Programowanie obiektowe C++

Tablice jako parametry funkcji

Tablice jako parametry funkcji

1. Przykład

Oceny punktowe ze sprawdzianu reprezentuje *n*-elementowy ciąg licz dodatnich. Poniżej zapisano program, w którym wyznaczana jest liczba studentów posiadających ocenę powyżej średniej, zawierający funkcje:

- **`drukuj()`** – wyświetlanie oceny studentów,
- **`czytaj()`** – wprowadzanie liczby studentów i ich ocen,
- **`srednia()`** – wyznaczanie średniej ocen (wynik przekazywany prze parametr),
- **`liczba()`** – określanie liczby ocen większych niż parametr (wynik przekazywany przez nazwę).

```cpp
#include <iostream>

using namespace std;

void czytaj(const char napis[], int *n, float tab[])
// void czytaj(const char *napis, int *n, float *tab)
{
  cout << napis << endl;
  cin >> *n;

  for (int i = 0; i < *n; i++)
  {
    cout << "Wartosc " << i + 1 << ": ";
    cin >> tab[i]; // cin >> *(tab + i);
  }
}

void drukuj(const char napis[], int n, float tab[])
{
  cout << napis << endl;

  for (int i = 0; i < n; i++)
    cout << tab[i] << endl;
}

void srednia(int n, float tab[], float *wynik)
{
  float suma = 0;
  for (int i = 0; i < n; i++)
    suma += tab[i];
  *wynik = suma / n;
}

int liczba(int n, float tab[], float w)
{
```

```
  int l = 0;
  for (int i = 0; i < n; i++)
    if (tab[i] > w) l++;
  return l;
}

int main()
{
  float oceny[100], sr;
  int n;

  czytaj("Podaj liczbe ocen i oceny.", &n, oceny);
  drukuj("Oceny punktowe studentow.", n, oceny);

  srednia(n, oceny, &sr);
  cout << "Srednia: " << sr << "." << endl;

  cout << "Powyzej sredniej: " << liczba(n, oceny, sr) << " osob." << endl;

  return EXIT_SUCCESS;
}
```

- Przepisz i uruchom program.
- Przetestuj działanie programu po zmianie odwołań do elementów tablic na wskaźnikowe (np.: **`x[i]`** na **`*(x + i)`**).

2. Zadanie

Uzupełni program z przykładu 1 o:

- funkcję obliczającą procent studentów z oceną w przedziale $\langle 0{,}3\,w\,;0{,}7\,w\rangle$, gdzie w to parametr funkcji. Wywołaj funkcję z parametrem równym wartości oceny średniej,
- funkcję obliczającą minimalną i maksymalną ocenę studenta,
- funkcję wyznaczające tablice zawierającą stopnie studentów (skala punktów od 0 pkt. do 40 pkt.): poniżej 20 pkt. – 2.0, 20 pkt. – 3.0, 24 pkt. – 3.5, 28 pkt. – 4.0 32 pkt. – 4.5, 36 pkt. i więcej – 5.0.

3. Zadanie

Dwa n-elementowe ciągi liczb $\{x_i\}$ i $\{y_i\}$ reprezentują współrzędne punktów na płaszczyźnie. Opracuj program zawierający funkcję:

- do wczytywania liczby punktów i ich współrzędnych,
- do wyświetlania na ekranie współrzędnych punktów z wybranej ćwiartki układu współrzędnych (nr ćwiartki będzie parametrem funkcji),
- wyznaczającą odległości poszczególnych punktów od początku układu współrzędnych,
- wyznaczającą punkt najdalej oddalony od początku układu współrzędnych,
- wyznaczającą dodatkowe tablice ze współrzędnymi punktów położonych w okręgu o środku w początku układu współrzędnych i promieniu *r*.

Zadania dodatkowe

1. Uzupełnij program z zadania 1 o funkcję wyznaczającą sześcioelementową tablicę zawierającą liczby studentów, którzy uzyskali poszczególne stopnie (pierwszy element tablicy – liczba stopni 2.0, drugi element tablicy – liczba stopni 3.0 itd.).

2. Dany jest wielomian $w(x)=a_n x^n+\ldots+a_2 x^2+a_1 x^1+a_0 x^0=a_n x^n+\ldots+a_2 x^2+a_1 x+a_0$. Napisz program (używając funkcji) do:

   - wyznaczania wartości wielomianu według schematu **Hornera**, w znanej postaci $w(x)=(\ldots((a_n x+a_{n-1})x+\ldots+a_2)x+a_1)x+a_0$, korzystając z podejścia rekurencyjnego i iteracyjnego (samodzielnie),

```
int hornerRek(int wsp[], int st, int x)
{
  if(st == 0)
    return wsp[0];

  return x * hornerRek(wsp, st - 1, x) + wsp[st];
}
```

```
int hornerIter(int wsp[], int st, int x)
{
  ...
}
```

   - wyznaczania pochodnej wielomianu $\frac{d}{dx}w(x)$,
   - wyznaczenie wartości pochodnej wielomianu.