

Państwowa Wyższa Szkoła Informatyki i Przedsiębiorczości	
Sprawozdanie nr: 5	Temat: Tablice
Wydział: Informatyki i Nauk o Żywności	Autor: Damian Jakub Kaczyński Nr albumu: 10492
Przedmiot: Podstawy Programowania	Prowadzący: mgr inż. Paweł Kamiński
Data: 21.11.2021	Ocena:
Uwagi:	

## Wstęp

Zagadnienia poruszane na zajęciach są coraz bardziej złożone przez co rosną możliwości pisanych przez nas programów a wraz z nimi poziom trudności ich tworzenia.

Na ostatnich zajęciach poznaliśmy tablice i możliwości ich użycia z wykorzystaniem pętli poznanych na poprzednich zajęciach. Tablice służą do przechowywania wielu wartości tego samego typu, a ich ponumerowanie umożliwia proste ich zapisywanie, odczytywanie i edycję.

## Treść zadania 1

Napisz program wczytujący z klawiatury 10 liczb całkowitych. Wczytane liczby należy wydrukować na ekranie.

## Rozwiązanie

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

```
int main() {
    int tablica[10];

    for(int i = 0; i < 10; i++){
```

```

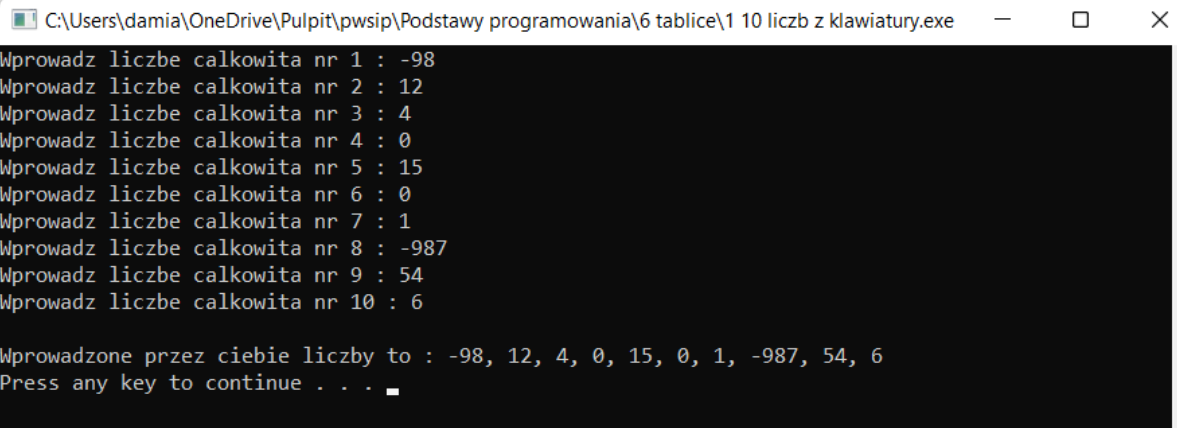
        printf("Wprowadz liczbe calkowita nr %d : ", i + 1);
        scanf("%d", &tablica[i]);
    }
    printf("\nWprowadzone przez ciebie liczby to : ");

    for(int i = 0; i < 10; i++){
        printf("%d", tablica[i]);
        if(i < 9){
            printf(", ");
        }
    }
    printf("\n");

    system("pause");
    return(0);
}

```

## Przykładowe uruchomienie



```

C:\Users\damia\OneDrive\Pulpit\pwsip\Podstawy programowania\6 tablice\1 10 liczb z klawiatury.exe
Wprowadz liczbe calkowita nr 1 : -98
Wprowadz liczbe calkowita nr 2 : 12
Wprowadz liczbe calkowita nr 3 : 4
Wprowadz liczbe calkowita nr 4 : 0
Wprowadz liczbe calkowita nr 5 : 15
Wprowadz liczbe calkowita nr 6 : 0
Wprowadz liczbe calkowita nr 7 : 1
Wprowadz liczbe calkowita nr 8 : -987
Wprowadz liczbe calkowita nr 9 : 54
Wprowadz liczbe calkowita nr 10 : 6

Wprowadzone przez ciebie liczby to : -98, 12, 4, 0, 15, 0, 1, -987, 54, 6
Press any key to continue . . . _

```

## Treść zadania 2

Napisz program wczytujący z klawiatury 10 liczb całkowitych. Wczytane liczby należy wydrukować na ekranie w odwrotnej kolejności.

