu lze posouvat myší a přibližovat/vzdalovat ji kolečkem na myši. Základní nastavení vzhledu objektů Grafického okna (Nákresna) se může nastavit v menu Nastavení, Pro pokročilé v záložce Předvolby – Nákresna. [12, s. 13]



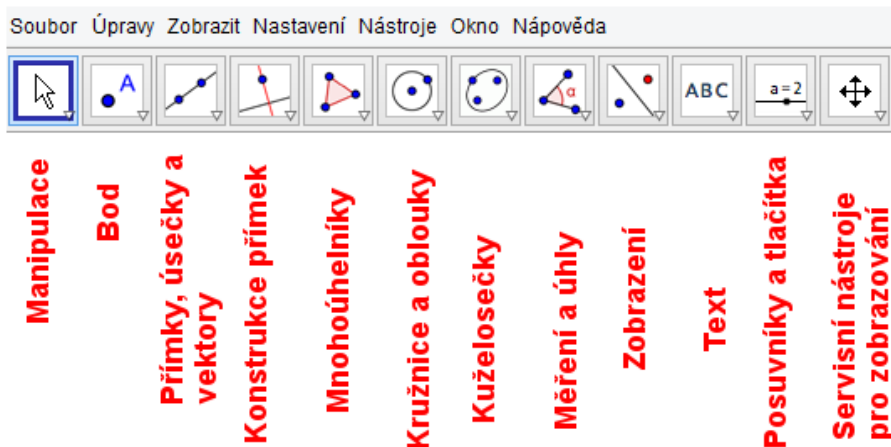
Obr. 9. Prostředí programu GeoGebra

4.2 Ovládání

Výběr a kreslení myší je možný pohybem, tažením či kurzorovými klávesami. Pokud po výběru objektu začneme psát na klávesnici, objekt se přejmenuje. Je možný výběr několika objektů: Shift, Ctrl nebo obdélníkový výběr myší v nákresně. Při pohybování pomocí kurzorových kláves: Shift zjemňuje krok, Ctrl krok zvětšuje. Chceme-li v manipulačním režimu posunout myší nákresnu, stiskneme klávesu Shift nebo Ctrl => nebudeme posouvat vybraným objektem ale celou nákresnou. Pro mazání funguje klávesa Delete nebo příkaz z kontextového menu. [12, s. 15]

4.3 Grafický vstup

Grafické okno umožňuje zobrazit objekty konstrukce zvoleným způsobem a se zvoleným popisem. Popis objektu nemusí být totožný s jeho (jednoznačným) názvem a může obsahovat i zobrazení hodnoty. Pomocí nástrojů umístěných v Hlavním panelu nástrojů zadáváme v grafickém okně prvky geometrických konstrukcí. K dispozici jsou následující sady nástrojů:



Obr. 10. Ovládací panel – nástroje

Z každé skupiny objektů lze vybrat několik nástrojů. V menu Manipulace jsou to Ukazovátka, Otočení a Zaznamenat do tabulky. Obdobně je to i pro další položky. Vždy se jedná o jednoduché rozevírací menu. [12, s. 14]

4.4 Příkazový řádek

dovoluje vkládat další příkazy, například i ty, které nelze zadávat v grafickém prostředí. Příkazů je velké množství – podporují (a přesahují) celé středoškolské téma Funkce,

Analytická geometrie a Diferenciální a integrální počet. Zadáváme-li do příkazové řádky funkci nebo logickou operaci, můžeme jí také vybrat z pomocné rozbalovací nabídky vpravo od příkazového řádku nebo zapsat pomocí klávesnice. Chceme-li operátory rovnoběžnost nebo kolmost můžeme je zadat z pomocné nabídky nebo z příkazového řádku pomocí příkazu *Kolmice[]*. Zadání komplexního čísla – např. $(1 + i) * (1 - i)$, $(1 + i) / (1 - i)$ apod. Matice se zadává po řádcích: $\{\{1, 2, 3\}, \{4, 5, 6\}, \{7, 8, 9\}\}$ má první řádek 1 2 3 a první sloupec 1 4 7. Matice lze sčítat, odčítat, násobit, transponovat, invertovat, počítat determinant. [12, s. 14]

4.4.1 Přímý vstup

Kromě použití předefinovaných příkazů lze v příkazovém řádku zadat některé hodnoty přímo. Pak je nutné respektovat několik syntaktických pravidel:

Názvy příkazů zapisujeme bez diakritiky – přesně tak, jak jsou uvedeny v seznamu.

- zadání ukončíme klávesou Enter
- desetinná čísla se zadávají s desetinnou tečkou
- ve výrazech lze používat konstantu pí, e pouze tehdy, nejmenuje-li se již tak nějaká použitá proměnná. Obě konstanty se dají vyvolat z pomocného výběrového okna
- znak násobení * lze nahradit mezerou
- zadání bodu: $A = (1.2, 4)$ – kartézské souřadnice, $B = (1; 45^\circ)$ – polární souřadnice; zadání komplexního čísla: $C = 2 + i$; zadání vektoru: $a = (1.1, 7)$ – jako bod, ale malým písmenem
- zadání funkce: $f(x) = \text{abs}(x-1)$ nebo $f: y = \text{abs}(x-1)$ (píšeme s rovnítkem). Pokud funkci nepojmenujeme, pojmenuje ji systém sám
- zadání obecné rovnice přímky a kuželosečky: $h: x^2 + y^2 = 4$ (uvozuje se dvojtečkou)
- zadání (parametrické) přímky: napřed musíme definovat proměnnou, která bude parametrem a pak lze definovat např. $p: M = (-2, 0) + t(1, -2)$ [12, s. 11-16]

II. PRAKTICKÁ ČÁST

5 PRÁCE S MATICEMI

Tato kapitola se bude zabývat tématem jak pracovat s maticemi. Jak je vytvořit jednoduše a rychle. Dále pak základními operacemi s nimi.

5.1 Tvorba matic

Matice lze v GeoGebře vytvářet dvěma způsoby:

1. Přes Příkazový řádek ve formátu:

$$A = \{ \{1, 6, 2, 4\}, \{3, 4, 5, 2\}, \{4, 7, 3, 8\}, \{5, 7, 1, 6\} \}$$

Tento příkaz vytvoří matici A o rozměru 4x4 s danými hodnotami.

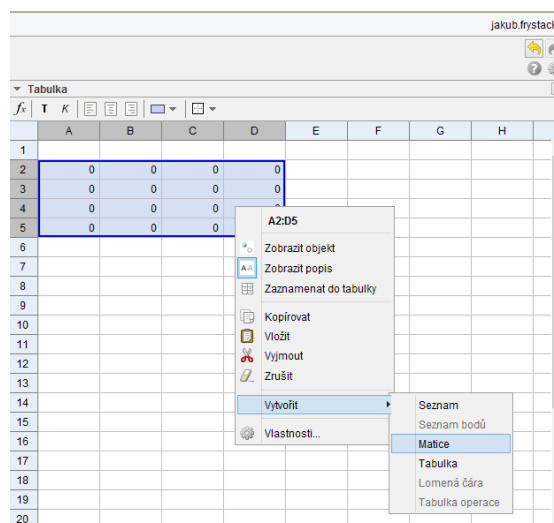
2. Přes okno Tabulka:

Tento způsob vkládání je mnohem přehlednější. Využijte se tabulkové prostředí (klávesová zkratka Ctrl+Shift+S nebo myší Hlavní menu – Zobrazit – Tabulka).

