Výpočet $\binom{N}{K} \pmod{M}$

Kombinační číslo $\binom{N}{K} = \frac{N!}{K!(N-K)!} = \frac{N(N-1)(N-2)\dots(N-K+1)}{K!},\ K = min(K,(N-K))$

je **přirozené číslo** definované pro $N \geq K,\ N,K \in \mathbb{N} \cup \{0\}$ a platí: $\binom{N}{0} = \binom{0}{0} = 1.$

Podle toho, zda N je větší nebo menší než M a podle toho, zda M je nebo není prvočíslo volíme různé způsoby výpočtu. Princip výpočtu v každé situaci je založen na vytvoření binomického koeficientu ve tvaru součinu jednotlivých prvků výsledku [4].

1. Je li M číslo složené, rozložíme M na součin nesoudělných mocnin prvočinitelů, použijeme čínskou větu o zbytcích [1] a výsledek vytváříme z dílčích výpočtů N nad K modulo jednotlivými mocninami prvočinitelů.

2. Jestliže M je prvočíslo a je menší než N, potom pro výpočet použijeme Lucasovu větu. [2]

3. Pokud je M prvočíslo, ale není větší než N, použijeme postup [4].

1.1 M není prvočíslo

Pokud máme počítat modulo M, kde M není prvočíslo, rozložíme M na prvočísla a použijeme Čínskou větu o zbytcích [1] k výpočtu binomického koeficientu modulo M.

$$M = p_1^{\ell_1} \dots p_m^{\ell_m},\ \text{kde } p_1^{\ell_1} \dots p_m^{\ell_m} \text{ jsou nesoudělná, určíme } x_1, \dots x_m :$$

$$\begin{aligned} \binom{n}{k} \pmod{p_1^{\ell_1}} &= x_1 \\ &\vdots \\ \binom{n}{k} \pmod{p_m^{\ell_m}} &= x_m \end{aligned}$$

Potom $\binom{n}{k} \pmod{M} = x_1 q_1 + \dots + x_m q_m$, kde $q_i \equiv 1 \pmod{p_i^{\ell_i}} \wedge q_i \equiv 0 \pmod{p_j^{\ell_j}}$ pro $j \neq i$.

Čísla $q_1, \dots, q_m$ nalezneme například takto : vytvoříme číslo $P_i = \prod_{j \neq i} p_j^{\ell_j}$ a položíme $q_i = P_i t_i$, kde t_i je inverzní prvek k číslu P_i v $(\mathbb{Z}_{p_i^{\ell_i}} \odot)$) . Platí $p_i^{\ell_i} \alpha + P_i t_i = 1$. Inverzní prvek můžeme určit rozšířeným Eukleidovým algoritmem, kterým najdeme vyjádření největšího společného dělitele dvou čísel jejich lineární kombinací.

1.2 M je prvočíslo

V roce 1878 Lucas navrhl metodu výpočtu binomických koeficientů modulo prvočíslo p (např. [2]):

$$\binom{n}{k} \equiv \prod_{i=0}^{\ell} \binom{n_i}{k_i} \pmod{p},$$

kde $n = n_\ell p^\ell + n_{\ell-1} p^{\ell-1} + \dots + n_1 p + n_0$ a $k = k_\ell p^\ell + k_{\ell-1} p^{\ell-1} + \dots + k_1 p + k_0$ jsou koeficienty rozkladu n a k v p-kové číselné soustavě, (reprezentace n a k v tělese $\mathbb{Z}_p$).
Pokud se v součinu vyskytuje aspoň jedna dvojice $n_i,\ k_i$ taková, že $n_i < k_i$, výsledek je nula.
To znamená, že je nutné spočítat binomické koeficienty pro čísla menší nebo rovna p.

1.3 M je mocnina prvočísla

Po více než sta letech mnozí autoři zobecnili Lucasovu větu na mocniny prvočísla, např. K.S.Davis a W.A.Webb nebo A. Granville [3, 2]:

$$\binom{np^{\ell+s} + n_0}{kp^{\ell+s} + k_0} \equiv \prod_{i=0}^{\ell} \binom{n_i}{k_i} \pmod{p}^{\ell+1},$$

kde ℓ, n, k, n_0, k_0 a s jsou přirozená čísla, taková, že $0 < n_0, k_0 < p^s$.

Princip výpočtu [4]

A. Vytvoření výsledku ve tvaru součinu mocnin prvočísel.

Binomický koeficient zapíšeme ve tvaru $\binom{N}{K} = N^1 \cdot (N-1)^1 \cdot \dots \cdot (N-K+1)^1 \cdot K^{-1} \cdot (K-1)^{-1} \cdot \dots \cdot 2^{-1}$.

