



Podstawy programowania

Ćwiczenie 11

Funkcje – część 2

Zadanie

Zadanie 1

Zdefiniować funkcję, która na podstawie promienia koła oblicza pole powierzchni oraz obwód koła. Obliczone wartości powinny być zwracane za pomocą zmiennych wskaźnikowych. Nagłówek funkcji:

```
void ObliczKolo1(float promien, float *pole, float *obwod);
```

Zdefiniować drugi wariant funkcji (obliczone wartości zwracane w postaci struktury **wynik**, za pomocą instrukcji **return**):

```
struct wynik
{
    float pole;
    float obwod;
};
```

Nagłówek funkcji: `struct wynik ObliczKolo2(float promien);`

Zdefiniować trzeci wariant funkcji (obliczone wartości zwracane w postaci struktury **wynik**, poprzez wskaźnik). Nagłówek funkcji:

```
void ObliczKolo3(float promien, struct wynik *w);
```

Zadanie

Zadanie 2

Zdefiniować funkcję ze zmienną liczbą argumentów:

```
void srednia(double *sa, double *sg, ...);
```

która dla liczb typu `double` podawanych jako argumenty wywołania, oblicza średnią arytmetyczną i średnią geometryczną.

Obliczone średnie zwracane są poprzez zmienne wskaźnikowe.

Przyjąć, że wartość zero oznacza koniec listy argumentów.

Wzory na średnie:

$$SA = \frac{\sum_{i=0}^{N-1} a_i}{N} = \frac{a_0 + a_1 + \dots + a_{N-1}}{N}$$

$$SG = \sqrt[N]{\prod_{i=0}^{N-1} a_i} = \sqrt[N]{a_0 \cdot a_1 \cdot \dots \cdot a_{N-1}}$$

do obliczenia
 $\sqrt[N]{}$
użyć funkcji `pow`

Zadanie

Zadanie 3

Zdefiniować funkcję ze zmienną liczbą argumentów:

```
int szukaj(double szukana, ...);
```

która wśród liczb typu `double` podawanych jako argumenty wywołania poszukuje wartości `szukana` i w zależności od tego czy została ona znaleziona lub nie, zwraca prawdę (wartość 1) lub fałsz (wartość 0).

Przyjąć, że liczba zero oznacza koniec listy argumentów.

Zwrócić uwagę na problem porównywania liczb zmiennoprzecinkowych za pomocą operatora porównania `==` (patrz wykład nr 4 „*Operatory*”).

W celu rozwiązania tego problemu, zdefiniować funkcję:

```
int porownaj(double x, double y, double eps)
```

która porównuje dwie wartości `x` i `y` z dokładnością `eps`.

Jeżeli są one równe, to funkcja zwraca prawdę (wartość 1), a jeżeli nie są równe, to zwraca fałsz (wartość 0).