Matice se pak vloží v následujících krocích.

- Každý prvek matice se zapíše do jedné buňky. Řádky i sloupce musí odpovídat zadání. Buňky tvoří souvislou oblast.
- Tažením myší se provede výběr matice.
- Kliknutím pravého tlačítka se vyvolá kontextové menu. Vybere se položka Vytvořit – Matice. (viz. Obr. 11)

Takto zadávané matice se objevují v Algebraickém okně pod názvy matice1, matice2,... Proto je vhodné nakonec si takto vytvořené matice přejmenovat dle vlastního uvážení.



Obr. 11. Vytvoření matice přes okno Tabulka

5.2 Součet matic

Syntaxe příkazu: $\langle \text{Matice} \rangle + \langle \text{Matice} \rangle$

Součet matic je definován pro 2 matice stejného rozměru. Při součtu dvou matic $m \times n$ je výsledná matice taktéž $m \times n$. Součet matic má smysl pouze pro matice stejného typu. Sčítání dvou matic se provádí sečtením odpovídajících prvků.

Ukázka součtu matic v programu GeoGebra (Obr. 12 a 13):

Součet matic

Výběrem (zatržením) zvolte režim zadávání

Ruční zadávání

Zadejte první matici:

$$\begin{pmatrix} 1 & 16 & 2 & 48 \\ 35 & 41 & 5 & 23 \\ 64 & 4 & 33 & 84 \\ 55 & 37 & 14 & 16 \end{pmatrix}$$

Zadejte druhou matici:

$$\begin{pmatrix} 15 & 19 & 36 & 45 \\ 7 & 6 & 8 & 63 \\ 4 & 9 & 55 & 46 \\ 4 & 36 & 41 & 5 \end{pmatrix}$$

Sečti

Výsledek
$$\begin{pmatrix} 1 & 16 & 2 & 48 \\ 35 & 41 & 5 & 23 \\ 64 & 4 & 33 & 84 \\ 55 & 37 & 14 & 16 \end{pmatrix} + \begin{pmatrix} 15 & 19 & 36 & 45 \\ 7 & 6 & 8 & 63 \\ 4 & 9 & 55 & 46 \\ 4 & 36 & 41 & 5 \end{pmatrix} = \begin{pmatrix} 16 & 35 & 38 & 93 \\ 42 & 47 & 13 & 86 \\ 68 & 13 & 88 & 130 \\ 59 & 73 & 55 & 21 \end{pmatrix}$$

Obr. 12. Součet matic – ruční zadávání

Součet matic

Výběrem (zatržením) zvolte režim zadávání

Automatické zadávání

První matice
$$\begin{pmatrix} 33 & 1 & 77 & 83 \\ 72 & 9 & 14 & 74 \\ 41 & 63 & 55 & 16 \\ 2 & 28 & 35 & 8 \end{pmatrix}$$
 Generuj

Druhá matice
$$\begin{pmatrix} 73 & 1 & 4 & 60 \\ 79 & 66 & 89 & 55 \\ 93 & 65 & 7 & 47 \\ 65 & 62 & 18 & 84 \end{pmatrix}$$
 Generuj

Sečti

Výsledek
$$\begin{pmatrix} 1 & 16 & 2 & 48 \\ 35 & 41 & 5 & 23 \\ 64 & 4 & 33 & 84 \\ 55 & 37 & 14 & 16 \end{pmatrix} + \begin{pmatrix} 15 & 19 & 36 & 45 \\ 7 & 6 & 8 & 63 \\ 4 & 9 & 55 & 46 \\ 4 & 36 & 41 & 5 \end{pmatrix} = \begin{pmatrix} 106 & 2 & 81 & 143 \\ 151 & 75 & 103 & 129 \\ 134 & 128 & 62 & 63 \\ 67 & 90 & 53 & 92 \end{pmatrix}$$

Obr. 13. Součet matic – automatické zadávání

Ukázkový program – jeho zdrojový kód je dostupný jako příloha v souboru Součet matic.ggb. Jedná se o jednoduchý součet dvou libovolných matic pro ruční zadávání nebo matic o rozměru 4x4 po automatickém zadávání. Při spuštění se vybere, zda se matice budou zadávat ručně do textového pole (obdoba Příkazového řádku) nebo zda se budou sčítat náhodné matice o rozměru 4x4. Při volbě ručního zadávání se obě matice zadají do příslušných polí a stiskem tlačítka Sečti se odpovídající prvky obou matic sečtou. Pokud matice nemají stejný rozměr, výsledkem je prázdná matice, která naznačuje chybu. Všechny průběžné matice se pro kontrolu uživatele vypisují po stranách.

Oba režimy nelze aktivovat současně vždy jen jeden nebo žádný režim zadávání.

V režimu automatické zadávání (Obr. 13) jsou rozměry matice pevně definované. Jednotlivé prvky matic se jednoduše vygenerují, stisknutím příslušného tlačítka Generuj. Tlačítka Sečti dané matice sečte a vypíše výsledek. Oba dva režimy lze libovolně přepínat.

5.3 Součin matic

Syntaxe příkazu: $\langle \text{Matice} \rangle \langle \text{Matice} \rangle$

Máme-li první matici (matice A) typu $m \times s$ a druhou matici (matice B) typu $s \times n$, pak jejich součinem je matice C typu $m \times n$, který značíme $C=AB$.

Ukázkový program Součin matic.ggb (Obr. 14. a 15.) je laděn do stejné grafiky a ovládání jako předchozí program. Opět jsou zde dostupné dva režimy zadávání – ruční a automatické zadávání viz následující obrázky.

Součin matic

Výběrem (zatržením) zvolte režim zadávání

Ruční zadávání

Zadejte první matici:

Zadejte druhou matici:

$$\begin{pmatrix} 1 & 16 & 2 & 48 \\ 35 & 41 & 5 & 23 \\ 64 & 4 & 33 & 84 \end{pmatrix}$$

$$\begin{pmatrix} 15 & 19 & 36 & 45 \\ 7 & 6 & 8 & 63 \\ 4 & 9 & 55 & 46 \\ 4 & 36 & 41 & 5 \end{pmatrix}$$

Násob první-druhá

Násob druhá-první

Výsledek
$$\begin{pmatrix} 327 & 1861 & 2242 & 1385 \\ 924 & 1784 & 2806 & 4503 \\ 1456 & 4561 & 7595 & 5070 \end{pmatrix}$$

Obr. 14. Součin matic – ruční zadávání

V tomto ukázkovém programu je řešena problematika součinu matic při ručním zadávání. Lze zadat libovolné matice s libovolnými rozměry. Násobení probíhá pomocí dvou tlačítek. Je možné vynásobit první zadanou matici s druhou nebo druhou zadanou matici s první. Výsledek se vypíše v dolní části programu. V některých případech nelze spolu matice vynásobit kvůli rozměru matice např. můžeme mít matici A typu 1x4 a matici B typu 4x4. Matici A lze vynásobit s maticí B, platí podmínka při násobení matic. Při násobení matice B maticí A GeoGebra problém sama rozpozná a obě matice spolu nevynásobí, zůstane pouze prázdná množina, která je pro uživatele znakem, že násobení v tomto pořadí není možné.

