

**MATURITNÍ PRÁCE**

**NÁZEV PRÁCE**

**Tomáš Dvořák**

Vedoucí práce: Mgr. Karel Pazourek, Ph.D.

**V Českých Budějovicích 2022/2023**

**Prohlášení**

Prohlašuji, že jsem maturitní práci vypracoval/a samostatně s vyznačením všech

použitých pramenů.

V Českých Budějovicích dne …………………………

**Abstrakt**

Maturitní práce popisuje důležitost kryptografie a její využití v dnešní době. V obecné části se zaměřuje na obecnou teorii kryptografie, popisuje symetrické a asymetrické šifrování a jejich používání v komunikačních technologiích. V krátkosti zde také popisuje matematické postupy využívané k šifrování dat. Praktická část se zaobírá napsáním jednoduchého počítačového programu napsaném v jazyce Python. Tento program má využít teorii obecné části k zabezpečení dat pomocí blokové šifry.

**Klíčová slova**

Kryptografie, kryptogram, přenosový kanál, původce, adresát, kryptografický systém, klíč, symetrické šifrování, bloková šifra

**Poděkování**

Děkuji

Obsah

[1 Úvod 5](#_gjdgxs)

[2 Základny kryptografie 7](#_30j0zll)

[2.1 Pojmy 8](#_1fob9te)

[2.1.1 Autentizační kryptosystémy 9](#_3znysh7)

[2.2 Historie kryptografie 9](#_2et92p0)

[2.2.1 Klasické šifry 10](#_tyjcwt)

[*Image 3 Vigenèrova šifra* 11](#_3dy6vkm)

[3 Užití matematiky v kryptografii 11](#_1t3h5sf)

[3.1 Binární číselná soustava 11](#_4d34og8)

[3.2 Logické operace 12](#_2s8eyo1)

[3.3 Logické oprace 13](#_17dp8vu)

[3.4 Rotace 13](#_3rdcrjn)

[3.5 Substituce 13](#_26in1rg)

[3.6 Posuv 13](#_lnxbz9)

[4 Závěr 14](#_35nkun2)

[5 Použité zdroje 15](#_1ksv4uv)

[6 Seznam obrázků 16](#_44sinio)

[7 Seznam tabulek 17](#_2jxsxqh)

[8 Seznam příloh 18](#_z337ya)

[9 Přílohy 19](#_3j2qqm3)

# 

# Úvod

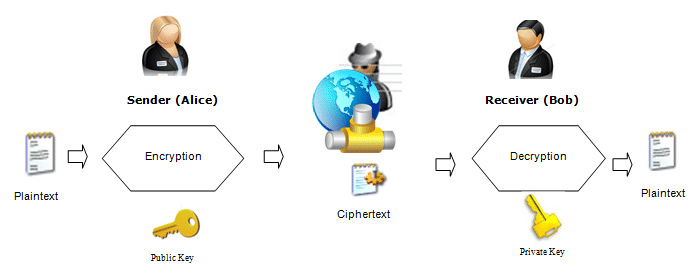
Informace se od dob vzniku písma stala kritickou. Lidé nežili všichni v jednom kmeni, ale dělili se na národy, které bojovaly o životní prostor. Informace zde představovaly výhody nad protivníkem v podobě vojenské informace, příkladem by mohla být bitva u Marathónu, kde zvítězili Řekové nad Peršany i díky poslovi, který donesl zprávu neboli informaci Spartě[[1]](#footnote-0), nebo informace hospodářské[[2]](#footnote-1) . Informaci bylo potřeba uchovat po určitou dobu, tzv. dobu rezistence kryptogramu a také ochránit před znemožněním doručení zprávy . V Mezopotámii k tomuto účelu využili kódování, které se později rozvinulo v kryptografii. Ve 20. a 21. století se stala kryptografie základní zárukou bezpečnosti v informačních systémech, které se rozšířily do všech oblastí společnosti od hospodářství, bankovnictví, vzdělání až po internetové seznamky představující oblast soukromého života (využití informačních technologií).