## Rozwiązanie

```

#include <stdio.h>
#include <stdlib.h>

int main() {
    int tablica[10];

    for(int i = 0; i < 10; i++){

```

```

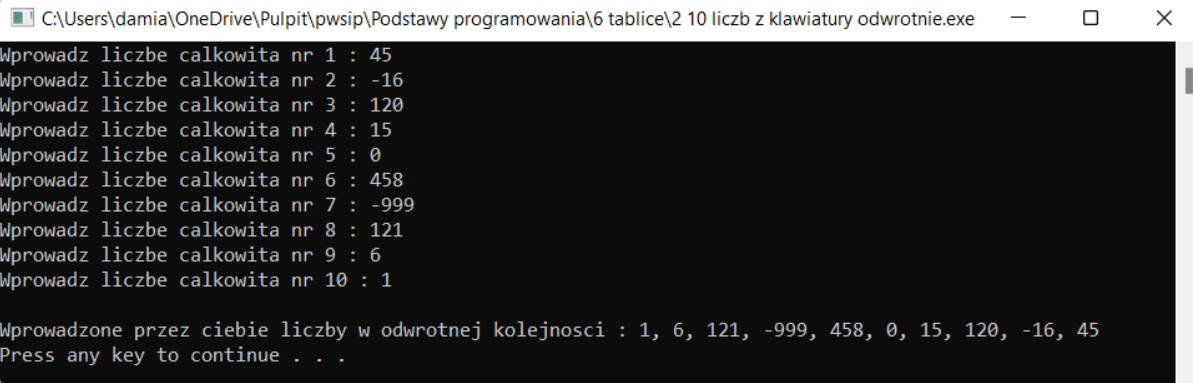
        printf("Wprowadz liczbe calkowita nr %d : ", i + 1);
        scanf("%d", &tablica[i]);
    }
    printf("\nWprowadzone przez ciebie liczby w odwrotnej kolejnosci : ");

    for(int i = 9; i >= 0; i--){
        printf("%d", tablica[i]);
        if(i > 0){
            printf(", ");
        }
    }
    printf("\n");

    system("pause");
    return(0);
}

```

## Przykładowe uruchomienie



```

C:\Users\damia\OneDrive\Pulpit\pwsip\Podstawy programowania\6 tablice\2 10 liczb z klawiatury odwrotnie.exe
Wprowadz liczbe calkowita nr 1 : 45
Wprowadz liczbe calkowita nr 2 : -16
Wprowadz liczbe calkowita nr 3 : 120
Wprowadz liczbe calkowita nr 4 : 15
Wprowadz liczbe calkowita nr 5 : 0
Wprowadz liczbe calkowita nr 6 : 458
Wprowadz liczbe calkowita nr 7 : -999
Wprowadz liczbe calkowita nr 8 : 121
Wprowadz liczbe calkowita nr 9 : 6
Wprowadz liczbe calkowita nr 10 : 1

Wprowadzone przez ciebie liczby w odwrotnej kolejnosci : 1, 6, 121, -999, 458, 0, 15, 120, -16, 45
Press any key to continue . . .

```

## Treść zadania 3

Napisz program wczytujący z klawiatury n liczb całkowitych. Program ma znaleźć największą spośród podanych liczb i wydrukować ją na ekranie.

## Rozwiązanie

```

#include <stdio.h>
#include <stdlib.h>

```

```

int main() {

```

```

    int wielkoscTab, max;

```

```

    printf("Wybierz ilosc liczb ktore chcesz wprowadzic : ");
    scanf("%d", &wielkoscTab);
    int tablica[wielkoscTab];

