

Lista 1

Uwaga. Zadania 1-4 są „tablicowe”, zadanie 5 jest „implementacyjne”. W zadaniach 1-3 przyjmujemy, że zliczamy tylko operacje arytmetyczne (występujące *explicite* w kodzie).

Zadanie 1 (0,25 punktu) Podaj złożoność następującego algorytmu w notacji $\Theta(\cdot)$.

```
for i in range(n):  
    k = 2 + 2
```

Zadanie 2 (0,25 punktu) Podaj złożoność następującego algorytmu w notacji $\Theta(\cdot)$.

```
for i in range(n):  
    for j in range(n):  
        for k in range(j): # range(j)!  
            a = 2 + 2
```

Zadanie 3 (0,75 punktu) Podaj złożoność następującego algorytmu w notacji $\Theta(\cdot)$.

```
while n > 0:  
    n //= 2
```

Zadanie 4 (1,5 punktu) Wyznacz pesymistyczny czas działania (w notacji $\Theta(\cdot)$) algorytmu Gaussa (sprowadzania macierzy rzeczywistej $n \times n$ do postaci schodkowej). Jako operacje elementarne przyjmij operacje arytmetyczne (na indywidualnych współczynnikach macierzy) i operacje porównywania liczb.

Zadanie 5 (1,25 punktu) Zaimplementuj dowolny algorytm sprawdzający, czy podana para napisów równej długości zawiera wspólny znak. Zdefiniuj rozmiar danych tego algorytmu, oraz szczegółowo uzasadnij (jako komentarz w kodzie) pesymistyczny czas działania algorytmu w zależności od tego rozmiaru. Możesz przyjąć, że następujące operacje są elementarne (wymagają stałego czasu):

- Operacje arytmetyczne na liczbach.
- Operacje logiczne.
- Przypisanie.
- Odczytanie elementu napisu, listy lub słownika pod zadaniem indeksem lub kluczem.
- Stworzenie pustej listy, zbioru i słownika.
- Sprawdzenie, czy element jest elementem zbioru lub kluczem w słowniku.
- Sprawdzenie długości listy, napisu, etc.
- Doklejenie elementu na koniec listy.
- Dodanie lub usunięcie elementu zbioru.
- Dodanie, usunięcie lub podmiana pary klucz-wartość w słowniku.

Nie musisz uwzględniać kosztu czasowego kontroli sterowania (np. tworzenia obiektów `range` w pętli `for`). Nie zakładaj niczego o innych operacjach. Nie musisz używać wszystkich podanych operacji.