

PWSZ Elbląg
Instytut Informatyki Stosowanej im. Krzysztofa
Brzeskiego

Podstawy programowania I – laboratorium

Studium Stacjonarne, sem. 1, 2021/2022

Sprawozdanie nr 8,
środa grupa V, 10:45 – 12:15

Data wykonania ćwiczenia: 29.01.2022

Data oddania sprawozdania: 30.01.2022

Nazwisko i imię: Gajewski Mikołaj

Nr albumu: 20180

Nazwa pliku : lab8_SD_Gajewski_Mikołaj_20180.pdf

1. Zainicjować 5-elementowe tablice tab1, tab2, tab3 typu int .

a) tablicę tab1 liczbami dziesiętnymi

b) tablicę tab2 liczbami w zapisie oktalnym

c) tablicę tab3 liczbami w zapisie heksadecymalnym

Rozwiązanie : (a, b i c)

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* run this program using the console pauser or add your own getch, system("pause") or  
input loop */
```

```
int main()
```

```
{
```

```
    int tab1[5]={0,-1,2,3,5};
```

```
    int tab2[5]={015,012,022,031,052};
```

```
    int tab3[5]={0x21,0xFF,0xAB,0x31,0x22};
```

```
    int i ;
```

```
    printf("\n Drukowanie liczb w tablicy nr 1.");
```

```
    for (i =0 ; i <5 ; i++)
```

```
    {
```

```
        printf("\n tab1[%d]=%d",i,tab1[i]);
```

```
    }
```

```
    printf("\n Drukowanie liczb w tablicy nr 2.");
```

```

    for ( i =0 ; i<5;i++)
    {
        printf("\n tab2[%d]=%o",i,tab2[i]);
    }

    printf("\n Drukowanie liczb w tablicy nr 3.");
    for (i =0; i < 5; i++)
    {
        printf("\n tab3[%d]=0x%x",i, tab3[i]);
    }

    return 0;
}

```

Widok z terminala:

```

Drukowanie liczb w tablicy nr 1.
tab1[0]=0
tab1[1]=-1
tab1[2]=2
tab1[3]=3
tab1[4]=5
Drukowanie liczb w tablicy nr 2.
tab2[0]=15
tab2[1]=12
tab2[2]=22
tab2[3]=31
tab2[4]=52
Drukowanie liczb w tablicy nr 3.
tab3[0]=0x21
tab3[1]=0xff
tab3[2]=0xab
tab3[3]=0x31
tab3[4]=0x22
-----
Process exited after 0.8956 seconds with return value 0
Press any key to continue . . .

```

2. Wczytać z klawiatury 6 - elementową tablicę liczb typu int z zabezpieczeniem formatu i wydrukować.

Rozwiązanie :

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* run this program using the console pauser or add your own getch, system("pause") or  
input loop */
```

```
int main(int argc, char *argv[]) {
```

```
    int tab[6];
```

```
    int i,x ;
```

```
    printf("\nWczytanie wartosci do tablicy ");
```

```
    for (i=0; i< 6; i++ )
```

```
    {
```

```
        printf("\nPodaj wartosc calkowitoliczbowa :");
```

```
        while(scanf("%d",&x)!=1 )
```

```
        {
```

```
            printf("\nBlad formatu! Prosze podac jeszcze raz :");
```

```
            fflush(stdin);
```

```
        }
```

```
        printf("\nWczytano : %d",x);
```

```
        tab[i]=x;
```

```
    }
```

```
    for ( i = 0 ; i < 6 ; i++)
```

```
    {
```

```
        printf("\n tab[%d]= %d",i,tab[i]);
```

```
    return 0;
}
```