```

```

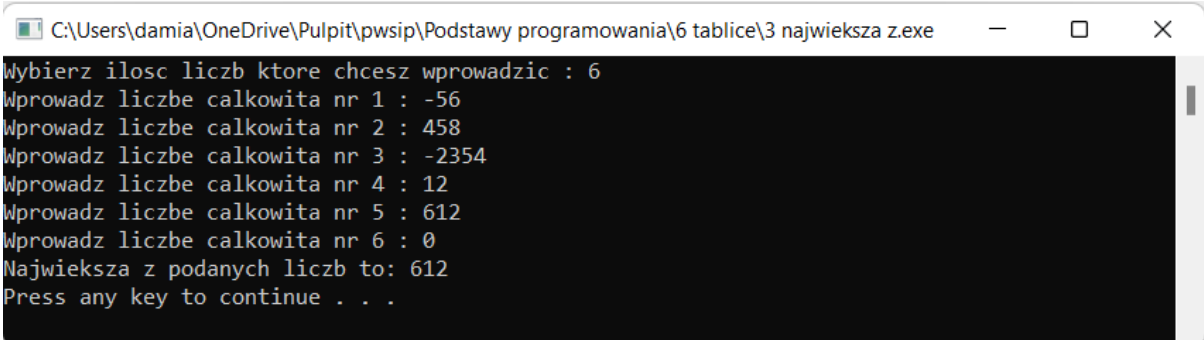
    for(int i = 0; i < wielkoscTab; i++){
        printf("Wprowadz liczbe calkowita nr %d : ", i + 1);
        scanf("%d", &tablica[i]);
    }

    max = tablica[0];
    for(int i = 1; i < wielkoscTab; i++){
        if(tablica[i] > max){
            max = tablica[i];
        }
    }
    printf("Najwieksza z podanych liczb to: %d\n", max);

    system("pause");
    return(0);
}

```

## Przykładowe uruchomienie



```

C:\Users\damia\OneDrive\Pulpit\pwsip\Podstawy programowania\6 tablice\3 najwieksza z.exe
Wybierz ilosc liczb ktore chcesz wprowadzic : 6
Wprowadz liczbe calkowita nr 1 : -56
Wprowadz liczbe calkowita nr 2 : 458
Wprowadz liczbe calkowita nr 3 : -2354
Wprowadz liczbe calkowita nr 4 : 12
Wprowadz liczbe calkowita nr 5 : 612
Wprowadz liczbe calkowita nr 6 : 0
Najwieksza z podanych liczb to: 612
Press any key to continue . . .

```

## Treść zadania 4

Napisz program wczytujący ciąg liczb rzeczywistych. Każdą podaną liczbę należy podnieść do kwadratu, a następnie wydrukować na ekranie.

## Rozwiązanie

```

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int main() {

    int wielkoscTab;
    printf("Wybierz ilosc liczb ktore chcesz wprowadzic : ");
    scanf("%d", &wielkoscTab);
    float tablica[wielkoscTab];

```

```

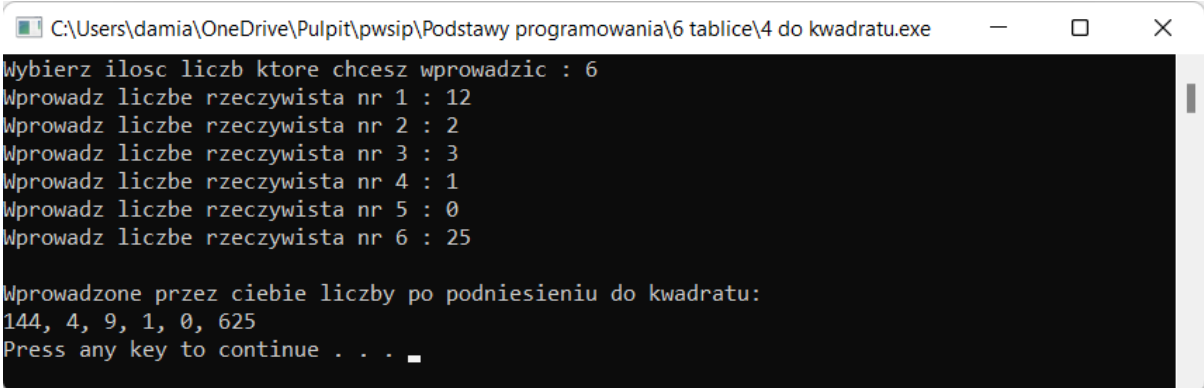
for(int i = 0; i < wielkoscTab; i++){
    printf("Wprowadz liczbe rzeczywista nr %d : ", i + 1);
    scanf("%f", &tablica[i]);
}

printf("\nWprowadzone przez ciebie liczby po podniesieniu do kwadratu: \n");
for(int i = 0; i < wielkoscTab; i++){
    printf("%.0f", pow(tablica[i], 2));
    if (i != wielkoscTab - 1){
        printf(", ");
    }
}
printf("\n");

system("pause");
return(0);
}

