

Lista na zajęcia 3

Za 0 punktów: znajdź co trzeba wpisać by fragment programu niżej się zapętlił:

```
while(true) {
    int k;
    cout << "podaj k" <, endl;
    cin >> k;
    if (k == 1) {
        cout << "wybrane 1" << endl;
    } else if (k == 99) {
        break;
    }
}
```

Podobnie (też za 0 punktów) dla fragmentu programu niżej:

```
while(true) {
    char c;
    cout << "podaj dowolny znak" << endl;
    cin >> c;
    if (c == 'q') {
        break;
    }
    cout << "Było " << c << endl;
}
```

1. Napisz funkcję która dla argumentów n i x obliczy sumę

$$\sum_{i=0}^n \frac{x^i}{i!}$$

Funkcja ma zwracać wartość sumy. Obliczenia proszę wykonać na typie `double`. Funkcja powinna zawierać pojedynczą pętlę i nie używać funkcji potęgowania (potrzebne wielkości proszę zapamiętać w zmiennych i uaktualniać z kroku na krok). Funkcja ma się nazywać `expsum_n` i działać z programem głównym niżej:

```
#include <iostream>
using namespace std;

int
main() {
    cout << expsum_n(0.5, 25) << endl;
    cout << expsum_n(1, 25) << endl;
}
```

tzn. pod dodaniu treści funkcji przed programem głównym program powinien się przekompiłować i poprawnie działać.

2. Liczby Fibonacciego F_n są zdefiniowane wzorami $F_0 = 0$, $F_1 = 1$, $F_{n+1} = F_n + F_{n-1}$. Napisz funkcję która obliczy wartość F_n . Funkcja ma używać pojedynczą pętlę i uaktualniać potrzebne zmienne z kroku na krok. Proszę nie używać rekursji. n ma być argumentem funkcji, funkcja ma zwracać obliczoną wartość.

Pobranie n i wypisanie wyniku proszę zrobić w programie głównym (tzn. funkcji `main`).

3. Uzupełnij program niżej tak by obliczał numer (indeks) maksymalnego elementu tablicy n -elementowej `a`. Zakładamy że n wynosi co najmniej 1. Właściwe obliczenia proszę wykonać wewnątrz funkcji, funkcja ma działać również dla innych danych. Program główny proszę modyfikować tylko by wprowadzić inne dane testowe.

```
int
maks_ind(int n, int a[]) {
    /* Tu trzeba wpisać treść */
}

#include <iostream>
using namespace std;

int
main() {
    int a[] = {1, 2, 12, -5};
    int i = maks_ind(4, a);
    cout << "max = " << a[i] << " na pozycji " << i << endl;
}

```

4. Wypełnij funkcję niżej treścią tak by po zakończeniu wywołania `sortuj` tablica `a` zawierała te same elementy jak na początku wywołania, ale posortowane.

```
void
sortuj(int n, int a[]) {
    /* Tu trzeba wpisać treść */
}

```

Sortowanie proszę wykonać przy pomocy podwójnej pętli, w ten sposób by najpierw znaleźć najmniejszy element tablicy (pętla wewnętrzna), następnie zamienić go z elementem na pozycji 0. Potem znaleźć najmniejszy element pomiędzy elementami na pozycjach 1 i $n - 1$ i zamienić go z elementem na pozycji 0. Powtarzać to w pętli aż tablica będzie posortowana (jest to metoda sortowania przez wybór).