Moje maturitní práce se bude zabývat problematikou zabezpečení informací pomocí kryptografie. Tuto problematiku jsem si zvolil z důvodů potřeby šifrování dat v počítačových technologií a předcházení rozsáhlým únikům osobních dat a jejich následného zneužití. Mým dalším důvodem je užitečnost znalosti kryptografie při penetračním testování, neboť právě kontrola rezistence kryptogramu a následná schopnost opravy slabého šifrovacího algoritmu je klíčovým prvkem ochrany dat z databází uživatelů. Pokud se tato ochrana nezajistí dostatečně silná a v dostatečné míře, pak může nastat problém jako například v srpnu roku 2022[[3]](#footnote-2) , kdy byla napadena firma LastPass, která zajišťovala ukládání a ochranu hesel a sloužila tedy jako password manager[[4]](#footnote-3) .

Hlavními zdroji mé práce byla kniha Karla Burdy - Aplikovaná kryptografie[[5]](#footnote-4) a kurz TryHackMe[[6]](#footnote-5).

# Základny kryptografie

Kryptografie je věda zabývající se změnou informace do takové formy, ve které není možné informaci využít. Dělá tak za účelem utajení smyslu zprávy. Informace je totiž předávána přes přenosový kanál, který je veřejným médiem a je tedy možné ji zjistit nebo pozměnit. V obecném modelu máme tedy tři strany a přenosový kanál. První stranou je původce, tedy ten, kdo vytvořil informaci. Druhou stranou je adresát nebo adresáti, pro které je zpráva určena. Třetí stranou bývá útočník, kvůli kterému se informace šifruje. Obecně se tento model označuje jako “Alice and Bob”[[7]](#footnote-6). Alice je v tomto modelu původcem a Bob adresátem. Do systému byla postupem času zavedena další jména, jako třeba Mallory představující útočníka, David, Erin, atd. Schéma je poté vyzobrazeno na obrázku číslo 1.



*Image 1 - Schéma modelu Alice and Bob[[8]](#footnote-7)*

Na obrázku výše si můžeme všimnout kompletního procesu šifrování dat. V první části se nachází nešifrovaná zpráva (plaintext), kterou transformuje, tedy zašifruje (encryption) odesílatel (sender), neboli Alice, pomocí klíče. V tomto případě je zde uveden veřejný klíč (public key), díky čemuž lze odvodit, že jde o asymetrické šifrování se dvěma typy klíčů. Více informací o asymetrickém šifrování níže. Kryptogram (ciphertext) se poté pošle přes přenosový kanál, v tomto případě internet, kde je vystaven nebezpečí ze strany třetích stran. Na konci kryptogram rozšifruje (decryption) příjemce Bob (receiver) užitím soukromého klíče (private key).

Informace jsou data s určitým významem. Potřeba ochrany je důležitá, protože mohou představovat výrobní postup, vojenské strategie, osobní data, obchodní transakce a další. Pokud je útočník získá, může nimi zvrátit postup války nebo je prodat a tím poškodit firmy a jejich klienty. U informací se snažíme zajistit důvěrnost, tedy soukromnost informace, a autentičnost[[9]](#footnote-8), poukazující na nezměněnost původní informace.

Informace se dá také ochránit skrytím skutečnosti, že vůbec existuje. V praxi se informace skrývá například do obrázků nebo jiných dokumentů, které nejsou tak nápadné. Ke skrývání informací se používá steganografie, která však není obsahem této práce.

## Pojmy

Základní kryptografická jednotka neboli zpráva, s označením Z, se pomocí kryptografické funkce a klíče K změní na kryptogram s označením C. Zpráva může mít formu textu nebo sekvence čísel, které mohou představovat data šířená obvody v binárním kódu nebo samotné hodnoty, jako například datumy. Kryptografická funkce pak pomocí kryptografické transformace změní původní hodnoty na pseudonáhodnou řadu čísel nebo znaků, tedy kryptogram. Pseudonáhodnou se označuje proto, že nezúčastněnému pozorovateli tak mohou připadat, avšak skrývají význam, kterého je možno dosáhnout dešifrováním. Dva následující pojmy jsou označením pro použití kryptografické funkce. Šifrování je postup použití kryptografické transformace na zprávu, která ji změní na kryptogram. Dešifrováním se myslí postup opačný k šifrování, při kterém se kryptogram mění ve zprávu za použití inverzního zobrazení k funkci šifrovací. Pro šifrování byla zavedena zkratka ENC a dešifrování je určeno zkratkou DEC.