```

## Przykładowe uruchomienie



```

C:\Users\damia\OneDrive\Pulpit\pwsip\Podstawy programowania\6 tablice\4 do kwadratu.exe
Wybierz ilosc liczb ktore chcesz wprowadzic : 6
Wprowadz liczbe rzeczywista nr 1 : 12
Wprowadz liczbe rzeczywista nr 2 : 2
Wprowadz liczbe rzeczywista nr 3 : 3
Wprowadz liczbe rzeczywista nr 4 : 1
Wprowadz liczbe rzeczywista nr 5 : 0
Wprowadz liczbe rzeczywista nr 6 : 25

Wprowadzone przez ciebie liczby po podniesieniu do kwadratu:
144, 4, 9, 1, 0, 625
Press any key to continue . . .

```

## Treść zadania 5

Napisać program, w którym należy:

- wygenerować tablice 100 liczb losowych (funkcje rand() i srand())
- znaleźć minimum i maksimum
- obliczyć sumę i średnią arytmetyczną
- znaleźć element średni (najbliższy średniej arytmetycznej)
- znaleźć element środkowy (taki, który po posortowaniu byłby na środku tablicy - algorytm bez sortowania)

## Rozwiązanie

```
#include <stdio.h>
#include <stdlib.h>
#include <cstdlib>
#include <cstdio>
#include <math.h>

int main(){
    int szczeniwyNr, tab[100], min, max, tab2[sizeof(tab)/sizeof(*tab)], srodkowy1, srodkowy2, minLokalne, indeksMinLok;
    double suma = 0, srednia, sredni;

    printf("Podaj swój szczeniwy numer: "); // a) tworzy tablice z losowymi liczbami
    scanf("%d", &szczeniwyNr);
    srand(szczeniwyNr);
    for(int i = 0; i < sizeof(tab)/sizeof(*tab); i++){
        tab[i] = rand();
    }

    for (int i = 0; i < sizeof(tab)/sizeof(*tab); i++){ // kopiuje tab do tab2
        tab2[i] = tab[i];
    }

    min = tab[0]; // b) min i max
    for(int i = 1; i < sizeof(tab)/sizeof(*tab); i++){
        if (tab[i] < min){
            min = tab[i];
        }
    }
    max = tab[0];
    for(int i = 1; i < sizeof(tab)/sizeof(*tab); i++){
```

```

        if (tab[i] > max){
            max = tab[i];
        }
    }
    printf("\nb)Najmniejszy i największy element tablicy: %d i %d \n", min, max);

    for(int i = 0; i < sizeof(tab)/sizeof(*tab); i++){ // c)suma i srednia
        suma += tab[i];
    }
    srednia = suma / (sizeof(tab)/sizeof(*tab));
    printf("c)Suma i srednia z liczb w tablicy: %.0f i %.2f \n", suma, srednia);

    sredni = tab[0]; // d)sredni
    for(int i = 1; i < sizeof(tab)/sizeof(*tab); i++){
        if (abs(tab[i] - srednia) < abs(sredni - srednia)){
            sredni = tab[i];
        }
    }
    printf("d)Sredni element tablicy: %.0f\n", sredni);

    if(sizeof(tab2)/sizeof(*tab2) % 2 == 1 && sizeof(tab2)/sizeof(*tab2) > 1){ // e) element srodkowy nieparzysta tablica
        for(int i = 0; i < ceilf(float (sizeof(tab2)/sizeof(*tab2))/2); i++){ // znajdz minimum i zastap maxem polowe + 0.5 razy
            minLokalne = tab2[0];
            indeksMinLok = 0;
            for(int i = 1; i < sizeof(tab2)/sizeof(*tab2); i++){ // znajdz min
                if (tab2[i] < minLokalne){
                    minLokalne = tab2[i];
                    indeksMinLok = i;
                }
            }
        }
        srodkowy1 = tab2[indeksMinLok];
    }

