

Imię i nazwisko	Kierunek	Rok studiów i grupa
Marek Kubicki	Informatyka techniczna ITE	1 rok grupa 5
Data zajęć	Numer i temat sprawozdania	
10.11.2022	Lab 6. – Pętle, tablice	

Cel:

- Opanowanie podstaw wykorzystania tablic i pętli w C.

Przebieg zajęć:

Utworzyłem katalog roboczy lab_6 i kopowałem do niego odpowiednie pliki

```
sh-4.2$ pwd
/home/METAL/markubi2/lab_6/loop
sh-4.2$ ls
oblicz_PI.c  simple_loop.c  skr1
```

skr1 to skrypt ułatwiający szybkie kompilowanie i uruchamianie programu

Następnie zacząłem wykonywać polecenia zawarte w pliku simple_loop.c

Wprowadziłem system numeracji tablic aby można było sprawdzić działanie całego programu na raz.

Program zawiera dużo powtarzających się elementów (np. pętla for wypisująca wynik przypisywania), ułatwiło to szybkie napisanie programu oraz debugowanie, ponieważ po skopiowaniu danego fragmentu kodu (i zmienieniu nazw zmiennych) mogłem być pewny że konstrukcja jest poprawna.

```
// 1. inicjowanie tablicy w momencie definowania:
// a. wszystkie wartości
printf("1. a\n");
int tablica1_int[10]={0, 1, 2, 3, 4, 5, 6, 7, 8, 9};

//tablica1_int[10];
for(i=0; i<10; i++){
    printf("Iteracja %d: tablica1_int[%d] = %d\n", i, i, tablica1_int[i]);
}
```

```
1. a
Iteracja 0: tablica1_int[0] = 0
Iteracja 1: tablica1_int[1] = 1
Iteracja 2: tablica1_int[2] = 2
Iteracja 3: tablica1_int[3] = 3
Iteracja 4: tablica1_int[4] = 4
Iteracja 5: tablica1_int[5] = 5
Iteracja 6: tablica1_int[6] = 6
Iteracja 7: tablica1_int[7] = 7
Iteracja 8: tablica1_int[8] = 8
Iteracja 9: tablica1_int[9] = 9
```

```
// b. początkowe wartości
printf("\n1. b\n");
int tablica2_int[10]={0, 1, 2};

for(i=0; i<10; i++){
    printf("Iteracja %d: tablica2_int[%d] = %d\n", i, i, tablica2_int[i]);
}
```

```
1. b
Iteracja 0: tablica2_int[0] = 0
Iteracja 1: tablica2_int[1] = 1
Iteracja 2: tablica2_int[2] = 2
Iteracja 3: tablica2_int[3] = 0
Iteracja 4: tablica2_int[4] = 0
Iteracja 5: tablica2_int[5] = 0
Iteracja 6: tablica2_int[6] = 0
Iteracja 7: tablica2_int[7] = 0
Iteracja 8: tablica2_int[8] = 0
Iteracja 9: tablica2_int[9] = 0
```

```
// 2. nadanie wartości w pętli
// a. zależnych od wartości zmiennej sterującej
printf("\n2. a\n");
int tablica3_int[10];

for(i=0; i<10; i++){
    tablica3_int[i]=i;
}
for(i=0; i<10; i++){
    printf("Iteracja %d: tablica3_int[%d] = %d\n", i, i, tablica3_int[i]);
}
```

```
2. a
Iteracja 0: tablica3_int[0] = 0
Iteracja 1: tablica3_int[1] = 1
Iteracja 2: tablica3_int[2] = 2
Iteracja 3: tablica3_int[3] = 3
Iteracja 4: tablica3_int[4] = 4
Iteracja 5: tablica3_int[5] = 5
Iteracja 6: tablica3_int[6] = 6
Iteracja 7: tablica3_int[7] = 7
Iteracja 8: tablica3_int[8] = 8
Iteracja 9: tablica3_int[9] = 9
```

```
// b. wczytywanych z klawiatury za pomocą scanf
printf("\n2. b\n");

int tablica4_int[10];
int a2;

for(i=0; i<10; i++){
    printf("Podaj liczbe\n");
    scanf("%d",&a2);
    tablica4_int[i]=a2;
}
for(i=0; i<10; i++){
    printf("Iteracja %d: tablica4_int[%d] = %d\n", i, i,
tablica4_int[i]);
}
```