Widok z terminala :

```
Wczytanie wartosci do tablicy
Podaj wartosc calkowitoliczbowa :5

Wczytano : 5
Podaj wartosc calkowitoliczbowa :5

Wczytano : 5
Podaj wartosc calkowitoliczbowa :4

Wczytano : 4
Podaj wartosc calkowitoliczbowa :3

Wczytano : 3
Podaj wartosc calkowitoliczbowa :2

Wczytano : 2
Podaj wartosc calkowitoliczbowa :4

Wczytano : 4
tab[0]= 5
tab[1]= 5
tab[2]= 4
tab[3]= 3
tab[4]= 2
tab[5]= 4
-----
Process exited after 7.318 seconds with return value 0
Press any key to continue . . .
```

3. Uzupełnić program opracowany w pkt.2 o obliczanie sumy elementów tablicy, sumy elementów parzystych oraz sumy elementów nieparzystych.

Rozwiązanie :

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* run this program using the console pauser or add your own getch, system("pause") or
input loop */
```

```
int main(int argc, char *argv[]) {
```

```
int tab[6];

int i,x ;

int sumaE=0;

int sumaP=0;

int sumaN=0;


printf("\nWczytanie wartosci do tablicy ");

for (i=0; i< 6; i++ )
{
    printf("\nPodaj wartosc calkowitoliczbowa :");

    while(scanf("%d",&x)!=1 )
    {
        printf("\nBlad formatu! Prosze podac jeszcze raz :");
        fflush(stdin);
    }

    printf("\nWczytano : %d",x);

    tab[i]=x;
}

for ( i = 0 ; i < 6 ; i++)
{
    printf("\n tab[%d]= %d",i,tab[i]);
}

for ( i = 0 ; i < 6 ; i++)
{
    sumaE = sumaE + tab[i];
}
```

```

        if ( tab[i]%2 == 0 )
        {
            sumaP= sumaP + tab[i];
        }

        if ( tab[i]%2!=0 )
        {
            sumaN= sumaN +tab[i];
        }
    }

    printf("\nsomaE : %d", sumaE);
    printf("\nsomaP : %d", sumaP);
    printf("\nsomaN : %d", sumaN);

    return 0;
}

```

Widok z terminala :

```

Podaj wartosc calkowitoliczbowa :5
Wczytano : 5
Podaj wartosc calkowitoliczbowa :4
Wczytano : 4
Podaj wartosc calkowitoliczbowa :2
Wczytano : 2
Podaj wartosc calkowitoliczbowa :5
Wczytano : 5
Podaj wartosc calkowitoliczbowa :6
Wczytano : 6
Podaj wartosc calkowitoliczbowa :3
Wczytano : 3
tab[0]= 5
tab[1]= 4
tab[2]= 2
tab[3]= 5
tab[4]= 6
tab[5]= 3
sumaE : 25
sumaP : 12
sumaN : 13
-----
Process exited after 3.951 seconds with return value 0
Press any key to continue . . .

```

4. Napisać program wczytujący z klawiatury 8 - elementową tablicę liczb typu **double**, następnie obliczający

a) sumy elementów dodatnich i ujemnych,,

b) sumę elementów o wartościach należących do przedziału $[a, b]$, granice przedziału $[a, b]$, należy wczytać z klawiatury z zabezpieczeniem formatu i przy spełnieniu warunku $a < b$,

c) sumę elementów, których indeksy należą do zadanego przedziału indeksów $[ind1, ind2]$, indeksy należy wczytać z klawiatury z zabezpieczeniem formatu i spełniające warunki $ind1 \geq 0$ $ind2 \geq 0$ oraz $ind1 \leq 7$ i $ind2 \leq 7$.

Rozwiązanie :

a)

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* run this program using the console pauser or add your own getch, system("pause") or input loop */
```

```
int main(int argc, char *argv[]) {
```

```
    double tab[8];
```

```
    int i;
```

```
    double sumaD = 0, sumaU=0;
```

```
    printf("\n Wczytywanie wartosci do tablicy : ");
```

```
    for (i =0 ; i < 8; i++)
```

```
    {
```

```
        printf("\nPodaj wartosc : ");
```

```
        scanf("%lf",&tab[i]);
```

```
        if ( tab[i]>0) sumaD=sumaD + tab[i];
```

```
        else sumaU = sumaU +tab[i];
```



```

    }

    printf("\n SumaD : %f",sumaD);

    printf("\n SumaU : %f",sumaU);

    return 0;

}