Celý proces se symbolicky zapisuje pomocí rovnic. První rovnice popisuje proces šifrování, kde kryptogram je vytvořen pomocí funkce ENC a údajů Z a KE, které v kontextu představují zprávu a šifrovací klíč, v tomto pořadí[[10]](#footnote-9).

Druhá rovnice znázorňuje proces dešifrování, popsáním zprávy Z rovností s funkcí DEC, jíž jsou zadány parametry kryptogramu C a dešifrovacího klíče KD.[[11]](#footnote-10)

### Autentizační kryptosystémy[[12]](#footnote-11)

Kryptosystém, který jsme nyní popisovali, se zabýval utajením obsahu zprávy, avšak jak již bylo zmíněno výše9, kryptografie může také sloužit k autentizaci zprávy, neboli zjištění jejího původu. Tento způsob se aplikuje především ke zjištění, zda-li zpráva skutečně pochází od odesílatele nebo byla v průběhu přenosu modifikována a tím porušena její integrita.

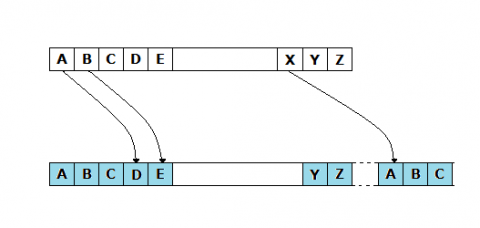
Při kontrole autenticity používáme dvě funkce. První se nazývá funkce pečetící PCT, která bere jako parametry zprávu Z a pečetící klíč KP. Výsledkem této zprávy je pečeť P, kterou použijeme jako vstupní parametr do ověřovací funkce.

Pro ověření zprávy s pečetí použijeme ověřovací funkci VER, jejíž vstupními parametry jsou zpráva Z, pečeť P a verifikační klíč KV. Výstupy této funkce existují pouze dva a to 1 nebo 0, které představují pravdivostní hodnoty. Důležité je zde poznamenat, že jediné, co v tomto případě určujeme, je pravdivost toho, jestli se zprávou bylo či nebylo nějak manipulováno, a ne to, v jakých místech byla zpráva změněna. Rovnice vyjadřující ověření zprávy vypadá takto:

## Historie kryptografie

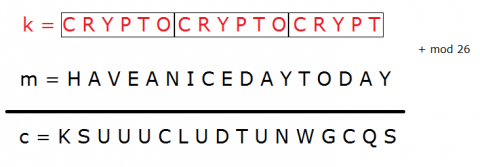
Slovo kryptografie pochází z latinského cryptographia, které má původ v řeckých slovech kryptos, které znamená skrytý, a graphia, znamenající psát nebo píši[[13]](#footnote-12). Avšak první použití se objevilo dlouho před samotným vznikem antického Řecka a to v Egyptě. V Egyptě bylo použití kryptografie zaznamenáno egyptskými hieroglyfy již v 19. století před naším letopočtem v hrobce kněze a šlechtice Chnumhotepa II. Nešlo o klasické použití šifrování k zakrytí obsahu, šlo spíše o změnu některých znaků tak, aby vypadaly důstojněji.

### Klasické šifry[[14]](#footnote-13)

Další použití kryptografie se objevuje okolo roku sto před naším letopočtem a je použito římským vojevůdcem Caesarem. Takzvaná Caesarova šifra[[15]](#footnote-14) je jednoduchou substituční šifrou. Princip fungování je takový, že pokud bychom pod sebe umístili dvakrát abecedu, tak spodní abecedu “posuneme” o několik pozic. První abeceda nám tak ukazuje písmeno zprávy a písmeno z druhé abecedy, které se nachází přímo pod ní, zobrazuje na jaký znak bude písmeno změněno v kryptogramu. V klasickém příkladu používaném Juliem Caesarem se abeceda posunula o tři místa doprava nebo doleva. Avšak můžeme abecedu posunout i o více míst, přesněji o počet znaků použité abecedy. Tento počet je klíčem Caesarovi šifry. Klíč však nemusí být nutně číslem. Lze použít i písmeno, které je písmenem přiřazeným písmenu A původní abecedy. Pokud tedy jako klíč vybereme písmeno C, tak je druhá abeceda posunuta o dvě písmena doprava, protože se písmeno C nachází o dvě písmena napravo od písmene A. Chybou celé šifry je její malý počet možných klíčů. Kvůli tomu musí být utajován proces šifrování, protože pokud je jednou odhalen, tak pro případného útočníka není velký problém vyzkoušet všechny možné klíče a získat šifrovanou informaci.

