Pro jednoduchost řádkového zápisu budeme označovat binomiální koeficient neboli kombinační číslo $\binom{n}{k}$ zápisem comb(n, k).
Opáčko: comb(14,4) =

A. 286
B. 364
C. 715
D. 816
E. 1001

Je dána množina malých písmen
P = {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} bez diakritiky.
P obsahuje celkem 26 písmen.

Počet jednoprvkových podmnožin P je

A. 1
B. 25
C. 26
D. 27
E. 26!

Je dána množina malých písmen
P = {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} bez diakritiky.
P obsahuje celkem 26 písmen. Počet dvouprvkových podmnožin P je

A. 26*(26+1) = 702
B. 26*26 = 676
C. 26*(26–1) = 650
D. 26*(26+1)/2 = 351
E. 26*(26–1)/2 = 325

Je dána množina malých písmen
P = {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} bez diakritiky.
P obsahuje celkem 26 písmen.

Počet osmiprvkových podmnožin P je

A. 1535524
B. 1542008
C. 1562275
D. 1562311
E. 1572448

Je dána množina malých písmen
P = {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} bez diakritiky.
P obsahuje celkem 26 písmen.
Máme vygenerovat tabulku T s 8 sloupci, jejíž každý řádek bude obsahovat jednu 8 prvkovou podmnožinu množiny P tak, že každá buňka na řádku bude obsahovat právě jeden znak. Tabulka bude obsahovat všechny možné navzájem různé 8 prvkové podmnožiny P a žádná podmnožina se v T nebude opakovat. Určete, jak bude tabulka velká. a zda ji váš počítač bude moci vyplnit během 1 sec.

A. Ano, s rezervou.
B. Ne, ani náhodou.
C. Skoro ano, je to tak tak.
D. Chybí další nutné údaje

Uvažujeme všechny podmnožiny množiny $\{1, 2, 3, 4, 5, 6, 7, 8\}$. Podmnožina $\{1, 2, 3, 6, 7, 8\}$ má rank

A. 123
B. 213
C. 231
D. 312
E. 321

Uvažujeme všechny podmnožiny množiny {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}. Podmnožina P má rank 777. P je

A. {1,2,7,10}
B. {1,4,9,10}
C. {2,5,6,7,9}
D. {3,6}
E. {7,8,9}

Uvažujeme všechny podmnožiny množiny {1, 2, 3, 4, ..., N}. Když sečteme ranky všech jednoprvkových podmnožin této množiny vyjde

A. $N - 1$
B. $N + 1$
C. $2^{\mathbf{N}} - 1$
D. $2^{\mathbf{N}}$
E. $2^{\mathbf{N}} + 1$

Permutace p množiny $\{1, 2, 3, ..., n\}$ se nazývá derangement, pokud platí
$1 \leq k \leq n \Rightarrow p(k) \neq k$. *(na k-té pozici není k, pro žádné k)*
Všechny derangement-y množiny $\{1, 2, 3\}$ jsou jen dva: (2 3 1), (3 1 2).
Napište všechny derangement-y množiny $\{1, 2, 3, 4\}$. Kolik jich je celkem?

A. 7
B. 9
C. 10
D. 12
E. 13

Permutace (2 1 3) množiny $\{1, 2, 3\}$ má rank

A. 1
B. 2
C. 3
D. 4
E. 5

Problem **11** of **25**

Permutace (2 4 3 1) množiny {1, 2, 3, 4 } má rank

A. 10
B. 11
C. 12
D. 13
E. 14

Permutace (3 1 2 4 5 6) množiny {1, 2, 3, 4, 5, 6} má rank
(hint: 6! = 720)

A. 120
B. 240
C. 359
D. 360
E. 719

Problem **13** of **25**

Permutace (2 4 6 8 10 9 7 5 3 1) množiny {1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } má rank

A. 452239
B. 461239
C. 462209
D. 462239
E. 462439

Rank permutace P množiny M = {1, 2, 3, 4, 5, 6, 7 } je právě 1000. Napište P

A. (2 4 3 5 7 1 6)
B. (2 4 3 6 7 1 5)
C. (2 4 3 6 7 5 1)
D. (2 7 3 6 4 1 5)
E. (3 4 2 6 7 1 5)

Problem **15** of **25**

Uvažujeme dvouprvkové podmnožiny množiny {1, 2, 3, 4, 5, 6, 7}. Podmnožina P = {4,7} má rank

A. 13
B. 15
C. 17
D. 19
E. 21

Uvažujeme pětiprvkové podmnožiny množiny $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$.
Podmnožina $P = \{1, 3, 5, 7, 9\}$ má rank

A. 49
B. 52
C. 64
D. 76
E. 82

Mějme permutace množiny M = $\{1, 2, 3, ..., n\}$, $n > 4$. Permutaci p této množiny prohlásíme za *výhodnou*, pokud platí: $p(3) \in \{3, n\}$, $p(n) \in \{3, n\}$, $p(1) = 1$, $p(2) = 2$, $p(i) \in \{4, ..., n-1\}$ pro $i = 4, ..., n-1$.
Určete počet *výhodných* permutací množiny M.