```

Widok z terminala :

```

Wczytywanie wartosci do tablicy :
Podaj wartosc : 5
Podaj wartosc : 4
Podaj wartosc : 2
Podaj wartosc : 5
Podaj wartosc : -64
Podaj wartosc : -345
Podaj wartosc : 345
Podaj wartosc : 3

SumaD : 364.000000
SumaU : -409.000000
-----
Process exited after 7.193 seconds with return value 0
Press any key to continue . . .

```

Rozwiązanie :

b)

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

/* run this program using the console pauser or add your own getch, system("pause") or input loop */

```
int main(int argc, char *argv[]) {

    double tab[8];

    int i;

    int a,b;

    double sumaE=0;

    printf("\n Wczytywanie wartosci do tablicy : ");

    for (i =0 ; i < 8; i++)

    {

        printf("\nPodaj wartosc : ");

        scanf("%lf",&tab[i]);

    }

    for (i =0; i <8; i++)

    {

        printf("\ntab[%d]=%f",i,tab[i]);

    }

    printf("\nPodaj wartosc dla b dla przedzialu (a->b) :");

    while(scanf("%d",&b)!=1 )

    {

        printf("\nBlad formatu! Podaj jeszcze raz :");

        fflush(stdin);

    }

}
```

```

printf("\nWczytano : %d", b);

printf("\nPodaj wartosc dla a dla przedzialu (a->b) :");

while(scanf("%d",&a)!=1 || a>=b )
{
    printf("\nBlad formatu! Podaj jeszcze raz :");
    fflush(stdin);
}

printf("\nWczytano : %d", a);

for(i =0; i < 8; i++)
{
    if (tab[i]>a && tab[i]<b)
    {
        sumaE= sumaE+tab[i];
    }
}

printf("\nSumaE : %f",sumaE);

return 0;
}

```

Widok z terminala :

```

Wczytywanie wartosci do tablicy :
Podaj wartosc : 5

Podaj wartosc : 4

Podaj wartosc : 3

Podaj wartosc : 5

Podaj wartosc : 6

Podaj wartosc : 3

Podaj wartosc : 4

Podaj wartosc : 5

tab[0]=5.000000
tab[1]=4.000000
tab[2]=3.000000
tab[3]=5.000000
tab[4]=6.000000
tab[5]=3.000000
tab[6]=4.000000
tab[7]=5.000000
Podaj wartosc dla b dla przedzialu (a->b) :6

Wczytano : 6
Podaj wartosc dla a dla przedzialu (a->b) :4

Wczytano : 4
SumaE : 15.000000
-----
Process exited after 8.144 seconds with return value 0
Press any key to continue . . .

```

c)

Rozwiązanie :

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* run this program using the console pauser or add your own getch, system("pause") or
input loop */
```

```
int main(int argc, char *argv[]) {
```

```

double tab[8];

int i;

int ind1,ind2;

double sumaE=0;


printf("\n Wczytywanie wartosci do tablicy : ");

for (i =0 ; i < 8; i++)

{

    printf("\nPodaj wartosc : ");

    scanf("%lf",&tab[i]);

}

for (i =0; i <8; i++)

{

    printf("\ntab[%d]=%f",i,tab[i]);

}

printf("\nPodaj wartosc dla ind2 dla przedzialu (ind1->ind2) :");

while(scanf("%d",&ind2)!=1 || ind2>=7 || ind2<=0)

{

    printf("\nBlad formatu! Podaj jeszcze raz :");

    fflush(stdin);