Součin matic

Výběrem (zatržením) zvolte režim zadávání

Automatické zadávání

První matice (10 46 39 25) Generuj

Druhá matice $\begin{pmatrix} 18 & 3 & 89 & 64 \\ 17 & 36 & 63 & 21 \\ 50 & 94 & 2 & 9 \\ 57 & 60 & 53 & 57 \end{pmatrix}$ Generuj

Násob první·druhá Násob druhá·první

Výsledek (4337 6852 5191 3382)

Obr. 15. Součin matic – automatické zadávání

V této části programu je demonstrován předcházející popsáný případ. První matice jde vynásobit s druhou, druhá však s první nikoliv a objeví se prázdná množina. To je pro uživatele upozornění, že dané matice nelze vynásobit. Rozměry obou matic nelze měnit. Lze měnit pouze čísla v obou maticích a to tlačítkem Generuj, které vygeneruje náhodná čísla do matice.

5.4 Determinant matice

Syntaxe příkazu: *Deteminant*[<Matice>]

Determinantem čtvercové matice řádu n nazýváme součet všech součinů n prvků této matice takových, že v žádném z uvedených součinů se nevyskytují dva prvky z téhož řádku ani z téhož sloupce. Každý součin přitom označíme znaménkem permutace.

Tento ukázkový program nazvaný Determinant matice.ggb (Obr. 16.) ukazuje práci s determinanty.

Determinant matice

Zadejte matici A: $\begin{pmatrix} 5 & 1 & 2 & 8 \\ 5 & 4 & 5 & 2 \\ 4 & 4 & 3 & 6 \\ 5 & 8 & 9 & 4 \end{pmatrix}$

nebo matici B $\begin{pmatrix} 94 & 2 & 23 & 13 \\ 39 & 52 & 89 & 70 \\ 45 & 78 & 27 & 65 \\ 44 & 1 & 36 & 66 \end{pmatrix}$

Determinant matice A= -270

Determinat matice B= -25174633

Determinat jednotkové matice

$$\det \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = 1$$

Obr. 16. Determinant matice

Tento ukázkový program spočítá determinant matice. Zadávání matic probíhá opět dvěma způsoby. První způsob je určený pro ruční zadávání podle syntaxe, umožňuje zadat libovolný rozměr matice a libovolné hodnoty prvků matice. Druhá možnost je určena matice pevných rozměrech 4x4, ve které je možné změnit prvky generováním náhodných čísel. V tomto programu není zapotřebí tlačítka na provádění výpočtu. Výpočet se dělá plně automaticky. V prostoru pod zadáním se vypočítá determinant obou zadaných matic. Determinant matice je pouze „číslo“ (nikoliv vektor nebo matice), což dokazuje výpočet. Determinant z jednotkové matice o jakémkoliv rozměru je vždy roven jedné. Determinanty matice 2x2 nebo matice 3x3 (spočítání determinantu podle Cramerova pravidla) lze spočítat poměrně snadno, ale determinant matice 4x4 již může být problém spočítat. Determinant se vždy počítá ze čtvercové matice. Tuto skutečnost si lze taky ověřit pomocí programu při ručním zadání matice. Pokud se stane, že zadaná matice není čtvercová, determinant matice A není vypočítán.

5.5 Inverzní matice

Syntaxe příkazu: *Invertovat*[<Matice>]

Inverzní matice k dané matici je taková matice, která po vynásobení s původní maticí dá jednotkovou matici. Inverzní matici lze sestavit pouze pro regulární matici (čtvercovou matici s nenulovým determinatem).

Program Inverzní matice.ggb (Obr. 17.) umožňuje výpočet libovolné inverzní matice nebo matice 4x4 s náhodně vygenerovanými prvky.

Inverzní matice

Zadejte matici A: $\begin{pmatrix} 5 & 1 & 2 & 8 \\ 5 & 4 & 5 & 23 \\ 64 & 4 & 33 & 6 \\ 5 & 8 & 9 & 41 \end{pmatrix}$

nebo matici B $\begin{pmatrix} 34 & 60 & 15 & 7 \\ 68 & 57 & 45 & 59 \\ 14 & 83 & 55 & 92 \\ 17 & 76 & 88 & 66 \end{pmatrix}$ Generuj Invertovat

Inverzní matice $A^{-1} = \begin{pmatrix} 1.29 & -1.56 & -0.01 & 0.62 \\ 6.13 & -9.45 & -0.06 & 4.12 \\ -3.13 & 3.95 & 0.06 & -1.62 \\ -0.67 & 1.17 & 0 & -0.5 \end{pmatrix}$ Inverzní matice $B^{-1} = \begin{pmatrix} 0 & 0.02 & -0.01 & 0 \\ 0.02 & -0.01 & 0.01 & 0 \\ -0.01 & 0 & -0.02 & 0.02 \\ -0.01 & 0.01 & 0.02 & -0.01 \end{pmatrix}$

Důkaz: $A \cdot A^{-1}$ nebo $A^{-1} \cdot A =$ jednotková matice

$$\begin{pmatrix} 5 & 1 & 2 & 8 \\ 5 & 4 & 5 & 23 \\ 64 & 4 & 33 & 6 \\ 5 & 8 & 9 & 41 \end{pmatrix} \begin{pmatrix} 1.29 & -1.56 & -0.01 & 0.62 \\ 6.13 & -9.45 & -0.06 & 4.12 \\ -3.13 & 3.95 & 0.06 & -1.62 \\ -0.67 & 1.17 & 0 & -0.5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Obr. 17. Inverzní matice

Vzorový program Inverzní matice.ggb vypočítá libovolnou inverzní matice ze zadané matice. Zadaná matice může být buď vepsaná ručně do textového pole, nebo se jedná o matici, která má pevně daný rozměr 4x4. V této matici je možné si jednotlivé prvky nechat vygenerovat tlačítkem Generuj. Inverzní matice k oběma zadaným maticím se vypíše o kousek níže. Jako důkaz a pro kontrolu, že vše je správně spočítáno, je provedený součin matice A a její inverzní matice B (násobíme zprava i zleva). Tento důkaz je provedený pouze pro matici A, ale s jistotou platí pro jakoukoliv matici. Princip důkazy spočívá v tom, že součin matice a její inverzní matice je vždy roven jednotkové matici. Při tomto násobení obou matic nezáleží na jejich pořadí, protože výsledek je vždy jednotková matice stejného rozměru jako původní matice a k ní inverzní matice. Celý tento důkaz je prakticky rozepsaný v poslední části programu.