```

```

        tab2[indeksMinLok] = max; // eliminacja najmniejszego elementu przez zastapienie najwiekszym
    }
    printf("e)Tablica zawiera nieparzysta ilosc elementow wiec ma jeden element srodkowy: %d. \n", srodkowy1);

} else if(sizeof(tab2)/sizeof(*tab2) % 2 == 0 && sizeof(tab2)/sizeof(*tab2) > 2) { // e) element srodkowy parzysta tablica
    for(int i = 0; i < (sizeof(tab2)/sizeof(*tab2)) / 2 + 1 ; i++){ // znajdz minimum i zastap maxem polowe + 1 razy
        minLokalne = tab2[0];
        indeksMinLok = 0;
        for(int i = 1; i < sizeof(tab2)/sizeof(*tab2); i++){ // znajdz min
            if (tab2[i] < minLokalne){
                minLokalne = tab2[i];
                indeksMinLok = i;
            }
        }
        if(i == (sizeof(tab2)/sizeof(*tab2)) / 2 - 1){ // przedostatnie min to srodkowy1
            srodkowy1 = tab2[indeksMinLok];
        } else if(i == (sizeof(tab2)/sizeof(*tab2)) / 2){ // ostatnie min to srodkowy2
            srodkowy2 = tab2[indeksMinLok];
        }
        tab2[indeksMinLok] = max; // eliminacja najmniejszego elementu przez zastapienie najwiekszym
    }
    printf("e)Tablica zawiera parzysta ilosc elementow wiec ma dwa elementy srodkowe: %d i %d . \n Ich srednia wynosi: %d
.\n", srodkowy1, srodkowy2, (srodkowy1 + srodkowy2) / 2);

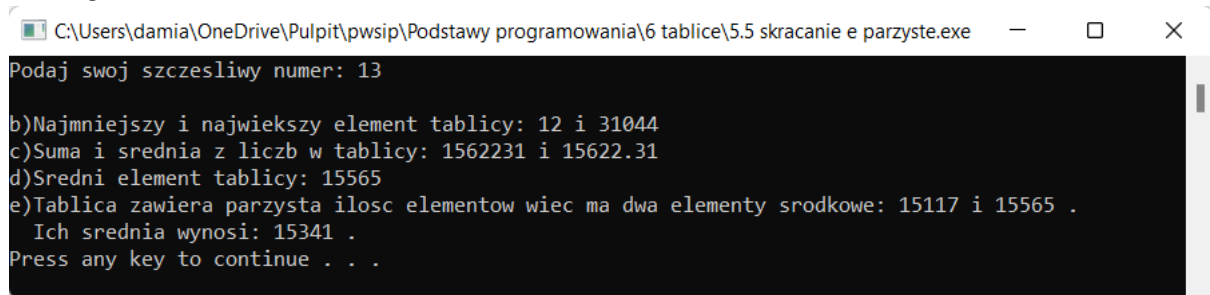
} else {
    printf("e)Nieprawidlowy rozmiar tablicy.\n");
}

system("pause");
return(0);
}

```



## Przykładowe uruchomienie



```
C:\Users\damia\OneDrive\Pulpit\pwsip\Podstawy programowania\6 tablice\5.5 skracanie e parzyste.exe
Podaj swój szczęśliwy numer: 13

b)Najmniejszy i największy element tablicy: 12 i 31044
c)Suma i srednia z liczb w tablicy: 1562231 i 15622.31
d)Sredni element tablicy: 15565
e)Tablica zawiera parzysta ilosc elementow wiec ma dwa elementy srodkowe: 15117 i 15565 .
   Ich srednia wynosi: 15341 .
Press any key to continue . . .
```

## Wnioski

Pierwsze cztery zadania były na podobnym poziomie trudności co te z poprzednich sprawozdań, jednak zadanie piąte było o wiele trudniejsze. Postanowiłem że w kodzie wielkość tablicy będzie podana tylko raz tak żeby można było ją bez problemu zmieniać, a wszędzie tam gdzie było trzeba skorzystać z ilości elementów w tablicy podstawiałem wielkość tablicy / wielkość elementu przechowywanego w tablicy. To znacznie ułatwiło testowanie pracy programu jednakże było niewystarczające w podpunkcie e). Kiedy program nie działał prawidłowo a ja nie mogłem znaleźć źródła problemu podczas czytania kodu, wprowadziłem elementy odpowiedzialne za wyświetlanie na monitorze stanu interesujących mnie zmiennych wraz z numerem iteracji pętli. Dzięki czemu udało mi się wychwycić błąd.

Tablice pozwalają na łatwiejsze operowanie na dużych zbiorach danych co jeszcze bardziej zwiększa przewagę użycia komputera w porównaniu do pracy człowieka.