}

printf("\nWczytano : %d", ind2);

printf("\nPodaj wartosc dla ind1 dla przedzialu (ind1->ind2) :");

while(scanf("%d",&ind1)!=1 || ind1>=ind2 || ind1<=0 )

{

    printf("\nBlad formatu! Podaj jeszcze raz :");

```

```
        fflush(stdin);  
    }  
    printf("\nWczytano : %d", ind1);  
  
    for(i =0; i <8 ; i++)  
    {  
        if(i>=ind1 && i<=ind2)  
        {  
            sumaE=sumaE+tab[i];  
        }  
    }  
    printf("\nSumaE : %f",sumaE);  
    return 0;  
}
```

Widok z terminala :

```

Wczytywanie wartosci do tablicy :
Podaj wartosc : 5

Podaj wartosc : 4

Podaj wartosc : 5

Podaj wartosc : 6

Podaj wartosc : 5

Podaj wartosc : 4

Podaj wartosc : 3

Podaj wartosc : 4

tab[0]=5.000000
tab[1]=4.000000
tab[2]=5.000000
tab[3]=6.000000
tab[4]=5.000000
tab[5]=4.000000
tab[6]=3.000000
tab[7]=4.000000
Podaj wartosc dla ind2 dla przedzialu (ind1->ind2) :6

Wczytano : 6
Podaj wartosc dla ind1 dla przedzialu (ind1->ind2) :5

Wczytano : 5
SumaE : 7.000000
-----
Process exited after 5.231 seconds with return value 0
Press any key to continue . . .

```

6. Napisać program wczytujący tablicę x liczb typu int o parzystej liczbie elementów i przepisujący z niej co drugi element do tablicy y. Wydrukować tablice x i y.

Rozwiązanie :

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* run this program using the console pauser or add your own getch, system("pause") or
input loop */
```

```
int main(int argc, char *argv[]) {
```

```

int x[8];

int y[8];

int i;


printf("\n Wczytywanie wartosci do tablicy : ");

for (i =0 ; i < 8; i++)

{

    printf("\nPodaj wartosc : ");

    scanf("%d",&x[i]);

    printf("\ntab[%d]=%d",i,x[i]);

    if(i%2==0)

    {

        y[i]=x[i];

    }

}

printf("\n");

for(i = 0; i <8; i++)

{

    printf("\ntab[%d]=%d",i, y[i]);

}

return 0;

}

```

Widok z terminala :


```

Wczytywanie wartosci do tablicy :
Podaj wartosc : 5

tab[0]=5
Podaj wartosc : 6

tab[1]=6
Podaj wartosc : 5

tab[2]=5
Podaj wartosc : 4

tab[3]=4
Podaj wartosc : 5

tab[4]=5
Podaj wartosc : 6

tab[5]=6
Podaj wartosc : 5

tab[6]=5
Podaj wartosc : 4

tab[7]=4

tab[0]=5
tab[1]=0
tab[2]=5
tab[3]=0
tab[4]=5
tab[5]=-1
tab[6]=5
tab[7]=0
-----
Process exited after 4.859 seconds with return value 0
Press any key to continue . . .

```

7. Wczytać 5 - elementową tablicę liczb typu double, zainicjalizować dwa indeksy tablicy, wydrukować tablicę, następnie zamienić w niej elementy o zadanych indeksach i ponownie wydrukować.

