

LABORATORIUM 6. INSTRUKCJE ITERACYJNE WHILE, DO...WHILE. INSTRUKCJA CONTINUE.

Cel laboratorium:

Zaznajomienie z realizacją algorytmów iteracyjnych. Nabycie praktycznych umiejętności programowania algorytmów iteracyjnych z wykorzystaniem instrukcji WHILE, DO...WHILE.

Zakres tematyczny zajęć:

- pojęcie i organizacja iteracji,
- instrukcje pętli z warunkiem wejścia WHILE, oraz pętli z warunkiem wyjścia DO...WHILE,
- zastosowanie instrukcji BREAK i CONTINUE w iteracjach.

Kompendium wiedzy:

Instrukcje iteracyjne (pętli): **while** i **do while** służą do powtarzania fragmentu programu, dopóki wartość wyrażenia wpisanego w instrukcję pętli jest różna od zera, czyli wyrażenie jest prawdziwe. W pętli **do...while** sprawdzenie wyrażenia następuje po pierwszym wykonaniu instrukcji, a więc taka pętla wykona się co najmniej raz.

```
while (wyrażenie) instrukcja;  
//dopóki wyrażenie jest prawdziwe wykonuj instrukcję  
  
while (wyrażenie)          //użycie instrukcji złożonej w while  
{instrukcja1;  
  instrukcja2;  
  .....  
}  
do {instrukcja} while (wyrażenie);  
//wykonuj instrukcję dopóki wyrażenie jest prawdziwe
```

Instrukcja **continue** powoduje wykonanie kolejnej iteracji (jeśli wyrażenie sterujące powtórzeniem jest prawdziwe) w bloku **while** (lub **do...while**) z pominięciem instrukcji następujących po **continue**.

```
np. int i=2; while (i<10) i=i*i;           //p1  
  
int i=2; do {i=i*i;} while (i<10);        //p2  
int i=0; while (i<6)                       //p3  
{ i++;  
  if (i==3) continue;  
  printf("%d\n",i);  
}
```



Pytania kontrolne:

1. Objasnij pojęcie iteracji.
2. Podaj składnię i zastosowanie instrukcji WHILE.
3. Podaj składnię i zastosowanie instrukcji DO ...WHILE.
4. Podaj zastosowanie instrukcji BREAK i CONTINUE w iteracjach.

Zadania do analizy

Zadanie 6.1. Instrukcje iteracyjne

- Przeanalizuj przykład programu wykorzystującego instrukcje iteracyjne.

Funkcja **suma** oblicza sumę podanych liczb, jeżeli są >100 .

Funkcja **tablicaWartosci** oblicza i wyświetla wartości wyrażenia $x + \sin x$ dla $x \in \langle a, b \rangle$, z krokiem k .

Funkcja **iloczyn** oblicza iloczyn liczb z przedziału $\langle a, b \rangle$.

Funkcja **zagadka** realizuje najwyżej pięciokrotne zgadywanie przez użytkownika wylosowanej liczby całkowitej jednocyfrowej. Zwraca liczbę prób do uzyskania sukcesu lub -1, jeżeli użytkownik nie zgadł.

- Podaj tekst w komentarzach.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  float suma(); //???
5  void tablicaWartosci(float a, float b, float k); //???
6  int iloczyn(int a, int b); //???
7  int zagadka (int a); //???
8
9  int main ()
10 {int nr=1;
11     while (nr) //???
12     {printf ("Wpisz numer funkcji (1, 2, 3 lub 4)\n,
13         Koniec programu - wpisz 0  ");
14     scanf( "%d", &nr);
15     if (nr==1) printf( "suma liczb = %f", suma()); //???
16     else
17     if(nr==2)
18     {    float od,do,krok;
19         printf("Podaj konce przedzialu i krok\n");
20         scanf("%f%f%f", &od, &do, &krok);
21         tablica_wartosci(od, do, krok); //???
22     }
23     else
24     if(nr==3)
25     {    int od,do;
```