5.6 Transponovaná matice

Syntaxe příkazu: *Transponovat* [<Matice>]

Transponovaná matice je matice, která vznikne z matice A vzájemnou výměnou řádků a sloupců. Pokud má matice A rozměry $(m \times n)$, pak její transpozicí vznikne matice o rozměrech $(n \times m)$. Ukázkový program Transponována matice.ggb poskytuje praktický náhled na celou situaci.

Transponovaná matice

Zadejte matici A: $\begin{pmatrix} 5 & 1 & 2 & 8 \\ 5 & 4 & 5 & 23 \\ 64 & 4 & 33 & 6 \\ 5 & 8 & 9 & 41 \end{pmatrix}$

nebo matici B $\begin{pmatrix} 23 & 45 & 65 & 31 \\ 35 & 19 & 9 & 27 \\ 36 & 54 & 66 & 56 \\ 92 & 68 & 6 & 71 \end{pmatrix}$ Generuj Transponovat

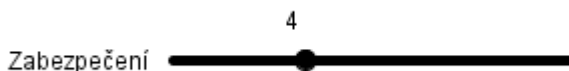
Transponovaná matice $A_T = \begin{pmatrix} 5 & 5 & 64 & 5 \\ 1 & 4 & 4 & 8 \\ 2 & 5 & 33 & 9 \\ 8 & 23 & 6 & 41 \end{pmatrix}$ Transponovaná matice $B_T = \begin{pmatrix} 23 & 35 & 36 & 92 \\ 45 & 19 & 54 & 68 \\ 65 & 9 & 66 & 6 \\ 31 & 27 & 56 & 71 \end{pmatrix}$

Obr. 18. Transponovaná matice

Stejně jako i v předcházejících ukázkových programech, tak i tento poslední ukázkový program je laděn ve stejném stylu jako předešlé ukázkové programy. Opět lze zdávat vlastní matici o libovolném rozměru a různých prvcích, nebo nechat generovat jednotlivé prvky v matici 4x4. Toho lze dosáhnout pomocí tlačítka Generuj. Další tlačítko Transponuj vypočítá k oběma zadaným maticím matice transponované. Oba výsledky se vypíší níže. Zajímavostí je, že pokud dvakrát transponujeme danou matici, dostáváme opět původní matici.

6 NÁSTROJ PRO ŠIFROVÁNÍ TEXTU

V této kapitole bude popsán nástroj pro šifrování textu v programu GeoGebra. Popsány budou jednotlivé části programu, doplněné grafickým výstupem z aplikace. U tlačítek bude popsán zdrojový kód a funkce jednotlivých částí. Pro větší přehlednost jsou tyto zdrojové kódy a vysvětlení uvedeny jako další kapitola.



Obr. 19. Posuvník

První krok, který je nutno provést, je nastavení posuvníku: Zabezpečení. Nastavit lze hodnotu od 1 do 10, vždy celé číslo. Posuvník slouží k určení velikosti matice klíče např. hodnota 4 = čtvercová matice klíče 4x4. Pro potvrzení vstupu se musí zmáčknout klávesa Enter na klávesnici.

Otevřený text: Frekvenční analýza

{70, 114, 101, 107, 118, 101, 110, 269, 110, 237, 32, 97, 110, 97, 108, 253, 122, 97}

{-26, 18, 5, 11, 22, 5, 14, 173, 14, 141, -64, 1, 14, 1, 12, 157, 26, 1}

Obr. 20. Otevřený text

Do textového pole Otevřený text uživatel zadá libovolný text, který chce zašifrovat. Ten se poté převede podle ASCII tabulky na seznam. Tento vektor se dále posune zpětně o hodnotu 96, aby písmeno *a* (ASCII = 97) odpovídalo číslu 1. Všechny průběžné změny jsou zobrazeny ve vektorech pod textovým polem.

Doplnění

{-26, 18, 5, 11, 22, 5, 14, 173, 14, 141, -64, 1, 14, 1, 12, 157, 26, 1, -64, -64}

Obr. 21. Doplnění

Pokud délka otevřeného textu není násobkem čísla nastaveného na posuvníku, je třeba doplnit zbývající čísla ve vektoru do nejbližšího násobku. Záměrně je zvoleno číslo -64,

protože se jedná o posunutý ASCII znak mezery. Stiskem tlačítka Doplnění se provede doplnění vektoru na požadovanou délku.

Matrice

$$\begin{pmatrix} -26 & 18 & 5 & 11 \\ 22 & 5 & 14 & 173 \\ 14 & 141 & -64 & 1 \\ 14 & 1 & 12 & 157 \\ 26 & 1 & -64 & -64 \end{pmatrix}$$

Obr. 22. Matice

Po stisku tlačítka Matice se daný vektor převede na Matici. Vektor se rozdělí na části o délce, která odpovídá hodnotě nastaveného posuvníku, jednotlivé části pak tvoří řádky hledané matice s názvem Matice. Nyní je otevřený text zcela nachystán na šifrování.

Klíč: Caesar

{67, 97, 101, 115, 97, 114}

Obr. 23. Klíč

Vstupní pole Klíč slouží uživateli k zadání klíče, podle kterého se zadaný otevřený text zašifruje. Opět je třeba po ukončení zadávání stisknout klávesu Enter a tím potvrdit konec zadávání do textového pole. Klíč by měl být delší než číslo na posuvníku. Dále by se neměly v klíči opakovat stejné úseky písmen. Mohlo by se stát, že daný text se sice zašifruje, ale už nikdy se nerozšifruje. Klíč se opět převádí na ASCII znaky a pro kontrolu se tento vektor vypisuje pod textovým polem.

Úprava klíče

{67, 97, 101, 115, 97, 114, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Obr. 24. Úprava klíče

Po stisku tlačítka Úprava klíče dojde k úpravě vektoru klíče, který byl vytvořen v předcházejícím kroku. Nejdříve je třeba si definovat, jakou délku bude mít klíč. Délka klíče se spočítá z nastaveného čísla na posuvníku. Jedná se o druhou mocninu zadaného

čísla. Nyní můžou nastat dva případy. První možnost nastává v případě, kdy je délka stejná nebo delší. Druhá možnost nastává v případě, kdy je délka zadaného klíče menší než požadovaná druhá mocnina čísla na posuvníku. V prvním případě, kdy je délka klíče stejná nebo delší se pouze vybere určitý počet prvků z vektoru a se zbytkem klíče se již dále nepracuje. V druhém případě, kdy je zadaný klíč menší než je nutné, se vektor doplní posloupností celých čísel od jedné do požadovaného počtu. Nedoplňují se zde záměrně stejná čísla jako při doplnění textu, protože by se poté mohly vyskytnout v matici klíče stejné řádky. Pokud by tato skutečnost nastala, text se zašifruje, ale k matici klíče nepůjde vytvořit inverzní matici. Bez inverzní matice se text nedá dešifrovat.

Matice klíč

$$\begin{pmatrix} 67 & 97 & 101 & 115 \\ 97 & 114 & 1 & 2 \\ 3 & 4 & 5 & 6 \\ 7 & 8 & 9 & 10 \end{pmatrix}$$

Obr. 25. Matice klíč

Dalším krokem po zmáčknutí tlačítka Matice klíč je vytvoření z daného vektoru matice. Tato matice nesmí být singulární, determinant se nesmí rovnat nule.