Rozwiązanie :

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* run this program using the console pauser or add your own getch, system("pause") or
input loop */
```

```
int main(int argc, char *argv[]) {

    double x[5];

    int ind1=4, ind2=3;

    int i;

    printf("\n Wczytywanie wartosci do tablicy : ");

    for (i =0; i < 5; i++)
    {

        printf("\nPodaj wartosc : ");

        scanf("%lf",&x[i]);

        printf("\ntab[%d]=%f",i,x[i]);

    }

    printf("\n");

    for(i = 0; i < 5; i++)

    {

        printf("\ntab[%d]=%lf",i, x[i]);

    }

    printf("\n");

    x[ind1]=x[ind2];

    x[ind2]=x[ind1];

    for(i =0; i< 5; i++)

    {

        printf("\ntab[%d]=%lf",i,x[i]);

    }

}
```

```
    return 0;
}
```

Widok z terminala :

```
Wczytywanie wartosci do tablicy :
Podaj wartosc : 5
tab[0]=5.000000
Podaj wartosc : 67
tab[1]=67.000000
Podaj wartosc : 65
tab[2]=65.000000
Podaj wartosc : 43
tab[3]=43.000000
Podaj wartosc : 5
tab[4]=5.000000

tab[0]=5.000000
tab[1]=67.000000
tab[2]=65.000000
tab[3]=43.000000
tab[4]=5.000000

tab[0]=5.000000
tab[1]=67.000000
tab[2]=65.000000
tab[3]=43.000000
tab[4]=43.000000
-----
Process exited after 3.604 seconds with return value 0
Press any key to continue . . .
```

8. Napisać program znajdujący we wczytanej tablicy liczb typu int wartości minimalną i maksymalną.

Rozwiązanie :

```
#define N 8

#include <stdio.h>

#include <conio.h>

int main(int argc, char argv[]){

    int i,i1,j,k,p,amin;
```

```

int a[N]= {9,5,-2,7,-4,15,-20,100};

for ( i=0; i < N ; i++){

    printf("\ntab[%d]=%d",i, a[i]);

}

printf("\n");

for (i=0;i<N;i++){

    amin=a[i];

    k=i;

    for (j=i+1;j<N;j++)

        if (a[j]<amin){

            amin=a[j];

            k=j; }

    if (k!=i){

        p=a[i];

        a[i]=a[k];

        a[k]=p;

    }

}

printf("\n Stan tablicy po sortowaniu ");

for (i=0;i<N;i++)

    printf(" % d ",a[i]);

printf("\nWartosc maksymalna wynosi : %d",a[7]);

printf("\nWartosc minimalna wynosi : %d",a[0]);

printf("\n");

system("pause");

}

```

Widok z terminala :

```
tab[0]=9
tab[1]=5
tab[2]=-2
tab[3]=7
tab[4]=-4
tab[5]=15
tab[6]=-20
tab[7]=100

Stan tablicy po sortowaniu -20 -4 -2 5 7 9 15 100
Wartosc maksymalna wynosi : 100
Wartosc minimalna wynosi : -20
Press any key to continue . . .
```

9. Napisać program odwracający we wczytanej tablicy kolejność elementów.

Rozwiązanie :

```
#define N 8

#include <stdio.h>

#include <conio.h>

int main(int argc, char argv[]){

    int i,j;

    int a[N]= {9,5,-2,7,-4,15,-20,100};

    int b[8];

    for ( i =0; i < N ; i ++ )

        printf("\ntab[%d]=%d",i, a[i]);

    for ( i = 0; i < N; i++)

        b[i]=a[N-1-i];