```
25         printf("Podaj konce przedzialu \n");
26         scanf("%d%d", &od, &do);
27         printf(" iloczyn liczb podzielnych przez 3
28         =%d\n", iloczyn(od, do)); //???
29     }
30     else
31     {
32         if(nr==4)
33         {
34             int a,b;
35             a=rand()%10;
36             b=zagadka(a); //???
37             if (b== -1) printf(" nie zgadles\n");
38             else printf("zgadles za %d razem\n",b);
39         }
40     }
41     printf("\n KONIEC\n");
42     return 0;
43 }
44 //???
45 float suma()
46 {
47     int i, n;
48     float s, x;
49     s=0;
50     printf("Podaj ilosc liczb:");
51     scanf("%d", &n);
52     i=1;
53     while(i<=n) //???
54     {
55         printf("Podaj liczbe:");
56         scanf("%f", &x);
57         if(x > 100) s+=x;
58         i++;
59     }
60     return s;
61 }
62 void tablicaWartosci(float a, float b, float k)
63 {
64     float x;
65     x=a;
66     while(x<=b) //???
67     {
68         printf("x=%f      x+sin(x)=%f\n", x, x+sin(x));
69         x+=k;
70     }
71 }
72 int iloczyn(int a, int b)
73 {
74     int w, il;
75     if (a > b)
76     {
77         w=a;
78         a=b;
79         b=w;
80     }
81 }
```



```
73  if ((a+1) % 3 ==0) a++;
74  else if ( (a+2) % 3 ==0) a+=2;
75  w=a;
76  il=1;
77  while( w <= b) //???
78  {      il*=w;
79          w+=3;
80  }
81  return il;
82 }

83  int zagadka (int a)
84  {      int w, il;
85  il=0;
86  do //???
87  {      printf(" Podaj liczbę ");
88          scanf("%d", &w);
89          il++;
90  } while (a!= w && il < 6);
91  if (a != w) il=-1;
92  return il;
93 }
```

Zadania do wykonania

Zadanie 6.2. Średnia liczb

Napisz funkcję `sredniaWhile` obliczającą średnią arytmetyczną **n** liczb z wykorzystaniem instrukcji **while** i funkcję `sredniaDoWhile` obliczającą średnią arytmetyczną **n** liczb z wykorzystaniem instrukcji **do...while**.

Zadanie 6.3. Średnia liczb parzystych i nieparzystych

Napisz i wywołaj funkcję, która pobiera od użytkownika liczby całkowite do momentu wpisania cyfry 0 i wyświetla średnią liczb parzystych i nieparzystych.

Zadanie 6.4. Zliczanie wystąpień litery i liczby cyfr

Napisz i wywołaj funkcję, która dla wprowadzonego przez użytkownika zdania zakończonego kropką obliczy, ile razy wystąpiła w nim podana litera (np. a) oraz ile było cyfr.



Zadanie 6.5. Znajdowanie największej liczby

Napisz i wywołaj funkcję, która pobiera od użytkownika liczbę, ostatnia jest równa -1 oraz zwraca największą z tych liczb.

Zadanie 6.6. Menu programu

Program główny skonstruuj na zasadzie menu, z którego użytkownik wielokrotnie wybiera pozycję do wykonania lub kończy program.

Zadania dodatkowe

Zadanie 6.7. Ciąg rosnący

Napisz i wywołaj funkcję, która pobiera od użytkownika n liczb i sprawdza czy tworzą ciąg rosnący. Wynik funkcji: 1 jeśli tak, 0 jeśli nie.

Zadanie 6.8. Obliczenie stanu konta bankowego

Napisz i wywołaj funkcję, która dla podanych w parametrach: wpłaty oraz oprocentowania oblicza i wyświetla, po jakim czasie kwota wpłaty będzie co najmniej podwojona i ile dokładnie wynosi.

Zadanie 6.9. Sprawdzenie wyników egzaminu

Napisz i wywołaj funkcję, która dla n studentów sprawdzi, czy wszyscy studenci zdali egzamin.

Zadanie 6.10. Hotel bez pokoju nr.13

Hotel posiada n pokoi (n>13), połowa na parterze, połowa na I piętrze (w przypadku gdy n jest nieparzyste na I piętrze jest o 1 pokój więcej). Napisz funkcję, która wyświetli listę pokoi hotelowych od nr 1 do n (bez nr 13). Jeżeli pokój ma nr nieparzysty to jest on jednoosobowy, jeżeli parzysty to dwuosobowy. Lista powinna zawierać: nr pokoju, jedno/dwu osobowy, parter/I piętro. Powinna być wyświetlona równo w kolumnach. Wykorzystaj instrukcję CONTINUE.

Zadanie 6.11.

Napisz i wywołaj funkcję, która wyświetla kody ASCII liter alfabetu w wierszach w układzie:

A -> 65 a -> 97
B -> 66 b -> 98

