# SPRAWDZIAN I

**Imię i nazwisko:**

**Nr indeksu:**

**Nr grupy:**

**Uwaga!** Sprawdzian jest testem wielokrotnego wyboru, gdzie wszystkie możliwe kombinacje odpowiedzi są dopuszczalne (tj. zarówno wszystkie odpowiedzi poprawne, część odpowiedzi poprawna jak i brak odpowiedzi poprawnych). Poprawne odpowiedzi należy zaznaczyć, z lewej strony kartki, symbolem "+". Natomiast symbol "-" jak i brak symbolu przy odpowiedzi oznacza odpowiedź niepoprawną. Pytanie jest uznane za poprawnie rozwiązane (tj. +1pkt) wtedy i tylko wtedy gdy wszystkie jego odpowiedzi zaznaczone są poprawnie. Życzymy powodzenia ...

1. Niech $f(n) = (\lg n!)^2$, wtedy prawdą jest, że:
   (a) [+] $f(n) = O\left(n^3\right)$,
   (b) [+] $f(n) = \Omega\left(\lg (n!)^2\right)$,
   (c) [+] $f(n) = \Theta\left(2^{\lg n^2} \lg^2 n\right)$.

2. Rozważmy funkcję $f : \mathbb{N} \to \mathbb{R}^+ \cup \{0\}$ postaci $f(n) = \sqrt{n}$, wtedy:
   (a) [+] $f(n) = \Theta\left(\frac{1}{c} \cdot f(n) + c\right)$, gdzie $c$ jest pewną dodatnią stałą,
   (b) [+] $f(n) = O(n \cdot f(n))$,
   (c) [+] $f(n) = \Omega\left(f(n)^{-2}\right)$.

3. Które z poniższych zdań jest prawdziwe:
   (a) [–] jeżeli $f(n) = O(n)$ i $g(n) = O(n)$, to $f(n) + g(n) = \Omega\left(n^2\right)$,
   (b) [+] jeżeli $f(n) = O\left(n^2\right)$ i $g(n) = O\left(n^2\right)$, to $f(n) + g(n) = O\left(n^2\right)$,
   (c) [+] jeżeli $f(n) = \Omega(n)$ i $g(n) = \Omega(n)$, to $f(n) \cdot g(n) = \Omega\left(n^2\right)$.

4. Załóżmy, że złożoność czasową pewnego algorytmu $A$ określa funkcja $T(A, n) = n^2$, gdzie $n$ jest rozmiarem danych wejściowych. Komputer $K$ wykonuje rozważany algorytm dla danych rozmiaru 8 w ciągu 32 sekund, tj. $T_K(A, 8) = 32$. Stąd:
   (a) [–] $T_K(A, 9) = 40$,
   (b) [+] $T_K(A, 10) = 50$,
   (c) [+] w ciągu 200 sekund komputer $K$ wykona rozważany algorytm dla danych wejściowych rozmiaru co najmniej 20.

5. Rozważmy następujący algorytm

```
void Algorytm(int n) {
   Alg1(n);
   for (i=0;i<n*n;i++) {
      Alg2(n);
   }
}
```

   gdzie $Alg_1$ oraz $Alg_2$ są algorytmami o złożoności czasowej odpowiednio $T(Alg_1 n) = O(\sqrt{n} \lg n!)$ oraz $A(Alg_2, n) = \Theta\left(n^2\right)$, $W(Alg_2, n) = \Omega\left(n^2\right)$, stąd:
   (a) [–] $T(Algorytm, n) = \Theta\left(n^2 \lg n!\right)$,

(b) [–] $A(Algorytm, n) = O\left(n^2 \lg n!\right)$,

(c) [+] $W(Algorytm, n) = \Omega\left(n^2 \lg n!\right)$.

6. Rozważmy następujący algorytm

```
int Cos(int n) { // wp:  n∈ ℕ
   int i=1;
   while (i*i≥0) i=i+1;
   return n; // wk:  n∈ ℕ
}
```

wtedy:

(a) [–] program Cos jest całkowicie poprawny w strukturze liczb naturalnych,

(b) [+] program Cos jest częściowo poprawny w strukturze liczb naturalnych,

(c) [–] program Cos jest całkowicie poprawny w strukturze liczb naturalnych przy założeniu, że operator dodawania zdefiniujemy jak dzielenie, tj. $+ =_{def} /$.