*Image 2 Caesarova šifra[[16]](#footnote-15)*

V šestnáctém století pak byla zavedena Vigenèrova šifra[[17]](#footnote-16), která funguje podobně jako Caesarova šifra. Klíč v ní však nepředstavuje jedno písmeno nebo číslo, ale dokonce slovo nebo celá fráze. Proces šifrování pak vypadá tak, že danou frázi napíšeme pod původní abecedu a pokud je fráze kratší než abeceda, tak jí opíšeme tolikrát za sebou dokud se nebude nacházet pod celou abecedou. Daná písmena fráze pod abecedou představují posun písmen, stejný jako v Caesarově šifře. Díky tomu pak různá písmena mohou nabývat stejných hodnot a existuje více možností šifrování.

*Image 3 Vigenèrova šifra([[18]](#footnote-17))*

# Užití matematiky v kryptografii

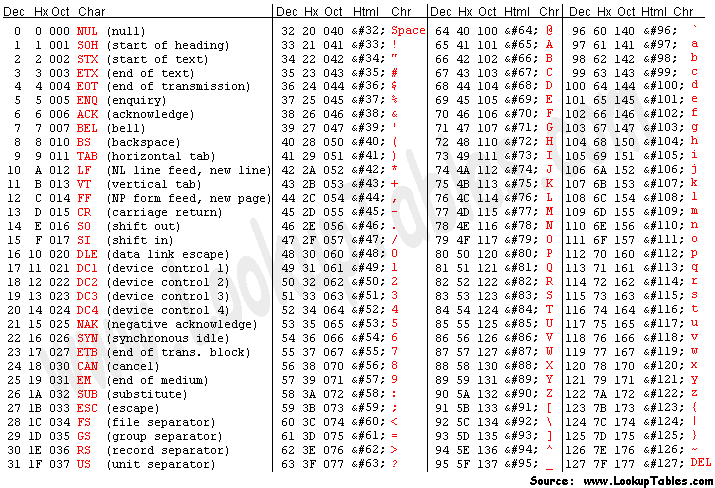
Informace se dají vyjádřit mnoha způsoby. Často se setkáme se slovním vyjádřením, které je pro nás jakožto lidská stvoření nejsrozumitelnější, avšak pro stroje je výhodnější numerické vyjádření. Proto kódujeme data do binární číselné soustavy, kterou stroje využívají pro ukládání dat v pamětích. Data v binární soustavě stroje zpracovávají pomocí matematických a logických operací, kterými se budeme zabývat v následujících kapitolách. Nejdříve si ale vysvětlíme binární soustavu a jakým způsobem v ní kódujeme data.

## Binární číselná soustava

Jak už název napovídá, tato soustava obsahuje dva stavy 1 a 0. Protože jsou tyto stavy dva, tak každá číslice, neboli bit, reprezentuje počet základů dvou na mocninu celého čísla začínajícího nulou zleva. Názorně to potom vypadá takto:

Zde vidíme nejprve rozvinutý zápis binární číslice[[19]](#footnote-18), kde jsou jednotlivá místa vyjádřena jako mocniny o základu dva. Jako druhé je výše za rovností zobrazeno samotné číslo v binární soustavě, a nakonec je zde číslo zapsáno v desetinné soustavě.

Data se kódují do binární soustavy pomocí standardizačních tabulek. V nich je zaneseno jaké číslo reprezentuje jaký znak, který je poté graficky zobrazen počítačem. V praxi se používá několik tabulek jako například ASCII[[20]](#footnote-19) (American Standard Code for Information Interchange) nebo UTF-8[[21]](#footnote-20). Níže je uvedena část ASCII tabulky, ve které je nejprve vidět vyjádření znaku v decimální soustavě, poté hexadecimální, osmičkové, HTML a nakonec samotný znak.



*Image 4 ASCII table[[22]](#footnote-21)*

## Logické operace

Logické operace jsou operace s bity, tedy hodnotami, které dokáží nabývat hodnot pravda/nepravda nebo 0/1.