Šifruj

$$\begin{pmatrix} 96 & -362 & -2484 & -2814 \\ 3212 & 4144 & 3854 & 4354 \\ 14430 & 17184 & 1244 & 1518 \\ 2170 & 2776 & 2888 & 3254 \\ 1199 & 1868 & 1731 & 1968 \end{pmatrix}$$

Obr. 26. Šifruj

Tlačítko Šifruj po stisknutí vynásobí Matici s Maticí klíče. Výsledná matice obsahuje šifrovaný text. Jedná se o zašifrovaný zadaný text pomocí zadaného klíče.

Šifrovaný text

{96, -362, -2484, -2814, 3212, 4144, 3854, 4354, 14430, 17184, 1244, 1518, 2170, 2776, 2888, 3254, 1199, 1868, 1731, 1968}

Obr. 27. Šifrovaný text

Tlačítko Šifrovaný text již pouze převede výslednou šifrovanou matici na vektor. Jedná se o formální zápis šifrovaného textu. Tento zápis je bezpečnější, nedá se rozpoznat rozměr šifrované matice.

7 NÁSTROJ PRO DEŠIFROVÁNÍ TEXTU

V následující kapitole bude popsán nástroj pro dešifrování textu. Popsán bude princip dešifrování doplněný ukázkami obdobně jako v předcházející kapitole.



Obr. 28. Posuvník

Zde má posuvník stejnou funkci jako při šifrování. Při dešifrování je důležité dobře nastavit pozici, aby se daný text správně dešifroval. Pro ukázkou v této kapitole bylo zvoleno jiné nastavení.

Šifrovaný text: `{442, -29424, -28828, -25339, -24505}`

Obr. 29. Šifrovaný text

Textové pole Šifrovaný text umožňuje zadat číselnou posloupnost pro dešifrování. Je však nutné zadávat posloupnost jako jeden vektor, jednotlivá čísla oddělovat čárkou a na začátku a na konci musí být zapsány složené závorky. Délka vektoru musí být násobkem hodnoty posuvníku. Tato syntaxe musí být dodržena. Pokud tomu tak není, program to vyhodnotí jako chybu. V případě, že není dodržena délka šifrovaného text v následujícím kroku po stisknutí tlačítka matice, se zadaný vektor nepřepíše do matice a to GeoGebra vyhodnotí jako chybu.

Matice

$$\begin{pmatrix} 5473 & 14663 & 13403 & 14221 & 13940 \\ -2804 & -826 & -812 & 3416 & -2362 \\ -28442 & -29424 & -28828 & -25339 & -24505 \end{pmatrix}$$

Obr. 30. Matice

Tlačítko matice slouží po stisku k vytvoření matice ze zadaného vektoru. Opět je zde důležité, aby byl správně nastaven posuvník, jinak se daná matice nevytvoří.

Klíč:

{109, 121, 115, 108, 105, 118, 101, 99, 32, 118, 32, 107, 97, 109, 105, 122, 111, 108, 101, 32, 122, 101, 108, 101, 110, 253}

Obr. 31. Klíč

Tlačítko klíč funguje úplně na stejném principu jako při šifrování textu. Zadá se Klíč. Potvrdí se stiskem tlačítka Enter. Text se převede do vektoru na ASCII znaky. Pro kontrolu se celý vektor vypíše pod textovým polem.

{109, 121, 115, 108, 105, 118, 101, 99, 32, 118, 32, 107, 97, 109, 105, 122, 111, 108, 101, 32, 122, 101, 108, 101, 110}

Obr. 32. Úprava klíče

Tlačítko Úprava klíče funguje naprosto stejně jako při šifrování textu. Na této ukázce je demonstrována úprava vektoru, když je klíč delší, než je zapotřebí. Vybere se pouze tolik prvků, kolik doopravdy klíč potřebuje.

$$\begin{pmatrix} 0.04 & -0.01 & -0.02 & -0.01 & -0.01 \\ 0.16 & -0.01 & -0.04 & -0.03 & -0.09 \\ -0.24 & 0.04 & 0.08 & 0.07 & 0.09 \\ 0.03 & -0.02 & -0.01 & -0.01 & 0 \\ 0.02 & 0 & -0.01 & -0.02 & 0 \end{pmatrix}$$

Obr. 33. Matice klíče

Tlačítko Matice klíč funguje podobně jako při šifrování. Zadaný vektor z předchozího kroku rozdělí podle daného čísla do řádků matice. Tuto matici navíc oproti šifrování invertuje a tím je získán klíč pro dešifrování.

Dešifruj

$$\begin{pmatrix} -13 & 11 & 129 & 11 & 1 \\ 12 & -64 & 16 & 5 & 19 \\ -64 & -39 & -39 & -64 & -64 \end{pmatrix}$$

Obr. 34. Dešifruj

Po stisku tlačítka Dešifruj se vynásobí Matice s Maticí klíče. Jedná se pouze o posunuté ASCII znaky. Pokud prvky matice tvoří desetinné čísla je jasné, že dešifrování neproběhlo správně.

Otevřený text

{83, 107, 225, 107, 97, 108, 32, 112, 101, 115, 32, 57, 57, 32, 32}

Skákal pes 99

Obr. 35. Otevřený text

Tlačítko Otevřený text převede matici z předcházejícího kroku na vektor. K číslům z tohoto vektoru se poté přičte číslo 96, aby byly správně převedeny jednotlivé prvky vektoru na správně znaky ASCII. Již posunutý vektor je vypisuje pod tlačítkem. Pod samotným vektorem se vypisuje již dešifrovaný otevřený text, který v ukázkovém případě zní: Skákal pes 99.

8 ZDROJOVÝ KÓD JEDNOTLIVÝCH TLAČÍTEK

V této kapitole bude stručně popsán zdrojový kód u jednotlivých tlačítek. Popisovány budou funkce a vše bude barevně odlišeno. Pokud dvě tlačítka mají stejnou funkci, popsáno bude jen jedno.

Programování jednotlivých tlačítek probíhá ve vlastnostech tlačítka v záložce Skriptování. Je zde nutné si přepnout na JavaScript. Je to hlavně kvůli cyklu for, který v GeoGebra Scriptu neexistuje.

Příkaz *ggbApplet.evalCommand* slouží k propojení GeoGebry a JavaScriptu. Příkaz *ggbApplet.getValue* převede hodnotu dané proměnné z GeoGebry do JavaScriptu. Příkaz z Geobery, který se běžně zadává do příkazového řádku, je označen hnědou barvou, je uzavřen v závorkách a oddělen uvozovkami např. ("*doplneni={}*"). Proměnné z JavaScriptu, které chce použít v GeoGebře je třeba zadávat ve formátu "+*promenna*+". Zeleně jsou označeny komentáře, kde */** značí začátek komentáře a **/* konec komentáře. Složené a kulaté závorky jsou označeny růžově. Další funkce pak libovolnými barvami. Každý řádek musí být ukončen středníkem. Zdrojový kód je minimalizovaný tak, aby zabíral co nejméně řádků. U každého řádku, pokud je to nutné, je stručný popis co daný řádek provádí. Nevýhodou psaní v JavaScriptu je, že se pro příkazy z GeoGebry musí používat anglické názvy.