7. Rozważmy następujący algorytm

```
int Cos(int n, int k) {
   int i=n, wynik=0;
   while (i≥k) {
      i=i div k;
      wynik=wynik+1;
   }
   return wynik; // wk:  wynik=⌊log_k n⌋
}
```

wtedy:

(a) [–] program Cos jest częściowo poprawny dla warunku początkowego $k, n \in \mathbb{N}$,

(b) [+] program Cos jest częściowo poprawny dla warunku początkowego dla $n \in \mathbb{N}$ i $k \in \mathbb{N} \setminus \{0, 1\}$,

(c) [+] program Cos jest całkowicie poprawny dla warunku początkowego $n = k^c$, dla $c \in \mathbb{N}^+$ i $k \in \mathbb{N} \setminus \{0, 1\}$,

8. Rozważmy następujący algorytm

```
int Cos(int n) { // wp:  n∈ ℕ
   int i=0, s=0;
   while (i≤n) {
      i=i+1;
      s=s+i;
   }
   return s;
}
```

wtedy:

(a) [+] niezmiennikiem pętli w programie Cos jest formuła $s = \frac{i(i+1)}{2}$,

(b) [–] niezmiennikiem pętli w programie Cos jest formuła $s = \frac{(i+1)(i+2)}{2}$,

(c) [–] niezmiennikiem pętli w programie Cos jest formuła $i \geq 0$, a warunkiem końcowym $s = \sum_{i=0}^{n-1} i$.

9. Które ze zdań jest prawdziwe:

(a) [+] sprawdzenie, czy dany element należy do nieuporządkowanego uniwersum rozmiaru $n$ wymaga $\Omega(\sqrt{n})$ porównań,

(b) [–] sprawdzenie, czy dany element należy do nieuporządkowanego uniwersum rozmiaru $\sqrt{n}$ wymaga $O(\lg n)$ porównań,

(c) [+] koszt czasowy optymalnego algorytmu wyszukania elementu minimalnego w nieuporządkowanym uniwersum rozmiaru $10^6$ wynosi $10^6 - 1$.

10. Rozważmy algorytm „turniej” dla danych rozmiaru $n = 2^k$, gdzie $k \in \mathbb{N}^+$. Które z poniższych stwierdzeń jest zawsze spełnione:
    (a) [–] koszt budowy drzewa turnieju wynosi dokładnie $n$ porównań,
    (b) [+] element 3-ci co do wielkości „pojedynkował się” z co najwyżej z $\lg n$ elementami,
    (c) [+] element 1-szy co do wielkości „pojedynkował się” z co najmniej $\lg n$ elementami.
11. Rozważmy iteracyjny algorytm dla problemu min-max i danych rozmiaru $n = 2^k$, gdzie $k \in \mathbb{N}^+$, wtedy:
    (a) [+] algorytm ten jest optymalnym rozwiązaniem dla rozważanego problemu,
    (b) [+] złożoność czasową algorytmu można oszacować przez $O(n)$,
    (c) [–] złożoność pamięciowa algorytmu jest rzędu $\Omega(\lg n)$.
12. Które ze zdań jest prawdziwe:
    (a) [–] sprawdzenie algorytmem BinSearch, czy dany element należy do nieuporządkowanego uniwersum rozmiaru $n$ wymaga $O(n)$ porównań,
    (b) [+] sprawdzenie algorytmem BinSearch, czy dany element należy do uporządkowanego uniwersum rozmiaru $n$ wymaga $O(n)$ porównań,
    (c) [–] koszt czasowy algorytmu BinSearch dla poprawnych danych rozmiaru $2 \cdot 10^3$ wynosi co najwyżej 8 porównań.
13. Załóżmy, że pewien algorytm $Alg$ dla danych wejściowych rozmiaru $n$ składa się z trzech części:
    - $n\sqrt{n}$-krotne wyszukanie elementu maksymalnego metodą sekwencyjną,
    - $n$-krotne wyszukanie elementu minimalnego algorytmem Hoare'a,
    - $\sqrt{n}$-krotne wyszukanie, algorytmem optymalnym, mediany w ciągu uporządkowanym.

    Które z oszacowań jest poprawne:
    (a) [+] $A(Alg, n) = \Omega(n^2)$,
    (b) [–] $W(Alg, n) = O(n^2)$,
    (c) [–] $S(Alg, n) = \Theta(n)$.
14. Który z poniższych ciągów jest poprawnym rezultatem wykonania procedury Partition dla danych wejściowych
    $$4, 7, 9, 11, 3, 5, 2,$$
    (a) [–] 4,7,3,2,11,9,5,
    (b) [+] 2,7,9,11,3,5,4,
    (c) [–] 2,3,4,5,7,9,11.
15. Rozważmy wyszukiwanie elementu $n$-tego co do wielkości, w $n$-elementowej uporządkowanej rosnąco tablicy wejściowej, przy zastosowaniu algorytmu Hoare'a z procedurą podziału zgodną z metodą Split, wtedy:
    (a) [+] złożoność czasową rozwiązania w tym przypadku szacujemy przez $O(n)$,
    (b) [+] złożoność czasową rozwiązania w tym przypadku szacujemy przez $\Omega(n)$,
    (c) [–] złożoność czasową rozwiązania w tym przypadku szacujemy przez $\Theta(n^2)$.
16. Prowadzący zajęcia ćwiczeniowe z ASD jest:
    (a) blondynem,
    (b) brunetem,
    (c) albo nie znam człowieka ... albo myślę, że liczba włosów na jego głowie daje się szacować przez $\Omega(1)$.