2. b		
Podaj liczbe	Podaj liczbe	Iteracja 0: tablica4_int[0] = 12
12	123	Iteracja 1: tablica4_int[1] = 312
Podaj liczbe	Podaj liczbe	Iteracja 2: tablica4_int[2] = 124
312	12	Iteracja 3: tablica4_int[3] = 121
Podaj liczbe	Podaj liczbe	Iteracja 4: tablica4_int[4] = 3
124	3	Iteracja 5: tablica4_int[5] = 123
Podaj liczbe	Podaj liczbe	Iteracja 6: tablica4_int[6] = 12
121	123	Iteracja 7: tablica4_int[7] = 3
Podaj liczbe	Podaj liczbe	Iteracja 8: tablica4_int[8] = 123
3	21	Iteracja 9: tablica4_int[9] = 21

```

// 3. zamiana pętli for nadawania wartości na pętlę while
printf("\n3\n");
int tablica5_int[10];
int b2;
i=0;

while(i<10){
    printf("Podaj liczbe\n");
    scanf("%d",&b2);
    tablica5_int[i]=b2;
    i++;
}
for(i=0; i<10; i++){
    printf("Iteracja %d: tablica5_int[%d] = %d\n", i, i,
tablica5_int[i]);
}

```

3	Podaj liczbe	Podaj liczbe	Iteracja 0: tablica5_int[0] = 312
	312	4436	Iteracja 1: tablica5_int[1] = 123
	Podaj liczbe	Podaj liczbe	Iteracja 2: tablica5_int[2] = 123
	123	2	Iteracja 3: tablica5_int[3] = 123
	Podaj liczbe	Podaj liczbe	Iteracja 4: tablica5_int[4] = 523
	123	34	Iteracja 5: tablica5_int[5] = 4436
	Podaj liczbe	Podaj liczbe	Iteracja 6: tablica5_int[6] = 2
	123	12	Iteracja 7: tablica5_int[7] = 34
	Podaj liczbe	Podaj liczbe	Iteracja 8: tablica5_int[8] = 12
	523	3	Iteracja 9: tablica5_int[9] = 3

```

// 4. zamiana poniższej pętli for na pętlę do
// (jak zwykle najlepiej wykomentować starą pętlę i wpisać poniżej nową pętlę)
/*
    for(i<10; i++++){

        printf("Iteracja %d: tablica_int[%d] = %d\n", i, i, tablica_int[i]);

    }
*/
    printf("\n4\n");
    i=0;
    do{

        printf("Iteracja %d: tablica5_int[%d] = %d\n", i, i, ta-
blica5_int[i]);
        i++;
    }while(i<10);

```

```
4
Iteracja 0: tablica5_int[0] = 312
Iteracja 1: tablica5_int[1] = 123
Iteracja 2: tablica5_int[2] = 123
Iteracja 3: tablica5_int[3] = 123
Iteracja 4: tablica5_int[4] = 523
Iteracja 5: tablica5_int[5] = 4436
Iteracja 6: tablica5_int[6] = 2
Iteracja 7: tablica5_int[7] = 34
Iteracja 8: tablica5_int[8] = 12
Iteracja 9: tablica5_int[9] = 3
```

```
// 5. wyszukiwanie wartości maksymalnej w tablicy (pętla for)
printf("\n5\n");
int max=tablica5_int[0];
for(i=0; i<10; i++){
    if(max<tablica5_int[i]){
        max=tablica5_int[i];
    }
}
printf("Maksymalna wartosc z tablica5_int[%d] = %d\n", i, max);
```

```
5
Maksymalna wartosc z tablica5_int[10] = 4436
```

```
// 6. obliczanie sumy elementów tablicy (pętla while)
printf("\n6\n");
int suma1=0;
i=0;
while(i<10){
    suma1 += tablica5_int[i];
    i++;
}
printf("Suma wartosci elementow z tablica5_int[%d] = %d\n", i, suma1);
```

```
6
Suma wartosci elementow z tablica5_int[10] = 5691
```

```
// 7. tablica z wartościami losowymi i wyszukiwanie zadanych wartości
// a. nowa definicja tablicy (np. tablicy znaków)

printf("\n7. a\n");
char string[3] = {'s','a','m'};

printf("String = %s\n", string);
```

```
7. a
String = sam
```

```
// b. nadawanie wartości losowych wyrazom tablicy
// (np. liczby z zakresu 33-126, czyli odpowiadające znakom ASCII)
// -> odpowiednie wydruki
printf("\n7. b\n");
char tablica6_char[10];
srand(time(NULL));
for(i=0; i<10; i++){
    tablica6_char[i]=(rand() / (RAND_MAX + 1.0) * 93.0)+33;
}
for(i=0; i<10; i++){
    printf("Iteracja %d: tablica6_char[%d] = %c\n", i, i, tablica6_char[i]);
}
```