První logickou operací jsou negace označování značkou NOT nebo symbolicky ¬. Negace mění pravdivostní hodnotu na hodnotu opačnou, tedy ¬1 = 0 a ¬0 = 1.

Konjunkce, značkou AND a symbolem ∧, pracuje se dvěmi pravdivostními hodnotami. Pokud jsou obě hodnoty 1, pak je výsledná pravdivostní hodnota také 1. Pokud však jen jedna z obou hodnot je rovna 0, pak je výsledná hodnota rovna 0. V českém jazyce se vyjadřuje konjunkce slovy “a zároveň”.

Disjunkce má značku OR a symbol ∨. Pokud je jedna nebo obě pravdivostní hodnoty rovna 1, pak se výsledná hodnota rovná taktéž 1. Jediný případ, kdy se disjunkce rovná 0 je, pokud jsou obě původní hodnoty 0. Disjunkce se označuje slovem “nebo”. Operace disjunkce se v kryptografii označuje jako blokový logický součet kvůli její podobnosti se součtem binárních čísel.

Poslední v praxi používanou logickou operací je exklusivní disjunkce se značkou XOR. Z tohoto důvodu se počítání s ní označuje jako xorování. Symbolicky se označuje znakem ⊕. Výsledná pravdivostní hodnota se rovná 1 za předpokladu, že hodnota obou původních výroků je opačná. Pokud se tedy pravdivostní hodnota obou výroků shoduje, tak výsledná hodnota se rovná 0.

Pravdivostní hodnoty již vysvětlených logických operací jsou shrnuty v následující tabulce.

| A | B | ¬A | A∧B | A∨B | A⊕B |
| --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 |

*Table 1 Tabulka pravdivostních hodnot logických operací*

## Modulo

Často používanou a téměř neopomenutelnou operací v kryptografii je modulo, symbolicky psáno jako mod. Zjednodušeně řečeno představuje modulo zbytek po celočíselném dělení. Rovnice vysvětlující tuto operaci pak vypadá takto:

V této rovnici se nám objevuje další typ operace vyjádřený pomocí závorek ⌊*x*⌋. Nazývá se dolní celá část čísla *x* a tato funkce nám říká, že číslu *x* je přiřazena hodnota nejbližšího celého čísla, které od čísla *x* leží směrem k -∞.[[23]](#footnote-22) Existence funkce dolní celá část čísla uvozuje i existenci funkce horní celá část čísla, která funguje podobně, ale přiřazuje číslu *x* celé číslo ležící směrem k +∞.

Operace modulo skýtá zvláštní vlastnost. Výsledkem je totiž číslo z množiny celých čísel, počínajíc nulou a končíc číslem o jedna menším než číslo *b*. Tento jev se dá vysvětlit tak, že pokud číslo dělíme samo sebou, tak zbytek po dělení je roven nule. Poté můžeme dělenec zvedat o 1, ale jen do té doby, než bude zbytek roven *b*-1. Pak totiž přičteme k dělenci další 1 a bude opět dělitelný číslem *b* a zbytek bude opět 0. Tato posloupnost se neustále opakuje a vytváří nám takzvané zbytkové třídy, která rozděluje čísla do množin. Množiny jsou charakterizovány tak, že v každé mají čísla stejný zbytek po dělení určitým číslem, které určuje počet možných zbytků a tedy i počet zbytkových tříd.

Při praktické implementaci se pak setkáváme ještě s jedním využitím a to s rovnicí *c* mod 2x. V této rovnici představuje *c* binární číslo, které je binárním součtem *a* + *b*. Proměnná *x* je celým číslem větším než nula a znázorňuje počet posledních bitů čísla *c*. Výsledek této rovnice je tedy posledních *x* bitů binárního součtu čísel *a* + *b*.

## Permutace

Permutace značí různá uspořádání systému, matice, množiny[[24]](#footnote-23). Pokud je již množina prvků uspořádána, poté její další permutací může být jiné uspořádání prvků. Ukázku permutace jsme mohli vidět již výše při vysvětlování Caesarovi šifry(), kdy novou permutací původní množiny znaků, neboli abecedy, byla permutace se všemi členy posunutými o tři znaky. Avšak při vysvětlování Caesarovi šifry byl použit termín “substituce”. Je důležité vysvětlit rozdíl mezi termíny permutace a substituce z důvodu jejich vysoké podobnosti. Při substituci máme dvě množiny, kde každému prvku z první množiny přiřadíme jeden nebo více prvků množiny druhé. U permutace toto však neplatí, neboť každému prvku musí být přiřazen pouze jeden prvek z druhé množiny. Nemůže se tedy například stát, že písmenu U by byla přiřazena písmena T a B, protože by nebylo možné invertovat tuto permutaci.

