Veřejně tajné šifrování, aneb Malá násobilka jinak

podklady pro U3V 2020

Kódování: Písmena a znaky textu převedeme na čísla, např. známý UNICODE.
(Lze také ovšem navíc tajně očíslovat některá často se vyskytující slova, i věty či zprávy.)

Zpráva bude přenesena jako (hodně veliké) ***číslo*** – event. po částech, tj. jako skupina čísel.

Jak přenést veliké číslo Q (tj. mou zprávu) příteli tak, aby ho ***špion na poště*** nerozluštili?

Jeden jednoduchý způsob: mám svůj ***tajný kód,*** číslo ***X***, které znám jen já. Já ho ke své zprávě přičtu (třeba bez přenosů = po cifrách, modulo 10), přítel ho pak musí odečíst. Např. X= ***3579***:

zpráva: Q = 479 056 002 006 001 105
můj kód: X = ***357 9***35 79***3 579*** 357 9***35*** a sečtu po cifrách (vlastně: cifry modulo 10):
pošlu: Q' = 726 981 795 575 358 030

Co s tím může udělat špión?

☺ Není-li X moc velké, mohl by vyzkoušet všechna X'=0000, 0001, 0002, 0003, atd. Ale:

☻ Špion neví, kolikaciferné mám X.

☻ Je-li číslo X dost velké, např. 10^{99} (100-ciferné), nemá ani šanci „vyzkoušet všechna X“. Náš Vesmír totiž trvá zatím jen asi 12×10^9 let $\approx 4\times10^{16}$ sekund. = 40 000 000 000 000 000 s.

☻ Ale tím vznikl nový problém: ***jak poslat svůj tajný kód X*** příteli, aby ho špion nezjistil?

Řešení a návod Připomeňme si:

$2 + 2 + 2 + 2 + 2 = \mathbf{2 \times 5}$ (= 10) , **násobení** je opakované sčítání. $(a \times b) \times c = a \times (b \times c)$.

$2 \times 2 \times 2 \times 2 \times 2 = \mathbf{2^5}$ (= 32) , **mocnění** je opakované násobení. $a^b \times a^c = a^{b+c}$; $(a^b)^c = a^{b\times c}$

1) Je veřejně známé velké číslo – základ Z.

2) Já si vymyslím své velké číslo A, spočítám $Z^A = R$ a pošlu příteli.

3) Přítel nezná A, ale zná nyní $R = Z^A$ (špión taky!).

4) Přítel si vymyslí své velké číslo B a spočítá dvě čísla: $\boldsymbol{S=Z^B}$ a jen pro sebe $W=R^B = Z^{A\times B}$ (to špión nemůže, protože nezná B). Přítel mi pošle jenom $S=Z^B$.

5) Já neznám B, ale znám nyní $S=Z^B$ (špión taky!)

6) Já si spočítám $W'=S^A=Z^{B\times A}$ (špión nemůže, nezná A).

7) Moje W'=přítelovo W! Platí totiž $W'=(S^A) = Z^{B\times A}$, a dále $W=(R^B) = Z^{A\times B}$, a $Z^{B\times A}=Z^{A\times B}$.

Tím je úloha vyřešena: já i přítel ***známe tajné W***, špión ne. Zopakujme si, co kdo známe:

já	Z	A	$R\ (= Z^A)$		$W = S^A = (Z^B)^A = Z^{B\times A}$
přítel	Z	B		$S\ (= Z^B)$	$W' = R^B = (Z^A)^B = Z^{A\times B} = W$
špión	Z		$R\ (= Z^A)$	$S\ (= Z^B)$	?? $R\times S = Z^{(A+B)} \neq W$!! nepomůže

Dva možné problémy:

1) Špión by mohl ***rozložit R na prvočinitele***, a tím rovnou nalézt mé A. To jde např. tak, že se opakovaně dělí R číslem Z tolikrát, kolikrát to jde beze zbytku; počet dělení je právě A.

To však vázne na tom, že A je veliké, má třeba 30 cifer (10^{30}). Kdyby na PC trvalo dělení 1 takt, trvala by miliarda (10^9) dělení celkem 1 sekundu. Ale už miliarda miliard (10^{18}) trvá 30 let, a našich tisíc miliard miliard miliard (10^{30}) trvá 25 000× déle, než zatím trvá Vesmír. Ale co kdyby našel rychlejší způsob rozkladu než postupné dělení po jednom Z?