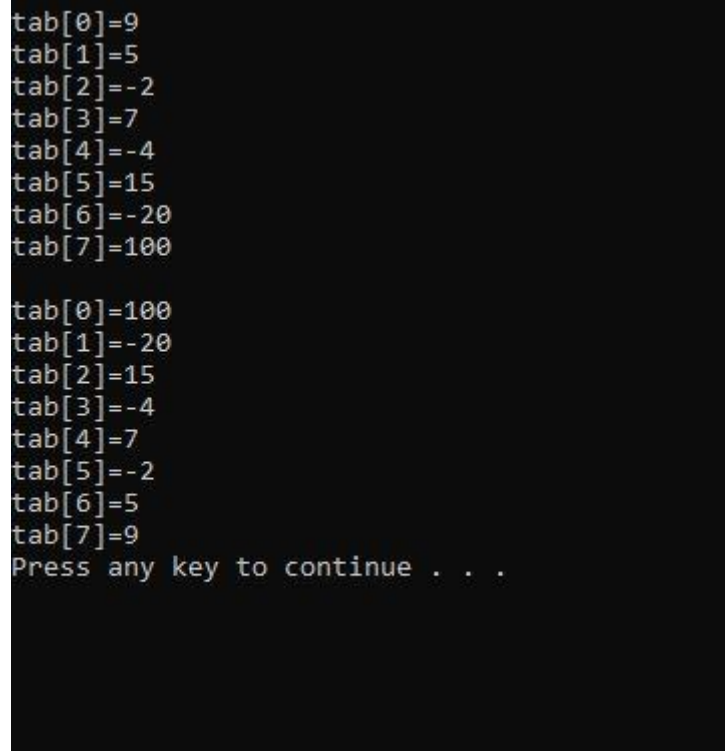
    printf("\n");

    for ( i =0; i < N ; i ++ )

        printf("\ntab[%d]=%d",i, b[i]);
```

```
    printf("\n");  
    system("pause");  
}
```

Widok z terminala :



```
tab[0]=9  
tab[1]=5  
tab[2]=-2  
tab[3]=7  
tab[4]=-4  
tab[5]=15  
tab[6]=-20  
tab[7]=100  
  
tab[0]=100  
tab[1]=-20  
tab[2]=15  
tab[3]=-4  
tab[4]=7  
tab[5]=-2  
tab[6]=5  
tab[7]=9  
Press any key to continue . . .
```

10. Napisać program wymieniający zawartości dwóch zainicjowanych tablic liczb typu int.

Rozwiązanie :

```
#define N 8  
  
#include <stdio.h>  
  
#include <conio.h>  
  
int main(int argc, char argv[]){  
    int i,j;  
  
    int a[N]= {9,5,-2,7,-4,15,-20,100};  
  
    int b[N]={1,2,3,4,5,6,7,8};  
  
    int c[N];  
  
    for(i = 0; i <N ; i ++){ //kopiowanie tablicy b
```

```
        c[i]=a[i];
    }

    printf("\npierwsza tablica :");

    for ( i =0; i < N ; i ++ )

        printf("\ntab[%d]=%d",i, a[i]);

    printf("\n");

    printf("\ndruga tablica :");

    for ( i =0; i < N ; i ++ )

        printf("\ntab[%d]=%d",i, b[i]);


    printf("\n");

    printf("\nZawartosc pierwszej z druga : ");

    for(i = 0; i <N ; i ++){

        a[i]=b[i];

    }

    for ( i =0; i < N ; i ++ )

        printf("\ntab[%d]=%d",i, a[i]);

    printf("\n");

    printf("\nZawartosc drugiej z pierwsza : ");

    for(i = 0; i <N ; i ++){

        b[i]=c[i];

    }

    for ( i =0; i < N ; i ++ )

        printf("\ntab[%d]=%d",i, b[i]);

    printf("\n");

    system("pause");
```

```
}
```

Widok z terminala :

```
pierwsza tablica :
tab[0]=9
tab[1]=5
tab[2]=-2
tab[3]=7
tab[4]=-4
tab[5]=15
tab[6]=-20
tab[7]=100

druga tablica :
tab[0]=1
tab[1]=2
tab[2]=3
tab[3]=4
tab[4]=5
tab[5]=6
tab[6]=7
tab[7]=8

Zawartosc pierwszej z druga :
tab[0]=1
tab[1]=2
tab[2]=3
tab[3]=4
tab[4]=5
tab[5]=6
tab[6]=7
tab[7]=8

Zawartosc drugiej z pierwsza :
tab[0]=9
tab[1]=5
tab[2]=-2
tab[3]=7
tab[4]=-4
tab[5]=15
tab[6]=-20
tab[7]=100
Press any key to continue . . .
```