

## **Pracownia 1 - instrukcja**

Niech A, B, C oraz D oznaczają liczbę oczek wyrzuconych na kościach.

Oblicz:

- $M = \max(B, 7-B)$
- $N = 18 + C + D$

W dalszej części używane będą oznaczenia:

- (1) - metoda Gaussa
- (2) - metoda Banachiewicza
- (3) - metoda Cholesky'ego
- (4) - metoda iteracji prostej
- (5) - metoda Gaussa-Jordana
- (6) - metoda iteracji Seidela

### **Zadanie 1. (4 punkty)**

Zbadaj złożoność obliczeniową metody A (np. jeśli na pierwszej kości wypadły 2 oczka – metoda Banachiewicza).

W tym celu:

- wykonaj kilka testów próbnych dla danej metody, by ustalić maksymalny rozmiar macierzy  $n$  (by obliczenia nie trwały zbyt długo – dla macierzy o największym rozmiarze czas powinien być poniżej 1 sekundy) – możesz wykorzystać w tym celu sekcję 4 pliku *pracownia1.py* **(1 punkt)**
- uzupełnij metodę *mierz\_czas* w pliku *zadanie.py* tak, by wykonywała ona po  $M$  iteracji algorytmu, który masz zbadać – jej działanie ma być analogiczne do metody *mierz\_czas* klasy *Sortowania* (możesz od razu tak ją skonstruować, by wewnątrz pojawiła się instrukcja warunkowa, która pozwoli Ci wykorzystywać obie metody, które masz porównać w zadaniu 2) **(1 punkt)**
- spróbuj wywołać metodę *badaj\_zlozonosc* dla obiektu klasy *Zadanie* (możesz wykorzystać w tym celu sekcję 5 pliku *pracownia1.py*); następnie narysuj odpowiedni wykres i opisz go **(1 punkt)**
- podsumuj wyniki i wyciągnij wnioski z przeprowadzonego doświadczenia **(1 punkt)**

### **Zadanie 2. (2 punkty)**

Porównaj efektywność uzyskiwania rozwiązania metodami A i B.

(jeśli  $A = B$ , porównaj metody A i  $(7-B)$ ).

W tym celu:

- zmodyfikuj metodę *mierz\_czas* tak, by wykonywała ona po  $M$  iteracji obu algorytmów, które masz zbadać
- narysuj odpowiedni wykres wywołując metodę *porownaj\_metody* i skomentuj go (czerwone punkty to pierwsza z metod, niebieskie – druga) – możesz wykorzystać w tym celu sekcję 5 pliku *pracownia1.py* **(1 punkt)**
- podsumuj wyniki i wyciągnij wnioski z przeprowadzonego doświadczenia **(1 punkt)**

W raporcie zamieść także kod metody *mierz\_czas* oraz zapisz argumenty z jakimi powołałeś do istnienia obiekt klasy *Zadanie* w obu zadaniach.

**Raport należy podpisać nazwiskiem i imieniem!** np. KleksAmbrozy-Raport1.pdf i wysłać do 24 godzin od momentu zakończenia pracowni na adres prowadzącego.

### Wskazówki:

1. pisząc raport pamiętaj, że w tym zadaniu pracujesz na macierzach, nie na listach,
2. losując układ równań wywołaj odpowiednią metodę losowania (albo *losuj*, albo *losuj\_symetryczny\_dodatnio\_okreslony*),
3. pamiętaj, by zmierzyć czasy wszystkich metod istotnych w procesie rozwiązywania układu, np. w przypadku iteracji prostej trzeba wywołać 2 metody – najpierw metodę *przygotuj*, potem metodę *iteruj\_roznica*, dobierając odpowiednie parametry (parametr *eps* możesz przyjąć  $1e-10$ , a jako normę – 0 lub 1). Dla każdego z algorytmów jedna z metod zwraca wartość 0, kiedy algorytm nie może być zastosowany, możesz to wykorzystać w eksperymencie.