Vylepšení počítáme „***modulo***“. Barbína, co neumí do pěti počítat, počítá modulo 5, a to takhle:

x:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14..
B(x):	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4..

Barbína tedy bere vždycky jen ***zbytek*** po dělení ***pěti*** (tomu se říká ***výpočty modulo* 5**).

Malá sčítanka $x+y$:

y→	0	1	2	3	4
x↓0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

Malá násobilka $x \times y$:

0	1	2	3	4
0	0	0	0	0
0	1	2	3	4
0	2	4	1	3
0	3	1	4	2
0	4	3	2	1

Malá dělenka x/y :

1	2	3	4
0	0	0	0
1	3	2	4
2	1	4	3
3	4	1	2
4	2	3	1

Dělení je vždy beze zbytku, např. 1/2 = 3, protože 3×2 = 1 (= 6 = 5+1), podobně 1/3 = 2 atp. Ověřte, že 2/3=4; 3/4=2 apod. Nejsou tu tedy ani zlomky, ani záporná čísla: –1 = 4 atd.

Lze počítat „**modulo cokoli**“. Modulo 10 je snadné: jen poslední cifra výsledku (proč?). Ale pro nás je zvláště výhodné „***modulo prvočíslo***“, protože nic neztrácí: $a \times b = 0$ jen tehdy, když $a = 0$ nebo $b = 0$. Násobilka pak vyčerpá všechna čísla a každé dělení vyjde beze zbytku. (Naproti tomu např. při počítání modulo 10 (=5×2) je 4 × 5 = 20 = 0 a např. 3/5 či 3/2 nejde.)

Při „modulo *P*“ jsme sice omezeni na čísla nanejvýš rovná *P*, ale má-li *P* třeba 100 cifer, tak toto omezení nevadí. Zato rozklad na prvočinitele ztrácí smysl: při počítání „modulo prvočíslo *P*“ totiž vychází každé dělení beze zbytku! Metoda rozkladu tedy selže principiálně.

2) **Práce s obrovskými čísly** *W, W′* **dlouho trvá.** Např. číslo $M = 10^{100}$ má cifer jen 100, což nevadí pro PC. Ale $N = 10^M$ má cifer $M = 10^{100}$; na to PC nestačí (viz doba trvání Vesmíru).

2a) Počítání modulo *P* problém řeší: nyní je každé číslo menší než *P* (a mezivýsledky než P^2).

2b) Umocnění *P* na miliardu ($=10^9 \approx 2^{30}$) převedeme následujícím trikem na pouhých nejvýš 2×30 násobení *P*. (Podobně umocnění na $10^{100} \approx 2^{333}$ potřebuje jen nejvýš 2×333 násobení.)

Názorná ukázka pro *P* = 100 (to sice není prvočíslo, ale „modulo 100“ se hezky počítá):
Spočtěte 47^{83} modulo 100.
Rozložíme exponent 83 = 1 + 2 + 16 + 64 postupným půlením se zbytkem na součet mocnin dvou („mocnitel“). V tabulce je vyznačeno ***tučně*** vždy, když je při „půlení“ zbytek 1:

$83_{10} = \mathtt{IOIOOII}_2 = 1 \times \mathbf{64} + 0 \times 32 + 1 \times \mathbf{16} + 0 \times 8 + 0 \times 4 + 1 \times \mathbf{2} + 1 \times \mathbf{1}$

Poté, vše modulo *P*, násobíme základ 47 postupně sám sebou (= „mocnina“ = základ 47 na „mocnitel“, tj. 47 na mocninu dvou). Vybrané mocniny spolu nakonec znásobíme („součin“).

83	***41***	***20***	10	5	***2***	1	***0***	půlení
	1	***2***	4	8	***16***	32	***64***	mocnitel
	47	$47\times47=_{22}\mathbf{09}$	$09\times09=_{00}81$	$81\times81=_{65}61$	$61\times61=_{37}\mathbf{21}$	$21\times21=_{04}41$	$41\times41=_{16}\mathbf{81}$	mocnina
		$47\times\mathbf{09}=_{04}23$			$23\times\mathbf{21}=_{4}83$		$83\times\mathbf{81}=_{67}23$	součin

Výsledek: $47^{83} = 23 \bmod 100$ (samo 47^{83} má přitom 139 číslic)

Pro U3V upravil ☺ jan.obdrzalek@mff.cuni.cz