Dále, proto aby byla daná permutace bezpečná, tak musí splňovat tři následující pravidla. Zaprvé, permutace musí být určena klíčem. Pokud je však zjištěn klíč, tak se permutace bude dát rozluštit, a proto se musí klíč udržovat v tajnosti. Zadruhé, různé klíče musí vytvářet různé permutace. Pokud by totiž dva a více klíčů vedlo ke stejné permutaci, potom by to znamenalo, že existuje méně možných výsledků a tedy by se dala zpráva uhádnout i bez znalosti správného klíče. A nakonec zatřetí, permutace by měla vypadat náhodně. Pokud by tak totiž nevypadala, tak by bylo pro útočníka snazší uhádnout, jak šifra funguje, a nebo dokonce odvodit z jednoho písmena písmeno jiné jako v případě Caesarovi šifry.[[25]](#footnote-24)

### Rotace[[26]](#footnote-25)

Rotace je speciálním typem permutace. Označuje se písmeny RORk(V) nebo ROLk(V). Tyto názvy pochází z anglického “rotation on right” a “rotation on left”, neboli rotace nalevo a napravo. V praxi nám tento zápis říká, že máme vzít pravou (popřípadě levou) k-tici bitů bloku V a přesunout ji nalevo (v opačném případě napravo).

### Posun[[27]](#footnote-26)

Posun je dalším často typem substituce s označením SHLk(V). Zkratka pochází taktéž z angličtiny ze spojení “Shift on left”, tedy posun nalevo. Stejně jako u rotace, i k posunu existuje opačná funkce posun napravo s označením SHRk(V). Posun v základu funguje tak, že u posunu nalevo vezmeme k-tici bitů z pravé části bloku V. Tato část nám bude vytvářet začátek výsledného bloku W a poté doplňíme do bloku W tolik nul, aby se velikostí rovnal bloku V. Jak již bylo zmíněno na začátku odstavce, posun není permutací, ale je substitucí. To proto, že zbytek bitů je doplněn nulami, neboli více prvkům původní množiny je přiřazena stejná hodnota, což činí posun substitucí.

## Další matematické operace a funkce

Řetězení používáme na skupiny hodnot. Říká nám, že máme vzít první sadu hodnot a přidat k ní druhou sadu hodnot. Ve vzniklé posloupnosti se tak nachází hodnoty z obou posloupností v pořadí, ve kterém se hodnoty nacházely v jednotlivých posloupnostech a tak, jakým pořadím byli spojeny posloupnosti. Hodnoty mohou být jakékoliv znaky. Zápis této operace vypadá následovně:

Pokud by se tedy A rovnalo (9, x) a B se rovnalo (7, \*), potom C bude (9, x, 7, \*).

A nyní si vysvětlíme blokové aritmetické operace. To jsou aritmetické operace, například násobení nebo sčítání, které jsou provedeny pouze na blok čísel určité velikosti. Udržení velikosti bloku dosáhneme tak, že s výsledkem dané operace provedeme ještě operaci modulo 2x, kde x představuje velikost výsledného bloku v bitech. Proč použijeme zrovna tento postup jsme si vysvětlili výše ().

Konečné těleso, neboli typ algebraické struktury, je množinou, na které jsou definované dvě oparace, vůči kterým je tato množina uzavřená. To znamená, že pokud je některá z těchto operací na konečném tělese provedena, pak je výsledkem opět prvek z této množiny[[28]](#footnote-27). Pro konečné těleso platí, že je vůči sčítání a násobení uzavřené. Dále platí asociativní, komutativní a distributivní zákon a existují zde neutrální prvky 0 a 1, které implikují existenci opačných a inverzních prvků[[29]](#footnote-28). Asociativní, komutativní a distributivní zákony zde nebudeme dále rozebírat, protože jsou příliš vzdáleny obsahu této práce, avšak neutrální prvky si probereme podrobněji. Neutrální prvek je prvkem, který nezmění číslo, se kterým provádíme aritmetickou operaci. Pokud tedy k číslu *a* přičítáme neutrální prvek, pak bude výsledkem opět číslo *a*. Neutrálním prvkem je tedy při sčítání 0 a při násobení 1.