8.1 Tlačítko Doplnění

```
ggbApplet.evalCommand("doplneni={}"); /* vytvoření prázdného vektoru */
ggbApplet.evalCommand("delka=Length[posunuti]");
a=ggbApplet.getValue("delka"); /* definice proměnné a, která udává délku vstupního textu
*/ r=ggbApplet.getValue("rozmer"); /* definice proměnné r, která nastaví hodnotu
posuvníku */
zbytek=a % r; /* funkce modulo spočítá zbytek po dělení čísla a s číslem r */
if(zbytek>0){ /* pokud je splněna podmínka vykoná se následující část */
ggbApplet.evalCommand("SetValue[doplneni,Join[posunuti,Sequence[-64,x,1,"+r+"-
" +zbytek+"]]]); /* doplnění vektoru a potřebným počtem čísel -64 pokud je to nutné */
else{ ggbApplet.evalCommand("SetValue[doplneni,posunuti]); /* pro neplatnou
podmínku se nic nedoplní */
```

8.2 Tlačítko Matice

```

ggbApplet.evalCommand("Matice={}"); /* vytvoření prázdné matice */

r=ggbApplet.getValue("rozmer"); /* převedení hodnoty posuvníku do Javy */

ggbApplet.evalCommand("delka=Length[posunuti]");

a=ggbApplet.getValue("delka"); /* převedení délky vstupního vektoru do Javy */

for(i=0;r*i<a;i++){ /* cyklus for – vytvoření vektoru z r-tice čísel z doplnění a přidání
tohoto vektoru na nový řádek Matice*/

ggbApplet.evalCommand("vektor=Take[doplneni, "+r+"*" +i+" +1, "+r+"*" +i+" "+" +r+"
");

ggbApplet.evalCommand("SetValue[Matice,Insert[{vektor},Matice, "+i+" +1]]"); };

```

Tlačítko Matice má stejnou funkci při šifrování i dešifrování, jen pracuje s odlišnými vstupními proměnnými.

8.3 Tlačítko Úprava klíče

```

r=ggbApplet.getValue("rozmer"); /* převedení hodnoty posuvníku do Javy */

ggbApplet.evalCommand("delkak=Length[seznamk]");

k=ggbApplet.getValue("delkak"); /* převedení hodnoty posuvníku do Javy */

if(k>= (r*r)){ /* pokud je délka klíče dostačující, vybereme ze seznamu jen potřebnou část*/

ggbApplet.evalCommand("seznamk_1=Take[ seznamk, 1, "+r*r+" ]"); }

else{ /* v případě krátkého klíče se seznam doplní posloupností přirozených čísel od 1 až
do čísla zbytekk */

ggbApplet.evalCommand("seznamk_2={}");

zbytekk=k % (r*r);

ggbApplet.evalCommand("SetValue[seznamk_2,Join[seznamk,Sequence["+r*r+" "-
"+zbytekk+"]]]"); };

```

Úprava klíče je stejná funkce jak pro šifrování, tak i pro dešifrování.

8.4 Tlačítko Matice klíč

```

ggbApplet.evalCommand("Klic_1={}"); /* vytvoření prázdné matice */

r=ggbApplet.getValue("rozmer"); /* převedení hodnoty posuvníku do Javy */

ggbApplet.evalCommand("delkak=Length[seznamk]");

a=ggbApplet.getValue("delkak"); /* převedení délky vstupního vektoru do Javy */

if(a>=(r*r)){ /* podmínka pro délku klíče */

for(i=0;i<r;i++){ /* při splnění podmínky se postupně vytváří vektor2 a přidává se na nový
řádek matice Klic_1*/

ggbApplet.evalCommand("vektor2=Take[seznamk_1,"+r+"*" +i+"+1,"+r+"*" +i+"+" +r
+"]");

ggbApplet.evalCommand("SetValue[Klic_1,Insert[{vektor2},Klic_1,"+i+"+1]"); } }

else{ for(i=0;i<r;i++){ /* při nesplnění podmínky se postupně vytváří vektor3 a přidává
se na nový řádek matice Klic_1 */

ggbApplet.evalCommand("vektor3=Take[seznamk_2,"+r+"*" +i+"+1,"+r+"*" +i+"+" +r
+"]");

ggbApplet.evalCommand("SetValue[Klic_1,Insert[{vektor3},Klic_1,"+i+"+1]"); } }

//ggbApplet.evalCommand("Klic=Invert[Klic_1]"); /* při dešifrování vytvoří inverzní
matici Klic, která dešifruje text */

```

Tlačítko Matice klíč má při šifrování i dešifrování stejnou funkci, jen při dešifrování se navíc provede poslední řádek, který vytvoří inverzní matici.

8.5 Tlačítko Šifruj

```

ggbApplet.evalCommand("Sifra={}"); /* vytvoření prázdné matice */

ggbApplet.evalCommand("SetValue[Sifra,Matice*Klic_1]"); /* Vynásobení matice Klic_1
a Matice, příkaz SetValue zajistí, aby se matice Sifra aktualizovala až po stisknutí tlačítka
*/

```

8.6 Tlačítko Šifrovaný text

```

ggbApplet.evalCommand("Vysledek={}"); /* vytvoření prázdné matice */
ggbApplet.evalCommand("delka=Length[posunuti]");
a=ggbApplet.getValue("delka"); /* převedení délky vstupního vektoru do Javy */
r=ggbApplet.getValue("rozmer"); /* převedení hodnoty posuvníku do Javy */
for(i=0;r*i<a;i++){ /* cyklus for z matice Sifra sestaví výsledný vektor */
ggbApplet.evalCommand("SetValue[Vysledek,Join[Vysledek,Element[Sifra,"+i+++"1]]]");
};

```

8.7 Tlačítko Dešifruj

```

ggbApplet.evalCommand("Unicode={}"); /* vytvoření prázdné matice */
ggbApplet.evalCommand("SetValue[Unicode,Matice*Klic]");

```

Tlačítko Dešifruj je velmi podobné jako tlačítko Šifruj, jen násobí mezi sebou jiné matice.

8.8 Tlačítko Otevřený text

```

ggbApplet.evalCommand("Vysledek={}"); /* vytvoření prázdné matice */
ggbApplet.evalCommand("Vektoru={}"); /* vytvoření prázdné matice */
ggbApplet.evalCommand("delka=Length[vstup]");
a=ggbApplet.getValue("delka"); /* převedení délky vstupního vektoru do Javy */
r=ggbApplet.getValue("rozmer"); /* převedení hodnoty posuvníku do Javy */
for(i=0;r*i<a;i++){ /* cyklus for převede matici Unicode na jeden řádek Vektoru */
ggbApplet.evalCommand("SetValue[Vektoru,Join[Vektoru,Element[Unicode,"+i+++"1]]]");
});
ggbApplet.evalCommand("Vysledek=round(Vektoru+96)"); /* přičtení k Vektoru číslo 96
kvůli posunu a následnému zobrazení do ASCII, nutno tento vektor zaokrouhlit kvůli
desetiným číslům, které by mohla způsobit špatné hodnoty */
ggbApplet.evalCommand("Text=UnicodeToText[Vysledek]"); /* funkce UnicodeToText
převede číselný vektor na text */

```

9 VYUŽITÍ APLIKACÍ

Ukázkové soubory práce s maticemi mohou sloužit jako výukový program nebo k ověření při výpočtu vlastních matic nebo jen ke generování jednotlivých zadání a ověření následného výpočtu.