Výsledek je přirozené číslo, takže (pro $N \geq K$) při dělení K!=K(K-1)...2 se nám všechna čísla ze jmenovatele vykrátí, takže rozklad výsledku na součin bude obsahovat pouze kladné mocniny. Rozklad výsledku budeme udržovat v poli, kde hodnota i-tého prvku je rovna exponentu E takovému, že i^E je přítomno v rozkladu výsledku. K tomu použijeme Eratosthenovo síto, ve kterém u složených čísel ještě označíme, jaké největší prvočíslo je dělí.

Např. síto pro N=19

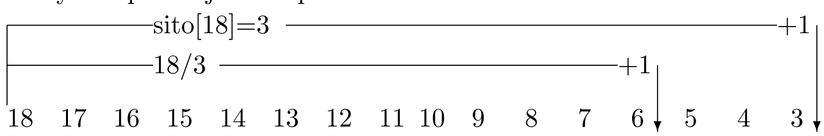
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	0	0	2	0	3	0	2	3	5	0	3	0	7	5	2	0	3	0

použijeme pro výpočet $\binom{19}{9} = \frac{19 \cdot 18 \cdot 17 \cdot 16 \cdot 15 \cdot 14 \cdot 13 \cdot 12 \cdot 11}{9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2} = 19 \cdot 17 \cdot 13 \cdot 11 \cdot 2.$

Nejprve vytvoříme pole rozklad:

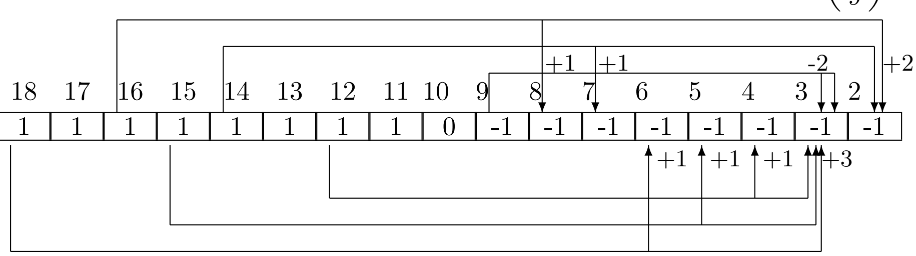
19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2
1	1	1	1	1	1	1	1	1	0	-1	-1	-1	-1	-1	-1	-1	-1

Potom sestupně prozkoumáme čísla od 19 do 2:
je-li zpracovávané číslo (i) prvočíslo (sito[i]==0), neupravujeme ho. Vezmeme další neprobraný základ i a odpovídající exponent E (i=18,E=1).
Je-li i číslo složené, pak v sítě najdeme největší prvočíslo P, které i dělí ($18 = 3 \cdot 6$, tj.`sito`[18] je 3; [i/`sito`[i]] je 6). Tím se nám rozloží i^E na P^E·(i/P) ^{E} (18 na $3^1 \cdot (18/3)^1$. Obě čísla jsou menší než i, přidáme je tedy s odpovídajícím exponentem do rozkladu a dále rozkládáme.



19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2
1	0	1	1	1	1	1	1	1	0	-1	-1	-1	0	-1	-1	0	-1

Postupně pro všechna složená čísla upravíme popsaným způsobem mocniny v rozkladu:
rozklad[sito[i]]=rozklad[sito[i]]+rozklad[i]; rozklad[i/sito[i]]=rozklad[i/sito[i]]+rozklad[i];
je li číslo prvočíslo, není čím ho krátit.
Až projdeme všechna čísla od N do 2, budeme mít v `rozkladu` na prvočíselných místech p_i mocniny těchto prvočísel, ve kterých se vyskytují ve výsledku, tj. budeme mít připravený prvočíselný rozklad $\binom{N}{K}$ jako $p_1^{l_1} \cdot p_2^{l_2} \cdot \dots \cdot p_k^{l_k}$. Na obrázku je znázorněno pole rozklad pro příklad $\binom{19}{9}$.



19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2
1	1	1	1	1	1	1	1	1	0	-1	-1	-1	-1	-1	-1	-1	-1

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2
1	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1

B. Násobení modulo M

V poli `rozklad` je: `rozklad`$[j] = 0$ pro $j \neq p_i$, `rozklad`$[p_i] = \ell_i$, $i = 1, \dots, k$, $j = 0, \dots, N$. Rozložíme ℓ_i na součet mocnin dvojky, kde se každá vyskytuje nejvýše jednou, $p^{\ell_i} = p^{2^a + 2^b + 2^c \dots 2^z} = p^{2^a} \cdot p^{2^b} \dots p^{2^z}$, kde $0 \leq a < b < c < d \dots < z \leq \log_2 \ell_i$ Násobit budeme tak, že začneme od $a = 0$, p^{2^0} mod M uložíme jako základ. Pak v každém kroku přistoupíme k další mocnině. Nejprve si ověříme, jestli tato mocnina zrovna v našem rozkladu figuruje, a pokud ano, číslo, kterým máme přinásobit výsledek, vypočítáme z předchozího: $p^{2^a} \pmod M = (p^{2^{a-1}} \pmod M) \cdot p^{2^{a-1}} \pmod M)$ mod M.