7. b

Iteracja 0: tablica6 char[0] = 5

```
// c. sprawdzenie losowości - obliczenie wartości średniej w tablicy
//      (w odpowiedniej pętli) - porównanie z teoretyczną średnią
//      -> odpowiednie wydruki

printf("\n7. c\n");

double suma2=0;
for(i=0; i<10; i++){
    suma2 += tablica6_char[i];
}
// teoretyczna średnia
double teo_srednia = (126) / 2;
printf("Srednia wartosc z tablica6_char[] to = %lf\n teoretyczna srednia
to = %lf\n ich ruznica to = %lf\n", (suma2/10), teo_srednia, (suma2/10) - teo_srednia);
```

(Musiałem sprawdzić działanie polecenia 7. c. na innym urządzeniu, poniżej podany jest wynik działania tej części programu)

```
m@m-VirtualBox:~/Lab_6$ pwd
/home/m/Lab_6
m@m-VirtualBox:~/Lab_6$ ./skr1
```

7. c

Srednia wartosc z tablica6_char[] to = 74.900000
teoretyczna srednia to = 63.000000
ich ruznica to = 11.900000

```
// d. wyszukiwanie zadanej wartości
//      -> odpowiednie wydruki w momencie znalezienia wartości
//      np. "Znaleziono wartość ... w %d elemencie tablicy\n", ...
```

```
// d1. wyszukanie tylko jednej (pierwszej) wartości - pętla while
//       (lub pętla for i break)
printf("\n7. d1\n");
int sz1;
printf("Podaj szukana wartosc\n");
scanf("%d",&sz1);
int sz1nr=-1;
for(i=0; i<10; i++){
    if(tablica5_int[i]==sz1){
        sz1nr=i;
        break;
    }
}
if(sz1nr==0){
    printf("nie znaleziono wartość %d w tablicy5\n", sz1);
}
else{
    printf("znaleziono wartość %d w %d elemencie tablicy5\n",sz1, sz1nr);
}
```

7. d1

Podaj szukana wartosc

123

znaleziono wartość 123 w 1 elemencie tablicy5

```
// d2. wyszukanie (i wypisanie) wszystkich wartości - pętla for
//       (lub pętla while)
printf("\n7. d2\n");
int sz2;
printf("Podaj szukana wartosc\n");
scanf("%d",&sz2);
int sz2nr=-1;
for(i=0; i<10; i++){
    if(tablica5_int[i]==sz2){
        printf("znaleziono wartość %d w %d elemencie tablicy5\n",sz2, i);
        sz2nr=1;
    }
}
if(sz2nr==0){
    printf("nie znaleziono wartość %d w tablicy5\n", sz2);
}
```

7. d2

Podaj szukana wartosc

123

znaleziono wartość 123 w 1 elemencie tablicy5

znaleziono wartość 123 w 2 elemencie tablicy5

znaleziono wartość 123 w 3 elemencie tablicy5

Wnioski

Tablice w języku C są niezwykle użytecznym narzędziem. Umożliwiają one łatwe grupowanie i zapisywanie(tymczasowe) dużej ilości zmiennych. Pomagają one też wykonywać operacje na wielu zmiennych. Za ich pomocą można też zapisać wyniki programu a potem wykonać na nich różne operacje aby sprawdzić ich poprawność.