Jméno a příjmení: ....

UČO: ....

1

Označme $f \preccurlyeq g \quad \Leftrightarrow \quad f \in O(g)$. Seřaďte podle rychlosti růstu funkce (proměnné n).

(A) $\frac{5^{n-2}}{4^{n+2}} \preccurlyeq \log_2(n^6) \preccurlyeq n \cdot \log_2(n!) + n \preccurlyeq n \cdot \sqrt{n}$

(B) $\frac{5^{n-2}}{4^{n+2}} \preccurlyeq \log_2(n^6) \preccurlyeq n \cdot \sqrt{n} \preccurlyeq n \cdot \log_2(n!) + n$

(C) $\log_2(n^6) \preccurlyeq \frac{5^{n-2}}{4^{n+2}} \preccurlyeq n \cdot \sqrt{n} \preccurlyeq n \cdot \log_2(n!) + n$

(D) $\log_2(n^6) \preccurlyeq n \cdot \log_2(n!) + n \preccurlyeq n \cdot \sqrt{n} \preccurlyeq \frac{5^{n-2}}{4^{n+2}}$

(E) $\log_2(n^6) \preccurlyeq n \cdot \sqrt{n} \preccurlyeq n \cdot \log_2(n!) + n \preccurlyeq \frac{5^{n-2}}{4^{n+2}}$

Odpověď: $\boxed{}$

2

Máme dvě číselné posloupnosti A, B, každá má délku n. Tyto posloupnosti nemusí být seřazené. Jaká je časová složitost *optimálního* algoritmu, který rozhodne, zda všechny prvky posloupnosti A jsou menší nebo nanejvýš rovny všem prvkům posloupnosti B? Algoritmus tedy musí zjistit, zda platí $\forall i, 1 \leq i \leq n \, \forall j, 1 \leq j \leq n. \, A_i \leq B_j$.
Pozor, hledáme ten *nejefektivnější* algoritmus!

(A) $\Theta(1)$ **(B)** $\Theta(n^2)$ **(C)** $\Theta(n)$ **(D)** $\Theta(n \cdot \log n)$ **(E)** $\Theta(n^{\frac{3}{2}})$

Odpověď: $\boxed{}$

3

Funkce `bintoint` převádí přirozená čísla z binárního zápisu, který je reprezentován neprázdným seznamem dvojkových číslic `0`, `1`, na číslo typu `Integer`. Dvojkové číslice jsou v seznamu v pořadí od nejnižšího po nejvyšší řád.
Například `bintoint [0]` $= 0$, `bintoint [0,0,1]` $= 4$, `bintoint [1,0,1,1]` $= 13$. Doplňte výraz v definici funkce.

```
bintoint [b]   =  b
bintoint (b:s)  =  .....................
```

4

Je dán algoritmus, který v n-prvkové (neseřazené) posloupnosti čísel najde největší číslo.

```
max := A_1;
i := 2;
while i ≤ n  /*invariant*/  do
{  if A_i > max then  max := A_i;
   i := i + 1
}
```

Vstupní podmínka: posloupnost A je neprázdná a obsahuje navzájem různá celá čísla,
formálně: $n \geq 1 \ \wedge \ \forall i, 1 \leq i \leq n \, \forall j, 1 \leq j \leq n. \, i \neq j \Rightarrow A_i \neq A_j$.
Výstupní podmínka: `max` obsahuje největší prvek posloupnosti A, formálně: $\forall i, 1 \leq i \leq n.$ `max` $\geq A_i$.
Určete invariant cyklu, který je splněn ve vyznačeném místě, a lze pomocí něho odvodit parciální korektnost algoritmu.