Realizace

Jsou realizovány 3 způsoby výpočtu:

1. pro složené M pomocí Čínské věty o zbytcích
2. pro prvočíselné $M < N$ podle Lucasovy věty
3. v případě, že M je mocnina prvočísla nebo $N < M$ používáme způsob [4].

V případě výpočtu podle Lucasovy věty potřebujeme vytvořit Eratosthenovo síto pro čísla $2 \ldots p-1$. Do pole `rozklad` budeme postupně pro jednotlivé n_i a k_i (zbytky po dělení n a k prvočíslem p) **přidávat** resp. **ubírat** jedničku, výsledně budeme mít v poli `rozklad` mocniny čísel $n_0, n_1, ...n_d$ a $k_0, k_1, ...k_d$, $0 < k_i < n_i < p-1$, ve kterých se vyskytují ve výsledném binomickém koeficientu.
Například pro $\binom{19}{9} \bmod 7 = \binom{5}{2}\binom{2}{1} \bmod 7$ se pole `rozklad` bude vytvářet:

6	5	4	3	2
0	0	0	0	0

$\longrightarrow$

6	5	4	3	2
0	1	1	0	-1

$\longrightarrow$

6	5	4	3	2
0	1	1	0	0

Tento rozklad upravíme popsaným způsobem na součin mocnin prvočísel a násobením určíme výsledek.

Program je vytvořen v **C**.
Ve funkci `main` se provádí načtení vstupu (N, K, M), kontrola smysluplnosti zadaných N, K, M, určení výsledku pro triviální případy ($K = 0,\ K = 1,\ N = K$). Dále funkce `main` zavolá funkci, která vytvoří síto pro větší z hodnot $N,\ M$ a podle zadaných N, M zavolá jednu z funkcí `crt, Lucas, n_nad_k`, ve kterých se provede výpočet. Podle jednotlivých mocnin prvočinitelů se z funkce `crt` zavolají funkce `Lucas` nebo `n_nad_k`. Výpočet inverzního prvku je realizován rozšířeným Eukliedovym algoritmem.

Časová a paměťová složitost

Realizace vyžaduje:
1. vytvoření Eratosthenova síta
2. vytvoření pole `rozklad`
3. zpracování rozkladu - průběžné krácení
4. násobení `modulo` M
Na rozklad na prvočísla potřebujeme dvě pole (`sito` a `rozklad`) o N resp. M prvcích. Ostatní operace vyžadují několik proměnných typu `integer`. Paměťová složitost je tedy $\mathcal{O}(N)$ resp. $\mathcal{O}(M)$.

Hledání prvočísel Eratosthenovým sítem má složitost $\mathcal{O}(N \log\log N)$, resp. $\mathcal{O}(M \log\log M)$.

Vytvoření pole `rozklad` pro $N < M$ vyžaduje jeden průchod polem velikosti N tj. $\mathcal{O}(N)$.
Pro $M < N$ v podstatě převádíme N do M-kové soustavy, což vyžaduje $\mathcal{O}(\log_M(N))$ dělení. Takže vytvoření pole `rozklad` byde v tomto případě mít složitost $\mathcal{O}(M \log_M(N))$.

Průběžné zkrácení vyžaduje jeden průchod polem `rozklad`, tj. $\mathcal{O}(N)$ resp. $\mathcal{O}(M)$.

Násobení. Složitost úpravy exponentů $\ell_1, \ldots, \ell_k$ pro prvočíselné M je $\mathcal{O}(N)$, výpočet Eulerovy funkce $\varphi(M)$ má složitost $\mathcal{O}(N \log M)$, přepočet použitím rozkladu na mocniny dvojky $\mathcal{O}(N \log N)$. Druhá část algoritmu bude asymptoticky alespoň $\mathcal{O}(N \log N)$.

Testování

Program jsem testoval pro $N, K, M \in \langle 2, 100000 \rangle$. Výsledky výpočtu funkce `crt` a `Lucas` jsem porovnával s výsledky funkce `n_nad_k`.

Literatura

[1] D. KNUTH, The art of Programming, vol. II

[2] A. GRANVILLE, Arithmetic properties of binomial coefficients. I. Binomial coefficients modulo prime powers, in Organic mathematics (Burnaby, BC, 1995), 253–276,CMS Conf. Proc., 20, Amer. Math. Soc., Providence, RI, 1997.

[3] K. S. DAVIS and W. A. WEBB, A binomial coefficient congruence modulo prime powers, J. Number Theory. 43 (1993), 20–23.

[4] Vzorové řešení KSP 21-4-3