A. $2*n$
B. $n*(n+4)$
C. $2*3*n*(n+1)$
D. $2*(n-4)!$
E. $(n-3)!$

Uvažujeme permutace množiny M = $\{1, 2, 3, ..., n\}$. Cyklus délky k ($k \geq 2$) v permutaci p definujeme jako množinu
A = $\{a_1, a_2, ..., a_k\} \subseteq$ M, pro kterou platí:
$1 \leq a_1 < a_2 < ... < a_k \leq n$, $\quad$ $p(a_j) = a_{j+1}$ pro $1 \leq j < k$, $p(a_k) = a_1$.
Počet cyklů v permutaci (4 6 12 1 5 9 7 10 2 3 8 11) je

A. 0
B. 1
C. 2
D. 3
E. 4

Uvažujeme permutace množiny M = $\{1, 2, 3, ..., n\}$. Cyklus délky k ($k \geq 2$) v permutaci p definujeme jako množinu
$A = \{a_1, a_2, ..., a_k\} \subseteq M$, pro kterou platí:
$1 \leq a_1 < a_2 < ... < a_k \leq n$, $p(a_j) = a_{j+1}$ pro $1 \leq j < k$, $p(a_k) = a_1$.
Určete, kolik je takových permutací množiny $\{1, 2, 3, ..., n\}$, které obsahují právě dva cykly, z nichž jeden má délku 4 a druhý délku $n-4$.

A. $n + (n-4)$
B. $n! * 4!$
C. $(n-4)! * 4!$
D. $\binom{n}{4}$
E. $\binom{n}{4} * \binom{n-4}{4}$

Napište pseudokód funkce, která vypíše všechny neprázdné podmnožiny množiny $\{0, 1, 2, ..., n-1\}$.

Všechny permutace množiny M s 98 prvky očíslujeme pořadovými čísly od 0 do 98!–1. V programu pak nepracujeme s permutacemi ale jen s jejich pořadovými čísly. Víme, že budeme zkoumat pokaždé najednou 100 permutací, čili v paměti budeme muset mít uloženo právě 100 pořadových čísel různých permutací množiny M. Kolik minimálně bitů si musíme v paměti rezervovat, abychom si těchto 100 reprezentací mohli uložit?

Uvažujme všechny k-prvkové podmožiny množiny M = {1, 2, 3, ..., n}, $1 \leq k \leq n$. Vyjděte z algoritmu transformujícího seznam prvků jedné podmnožiny na seznam prvků podmožiny bezprostředně následující v lexikografickém uspořádání těchto podmnožin. Navrhněte a popište algoritmus, který bude transformovat seznam prvků jedné podmnožiny na seznam prvků podmožiny bezprostředně předcházející v témže lexokografickém uspořádání. Bude mít stejnou asymptotickou složitost?

Rank permutace π množiny N = {0, 1, 2, ..., $n-1$} je pořadové číslo této permutace v seznamu všech permutací množiny N uspořádaném v rostoucím lexikografickém pořadí, přičemž prvky seznamu jsou číslovány od 0. Napište pseudokód funkce která v čase úměrném n vytiskne takovou permutaci π množiny N, jejíž rank je právě $n!/2$. Předpokládáme $n \geq 2$.

Uvažujme všechny permutace množiny M = {1, 2, 3, ..., n}. Vyjděte z algoritmu transformujícího danou permutaci na permutaci bezprostředně následující v lexikografickém uspořádání. Navrhněte a popište algoritmus, který bude transformovat danou permutaci na permutaci bezprostředně předcházející v témže lexokografickém uspořádání. Bude mít stejnou asymptotickou složitost?

Permutace p množiny $\{1, 2, 3, ..., n\}$ se nazývá derangement, pokud platí
$1 \leq k \leq n \Rightarrow p(k) \neq k$. Určete 1000000-tý prvek v lexikografickém uspořádání derangement-ů množiny $\{1, 2, 3, ..., 20\}$.