

1. Napisz funkcję rekurencyjną, która sprawdzi czy w tablicy znajduje się element mniejszy niż liczba podana przez użytkownika (**20 pkt**):

- Zdefiniuj funkcję rekurencyjną, która przyjmuje 3 argumenty: tablicę danych, rozmiar tablicy oraz liczbę do sprawdzenia
- Funkcja zwraca element typu *bool* (zdefiniowany w bibliotece *stdbool.h*)
- Nie używaj pętli
- Funkcja zwraca 1, jeśli tablica zawiera element mniejszy niż liczba, natomiast 0, jeśli warunek nie jest spełniony

W funkcji main:

- Zadeklaruj tablicę o rozmiarze 20. Użyj polecenia preprocesora *#define*, w celu stworzenia zmiennej *SIZE* o rozmiarze 20
- Wypełnij tablicę liczbami pseudolosowymi z przedziału 10 do 30
- Wywołaj funkcję rekurencyjną i wypisz wynik do konsoli w postaci:
 - a. Jeśli tablica zawiera element mniejszy niż liczba, to wypisz: Tablica zawiera element mniejszy niż <liczba>
 - b. Jeśli tablica nie zawiera elementu mniejszego niż liczba, to wypisz: Tablica nie zawiera elementu mniejszego niż <liczba>

2. Napisz funkcję rekurencyjną, która obliczy całkowitą potęgę liczby rzeczywistej (**20 pkt**):

- Zdefiniuj funkcję rekurencyjną, która przyjmuje 2 argumenty: podstawę potęgi oraz wykładnik potęgi
- Funkcja zwraca element typu *double*
- Nie używaj pętli

W funkcji main:

- Zadeklaruj zmienną typu *double* (podstawę potęgi) oraz zmienną typu *int* (wykładnik potęgi)
- Wywołaj funkcję rekurencyjną i wypisz wynik do konsoli w postaci: <wykładnik> potęga liczby <podstawa> wynosi: <wynik>

3. Napisz funkcję rekurencyjną, która zsumuje elementy tablicy (**20 pkt**):

- Zdefiniuj funkcję rekurencyjną, która przyjmuje 2 argumenty: tablicę danych, rozmiar tablicy
- Funkcja zwraca element typu *double*
- Nie używaj pętli

W funkcji main:

- Zadeklaruj tablicę o rozmiarze 20. Użyj polecenia preprocesora *#define*, w celu stworzenia zmiennej *SIZE* o rozmiarze 20
- Wypełnij tablicę liczbami pseudolosowymi z przedziału 0 do 1 (niech to będą liczby rzeczywiste)
- Wywołaj funkcję rekurencyjną i wypisz wynik do konsoli w postaci: Suma elementów tablicy wynosi: <wynik>

4. Napisz funkcję, która zsumuje elementy tablicy (**20 pkt**):

- Funkcja przyjmuje 3 argumenty: wskaźnik do pierwszego elementu tablicy, rozmiar tablicy, liczbę całkowitą
- Jeśli ostatni argument funkcji jest równy 1, to sumowanie odbywa się w pętli *while*, natomiast sumowanie odbywa się w pętli *for* dla ostatniego argumentu równego 2
- Użyj notacji wskaźnikowej

W funkcji main:

- Zadeklaruj tablicę o rozmiarze 20. Użyj polecenia preprocesora *#define*, w celu stworzenia zmiennej *SIZE* o rozmiarze 20
- Wypełnij tablicę liczbami pseudolosowymi z przedziału 0 do 1 (niech to będą liczby rzeczywiste)
- Wywołaj funkcję i wypisz wynik do konsoli w postaci:
Suma elementów tablicy wynosi: <wynik>

5. Wykonuj polecenia zgodnie z instrukcją (**20 pkt**):

- Zadeklaruj zmienną *float x* i dwa wskaźniki do zmiennych *float *pa* i *pb*,
- Przypisz wartość zmiennej *x*
- Przypisz adres zmiennej *x* do wskaźnika *pa*
- Przypisz *NULL* do wskaźnika *pb*
- **Wypisz adresy wszystkich zmiennych**
- **Wypisz wartości wszystkich zmiennych**
- Skopiuj wartość wskaźnika *pa* do wskaźnika *pb*
- **Wypisz wartości wszystkich zmiennych**
- Użyj wskaźnika *pb* do zmodyfikowania wartości zmiennej *x*
- **Wypisz wartości wszystkich zmiennych**
- **Wypisz wartości przechowywane przez wskaźniki**

Przykładowe rozwiązanie:

```
&x=0x7ffd29741068, &pa=0x7ffd29741060, &pb=0x7ffd29741058
```

```
x=10.000000, pa=0x7ffd29741068, pb=(nil)
```

```
x=10.000000, pa=0x7ffd29741068, pb=0x7ffd29741068
```

```
x=20.000000, pa=0x7ffd29741068, pb=0x7ffd29741068
```

```
*pa=20.000000, *pb=20.000000
```