Nástroj pro šifrování a dešifrování vytvořený v rámci této bakalářské práce by mohl být prakticky využitelný při psaní krátkých zpráv. Jedná se o běžnou zašifrovanou komunikaci mezi dvěma osobami Alicí a Bobem. Alice posílá zašifrované zprávy Bobovi a Bob si tyto zprávy dešifruje pomocí klíče, který mu Alice předá. A samozřejmě toto funguje i obráceně.

Protože tento vytvořený program spadá pod symetrickou kryptografii, je nutné si klíč předávat po zabezpečeném kanálu. Mohlo by dojít k jeho odposlechu a následnému rozluštění této šifry.

Při psaní delších zpráv může dojít k problému, že se daná matice nebo vektor budou překrývat s jiným textem nebo dokonce jejich část nebude viditelná vůbec.

Další příklad praktického využití je hashovací funkce. Jedná se o zašifrování textu, ale už nikoliv o jeho dešifrování. Tato situace může prakticky nastat, pokud při šifrování jsou v matici klíče dva nebo více stejných řádků. Z takové matice se totiž nedá spočítat inverzní matice a zašifrovaný text nejde rozšifrovat. Jedná se pouze o jednosměrnou operaci. Otázkou zůstává, jak moc účinný by byl v tomto případě útok hrubou silou resp., za jak dlouho by se útočnickovi podařilo text rozluštit.

Jedná se pouze o tzv. školní verzi, která umožňuje sledovat i jednotlivé kroky při šifrování a dešifrování. Při praktickém posílání zpráv, oba uživatelé zajímá jen otevřený a šifrovaný text a samozřejmě klíč. Průběžné kroky šifrování popř. dešifrování jsou pro běžného uživatele nepodstatné.

Obecně řečeno tento nástroj se dá použít pro šifrování psaného textu. Hlavně v případech, kde neklademe vysoké požadavky na bezpečnost. V případech, kdy klademe vysoké nároky na zabezpečení, např. v internetovém bankovníctví, by tento šifrovací a dešifrovací nástroj neobstál.

Při nasazení tohoto nástroje do praxe by se musel změnit výstup celé aplikace. Výstup jako plnohodnotná okenní aplikace. Zde v GeoGebře se jedná pouze o grafickou výstupní beta verzi.

ZÁVĚR

Cílem této bakalářské práce bylo vytvoření nástroje pro šifrování a dešifrování číselných posloupností v programu GeoGebra. Dalším hlavním požadavkem bylo demonstrovat možnosti práce s maticemi v GeoGebře.

Teoretickou část bakalářské práce jsem začal stručným popisem historie šifrování. Historie se dá rozdělit na čtyři hlavní části. První částí je období starověku. Do tohoto období spadají starověké civilizace Egypta, Mezopotámie a Indie. Zde se jedná pouze o tzv. obrázkové písmo. Nedá se vůbec hovořit o základech šifry. O několik století později jsou to národy starověkého Řecka a Říma. V tomto období se začínají používat jednoduché šifry např. posunovací šifry. Druhou významnou částí je středověk. Ve středověku došlo k největšímu rozmachu kryptologie. Používají se jednoduché substituční a transpoziční šifry. V tomto období se objevují i první kryptoanalytici, kteří pracují pro královské rody. Zvláště panovníci si uvědomují, že je třeba své tajné informace chránit šifrováním. Třetím, předposledním obdobím, je 20. století. K rozvoji kryptografie pomohla první světová válka ale i rozšíření bezdrátového telegrafu. Samotný vstup USA do první světové války pak byl důsledkem rozluštění telegramu. První světová válka vychovala i prvního z velikánů kryptologie dvacátého století - Williama Frederica Friedmana. Ten ovlivnil řadu lidí a kryptologie se stávala stále populárnější. Ve druhé světové válce vzrůstala potřeba šifrování. Šifrovalo se zejména pomocí šifrovacích strojů, nejznámější je šifrovací stroj Enigma. Poslední část historie se datuje od 20. století, kdy se začal při šifrování využívat veřejný klíč. Aplikovaná kryptologie byla dříve výsadou tajných služeb, armád a diplomacie. Během posledních let se stává věcí veřejnou a současně i výnosným obchodem. V další části jsem vyložil teorii šifrování přenosu číselných posloupností. Těchto principů jsem také využil při tvorbě nástroje pro šifrování a dešifrování. V další části teorie jsem stručně popsal program GeoGebra, který rozhodně neslouží jen jako geometrický program jak by se na první pohled mohlo zdát. Jedná se o plnohodnotný program, který lze využít na všech úrovních vzdělání. V praktické části jsem se zabýval možnostmi práce s maticemi v programu GeoGebra. Na ukázkových programech jsem demonstroval různé operace s maticemi. Jednalo se o součet matic, násobení matic, výpočet inverzní matice, výpočet transponované matice a výpočet determinantu matice. V těchto jednoduchých programech jsem se snažil o praktické využití a přínos pro uživatele. Uživatel má možnost zadat vlastní matice, nebo jen pracovat s předem definovanými typy matic, ve kterých lze volit čísla náhodně. Výstup je přehledně vytvořen

tak, aby se uživatel snadno orientoval. Další částí bylo vytvoření nástroje pro šifrování a dešifrování v programu GeoGebra. Je důležité na začátku nástroje zvolit hodnotu posuvníku, tato hodnota udává rozměr obou matic (šířku vstupu a rozměr čtvercové matice klíče). Uživatel chce zašifrovat text, zadá ho do textových polí a postupně stisknutím příslušných tlačítek dostává text předvedený do ASCII v dané matici. Po zadání příslušného klíče se opakuje podobný postup. Pokud je klíč příliš krátký, doplní se znaky, v opačném případě sama aplikace vybere potřebný počet prvků. Klíč se opět převede na matici. Při dešifrování se využívá inverzní matice klíče. Poté se obě matice vynásobí a výsledek se převede do vektoru. Velmi podobný postup se opakuje i při dešifrování, kdy vstupem je vektor číselné posloupnosti a výsledkem skoro čitelný text. Tato chyba je způsobena programem GeoGebra. Obě dvě aplikace jsou navrženy tak, aby byly maximálně přehledné a jednoduše se ovládaly. V poslední kapitole jsem popsal příklady využití těchto nástrojů vytvořených v rámci této bakalářské práce. Použití je podle mě velmi úzké a bylo by třeba zvolit lepší software pro grafický výstup. Nicméně daný výstup použitelný je jak pro pedagogické, tak i vědecko-výzkumné účely.