Zde vidíme, že pokud bychom na rovnice provedli ekvivalentní úpravy, tak dostáváme další dvě rovnice, pomocí kterých si můžeme definovat odčítání a dělení na konečném tělese.

Pokud tedy použijeme sčítání a odečítání na konečném tělese k zašifrování dat, pak máme díky opačnému prvku *-a* a inverznímu prvku *a-1* možnost provést inverzní operaci, a získat tak původní hodnoty před zašifrováním.

Vlastností konečných těles je, jak již název napovídá, omezený počet prvků této množiny. Obecně se konečné těleso zapisuje jako GF(*pn*), kde GF je vzato z angličtiny a znamená “Galois Field”, neboli Galoisovo těleso (jiné označení konečného tělesa), a *pn* je počtem prvků. Číslo *p* je prvočíslem a *n* je kladné celé číslo. V praxi se však používají pouze dva typy těles a to s počtem prvků *p* a 2*n*. U tělesa GF(*p*) jsou prvky z množiny {0,1,2,...,*p*-1} a operace násobení, sčítání a odčítání lze provést klasickým způsobem s tím, že se poté ještě na výsledku provede operace modulo *p.* Dělení je potřeba provést pomocí násobení inverzním prvkem. Tělesa s počtem prvků 2*n* jsou pak ideální pro počítačové užití, protože se dají reprezentovat binárním číslem, kde *n* je počet bitů.

# Symetrické kryptosystémy

# Závěr

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est l

# Použité zdroje

Zdroje je možné třídit, např. v následujícím pořadí:

* Nevydané prameny (archivní prameny, deníky, dopisy v soukromém vlastnictví apod.)
* Vydané prameny (edice kronik, korespondence, statistické ročenky apod.)
* Literatura (knihy, noviny, časopisy)
* Internetové zdroje

Zdroj použitý pro psaní speciálních znaků:

SZYNALSKI, Tomasz P. Type mathematical symbols - online keyboard. Typeit [online]. [cit. 2023-01-20]. Dostupné z: <https://math.typeit.org/>

String to binary in python:

[Python | Convert String to Binary - GeeksforGeeks](https://www.geeksforgeeks.org/python-convert-string-to-binary/)