SEZNAM POUŽITÉ LITERATURY

- [1] Šifrování – historické zajímavosti. In: *Šifrování – historické zajímavosti* [online]. 2013 [cit. 2014-02-15]. Dostupné z: http://www.fi.muni.cz/~xpelanek/ucitele/data/janasifry/sifrovani_historicke_zajimavosti.pdf
- [2] Egyptské hieroglyfy. *Egyptologie* [online]. 2010 [cit. 2014-02-15]. Dostupné z: <http://www.egyptologie.cz/262/egyptske-hieroglyfy/>
- [3] Studijní materiály. *Portál ZČÚ* [online]. 2007 [cit. 2014-02-15]. Dostupné z: http://courseware.zcu.cz/wps/portal!/ut/p/c4/04_SB8K8xLLM9MSSzPy8xBz9CP00s3hvH1cLf2cLEwMLH38DA8_QQAMXTwtDAws_M_3g5GL9gmxHRQCvM1e8/
- [4] HOLOUBKOVÁ, Hana a Olga KOPROVÁ. Typografie elektronických dokumentů. *Klínové písmo* [online]. 2010 [cit. 2014-02-15]. Dostupné z: <http://hci.webpark.cz/klinove.html>
- [5] HANUŠ, Petr. Kryptologie - aneb šifry včera, dnes a zítra. *Petr Hanuš osobní stránky* [online]. 2009 [cit. 2014-02-16]. Dostupné z: <http://www.petrhanus.webovka.eu/download/Kryptologie.pdf>
- [6] Praktické základy Kryptologie a Steganografie. *Security-Portal.cz* [online]. 2009 [cit. 2014-02-17]. Dostupné z: <http://www.security-portal.cz/clanky/praktick%C3%A9-z%C3%A1klady-kryptologie-steganografie>
- [7] BEIL, Vojtěch. Z historie šifrování. *Šifrování* [online]. 2002,2003 [cit. 2014-02-17]. Dostupné z: http://sifry.sourceforge.net/extra_history.html
- [8] Autokey Cipher - Crypto Corner. *Http://crypto.interactive-maths.com* [online]. 2013 [cit. 2014-02-21]. Dostupné z: <http://crypto.interactive-maths.com>
- [9] Šifra z doby d'Artagnana (2. díl). *Technet.idnes.cz* [online]. 2004-09-29 [cit. 2014-02-25]. Dostupné z: http://technet.idnes.cz/sifra-z-doby-d-artagnana-2-dil-d1x-/tec_technika.aspx?c=A040915_5283441_bezpecnost

- [10] JAŠEK, Roman a David MALÁNÍK. Základy praktické kryptografie, Ukázky šifer, Současné typy šifer symetrických a asymetrických, eliptické křivky, hash algoritmy: Cvičení 3. In: *Moodle - vyuka.fai.utb.cz - Kurz Bezpečnost informačních systémů* [online]. 2014 [cit. 2014-03-07]. Dostupné z: <http://vyuka.fai.utb.cz/mod/resource/view.php?id=11508>
- [11] VONDRUŠKA, Pavel. Bealovy šifry. *Crypto-World 5/2007* [online]. 2007, č. 5 [cit. 2014-03-07]. Dostupné z: http://crypto-world.info/casop9/crypto05_07.pdf
- [12] GERGELITSOVÁ, Šárka. *Počítač ve výuce nejen geometrie: průvodce Geogebrou*. 1. vyd. Praha: Generation Europe, 2011, 247 s. ISBN 978-809-0497-436.
- [13] BALEE, Maram, K LAKSHMANA, Rao, Y RAMESH, Kumar. *Encryption and Decryption Algorithm using 2-D Matrices*. International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 4, 2013, 352-356 s. ISSN 2277 128X.
- [14] VELLAJKANNAN, B.; Dr. MOHAN, V.; GNANARAJ. V. *A Note on the Application of Quadratic Forms in Coding Theory with a Note on Security*, Int. J. Comp.Tech. Appl, Vol 1 (1), 2010, 78-87 s.
- [15] ZELENKA, Josef. *Ochrana dat: kryptologie*. Vyd. 1. Hradec Králové: Gaudeamus, 2003, 198 s. ISBN 80-7041-737-4.
- [16] VONDRUŠKA, Pavel. *Kryptologie, šifrování a tajná písmena*. 1. vyd. Praha: Albatros, 2006, 340 s. ISBN 8000018888.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ASCII American Standard Code for Information Interchange

OT Otevřený text

RSA Šifra s veřejným klíčem, iniciály autorů Rivest, Shamir a Adleman

ŠT Šifrovaný text

SEZNAM OBRÁZKŮ

Obr. 1. Hieroglyfy [3]	12
Obr. 2. Klínové písmo [4]	13
Obr. 3. Spartská Skytala [5]	14
Obr. 4. Šifrovací zařízení [1]	18
Obr. 5. Tabula recta [8]	18
Obr. 6. Jednoduchá šifra Marie Stuartovny [9]	19
Obr. 7. Baconova šifra [1]	20
Obr. 8. Jeffersonův váleček [1]	20
Obr. 9. Prostředí programu GeoGebra	27
Obr. 10. Ovládací panel – nástroje	28
Obr. 11. Vytvoření matice přes okno Tabulka	31
Obr. 12. Součet matic – ruční zadávání	32
Obr. 13. Součet matic – automatické zadávání	32
Obr. 14. Součin matic – ruční zadávání	33
Obr. 15. Součin matic – automatické zadávání	34
Obr. 16. Determinant matice	35
Obr. 17. Inverzní matice	36
Obr. 18. Transponovaná matice	37
Obr. 19. Posuvník	38
Obr. 20. Otevřený text	38
Obr. 21. Doplnění	38
Obr. 22. Matice	39
Obr. 23. Klíč	39
Obr. 24. Úprava klíče	39
Obr. 25. Matice klíč	40
Obr. 26. Šifruj	40
Obr. 27. Šifrovaný text	40
Obr. 28. Posuvník	42
Obr. 29. Šifrovaný text	42
Obr. 30. Matice	42
Obr. 31. Klíč	43
Obr. 32. Úprava klíče	43

Obr. 33. Matice klíče	43
Obr. 34. Dešifruj	44
Obr. 35. Otevřený text	44

SEZNAM TABULEK

Tab. 1. Otevřená a šifrovaná abeceda šifry Atbaš	13
Tab. 2. Polybiův čtverec	15
Tab. 3. Otevřená a šifrovaná abeceda Caesarovy šifry	16

SEZNAM PŘÍLOH

Příloha PI: Obsah vloženého CD-ROMu

PŘÍLOHA P I: OBSAH VLOŽENÉHO CD-ROMU

V papírovém obalu na zadní straně desek je vložen CD-ROM, který obsahuje bakalářskou práci v plném znění ve formátu PDF v souboru „fulltext.pdf“. Zbývající obsah CD-ROMu je rozdělen do následujících složek:

- Složka **GeoGebra** a v ní následující složky:
 - **Nástroj pro šifrování a dešifrování** – Obsahuje nástroje pro šifrování a dešifrování číselných posloupností v programu GeoGebra
 - **Práce s maticemi** – Obsahuje nástroje pro možnost práce s maticemi v programu GeoGebra