# Seznam obrázků

# Seznam tabulek

# Seznam příloh

# Přílohy

1. (CARTWRIGHT, Mark. Battle of Marathon [online]. 19 May 2013 [cit. 2023-01-04]. Dostupné z: [https://www.worldhistory.org/marathon/](https://www.worldhistory.org/marathon/#inbox/_blank)) [↑](#footnote-ref-0)
2. ( šifrovaný popis výroby keramiky, Mezopotámie) [↑](#footnote-ref-1)
3. (WINDER, Davey. New LastPass Hack Confirmed—Here’s What We Know So Far. Forbes [online]. Forbes, 2022, Dec 1, 2022,07:07am EST [cit. 2023-01-07]. Dostupné z: https://www.forbes.com/sites/daveywinder/2022/12/01/new-lastpass-hack-confirmed-heres-what-we-know-so-far/) [↑](#footnote-ref-2)
4. (Password manager. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2023-01-07]. Dostupné z: <https://en.wikipedia.org/wiki/Password_manager).> [↑](#footnote-ref-3)
5. BURDA, Karel. Aplikovaná kryptografie. Brno: VUTIUM, 2013. ISBN 978-80-214-4612-0. [↑](#footnote-ref-4)
6. SAVPANI, Ashu a Ben SRING. TryHackMe. TryHackMe [online]. 128 City Road, London, EC1V 2NX: TryHackMe, 2022 [cit. 2023-01-07]. Dostupné z: https://tryhackme.com/ [↑](#footnote-ref-5)
7. Alice and Bob. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2023 [cit. 2023-01-08]. Dostupné z: https://en.wikipedia.org/wiki/Alice\_and\_Bob [↑](#footnote-ref-6)
8. Public-Key-cryptographic-scheme. In: ResearchGate [online]. ResearchGate, 2023, Mar 23, 2018 [cit. 2023-01-08]. Dostupné z: https://www.researchgate.net/profile/Mohammad\_Alia/publication/314059316/figure/fig3/AS:465905328103425@1488091894098/Public-Key-cryptographic-scheme.png [↑](#footnote-ref-7)
9. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 22. ISBN 978-80-88168-52-2. [↑](#footnote-ref-8)
10. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 20. ISBN 978-80-88168-52-2. [↑](#footnote-ref-9)
11. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 21. ISBN 978-80-88168-52-2. [↑](#footnote-ref-10)
12. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 22. ISBN 978-80-88168-52-2. [↑](#footnote-ref-11)
13. Cryptography | Etymology, origin and meaning of cryptography by etymonline. Etymonline.com: Online Etymology Dictionary [online]. Douglas Harper, 2022, May 26,2018 [cit. 2023-01-13]. Dostupné z: https://www.etymonline.com/word/cryptography [↑](#footnote-ref-12)
14. SIDHPURWALA, Huzaifa. A Brief History of Cryptography. RedHat [online]. © 2023 Red Hat, 2023, January 12, 2023 [cit. 2023-01-13]. Dostupné z: https://www.redhat.com/en/blog/brief-history-cryptography [↑](#footnote-ref-13)
15. Caesarova šifra. Matweb [online]. matweb, 2022 [cit. 2023-01-13]. Dostupné z: https://www.matweb.cz/caesarova-sifra/ [↑](#footnote-ref-14)
16. Caesar\_cipher\_1. In: RedHat [online]. © 2023 Red Hat, 2023 [cit. 2023-01-13]. Dostupné z: https://www.redhat.com/rhdc/managed-files/styles/wysiwyg\_full\_width/private/caeser-cipher\_1.png?itok=0A1OmiFq [↑](#footnote-ref-15)
17. Vigenerova šifra. Matweb [online]. matweb, 2022 [cit. 2023-01-13]. Dostupné z: https://www.matweb.cz/vigenerova-sifra/ [↑](#footnote-ref-16)
18. Vigenere\_1. In: RedHat [online]. © 2023 Red Hat, 2023 [cit. 2023-01-13]. Dostupné z: https://www.redhat.com/rhdc/managed-files/styles/wysiwyg\_full\_width/private/vingenere\_1.png?itok=jPgDwoyu [↑](#footnote-ref-17)
19. Binary Number System. Byju's [online]. © 2023, BYJU'S. All rights reserved., 2023 [cit. 2023-01-14]. Dostupné z: https://byjus.com/maths/binary-number-system/ [↑](#footnote-ref-18)
20. ASCII Table - ASCII Character Codes, HTML, Octal, Hex, Decimal. Asciitable [online]. © https://www.asciitable.com [cit. 2023-01-14]. Dostupné z: https://www.asciitable.com/ [↑](#footnote-ref-19)
21. Unicode/UTF-8-character table. Utf8-chartable [online]. [cit. 2023-01-14]. Dostupné z: https://www.utf8-chartable.de/ [↑](#footnote-ref-20)
22. Asciifull. In: Asciitable [online]. © https://www.asciitable.com [cit. 2023-01-14]. Dostupné z: https://www.asciitable.com/asciifull.gif [↑](#footnote-ref-21)
23. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 35. ISBN 978-80-88168-52-2. [↑](#footnote-ref-22)
24. Permutation. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2023 [cit. 2023-01-25]. Dostupné z: https://en.wikipedia.org/wiki/Permutation [↑](#footnote-ref-23)
25. In: AUMASSON, Jean-Philippe. Serious Cryptography: A Practical Introduction to Modern Encryption. 245 8th Street, San Francisco, CA 94103: No Starch Press, 2018, s. 4-5. ISBN 978-1-59327-826-7. [↑](#footnote-ref-24)
26. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 39-40. ISBN 978-80-88168-52-2. [↑](#footnote-ref-25)
27. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 39-40. ISBN 978-80-88168-52-2. [↑](#footnote-ref-26)
28. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 41. ISBN 978-80-88168-52-2. [↑](#footnote-ref-27)
29. BURDA, Karel. In: Kryptografie okolo nás. Milešovská 5, 130 00 Praha 3: CZ.NIC, z. s. p. o., 2019, s. 42. ISBN 978-80-88168-52-2. [↑](#footnote-ref-28)