

Programowanie w języku C (EAR1S02005)

Politechnika Białostocka - Wydział Elektryczny
Automatyka i Robotyka, semestr II, studia stacjonarne I stopnia
Rok akademicki 2020/2021

Zajęcia nr 5 (09.03.2021)

dr inż. Jarosław Forenc

Język C - tablica elementów

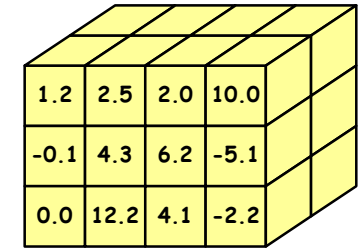
- **Tablica** - ciągły obszar pamięci, w którym umieszczone są elementy tego samego typu

wektor

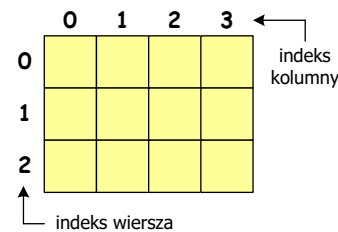
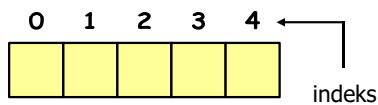
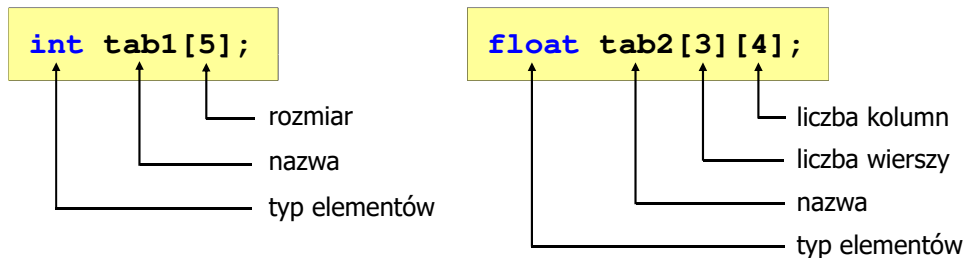
5	3	-2	1	-4
---	---	----	---	----

macierz

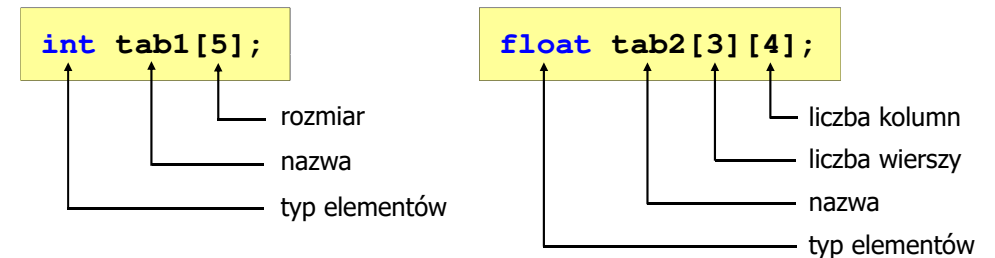
a	c	d	m
p	d	q	l
a	t	x	v



Język C - deklaracja tablicy



Język C - deklaracja tablicy



- **Rozmiar** tablicy to wartość:
 - całkowita, dodatnia
 - znana na etapie kompilacji programu (stała liczbowa: `5` `#define N 5` `const int n = 5;`)

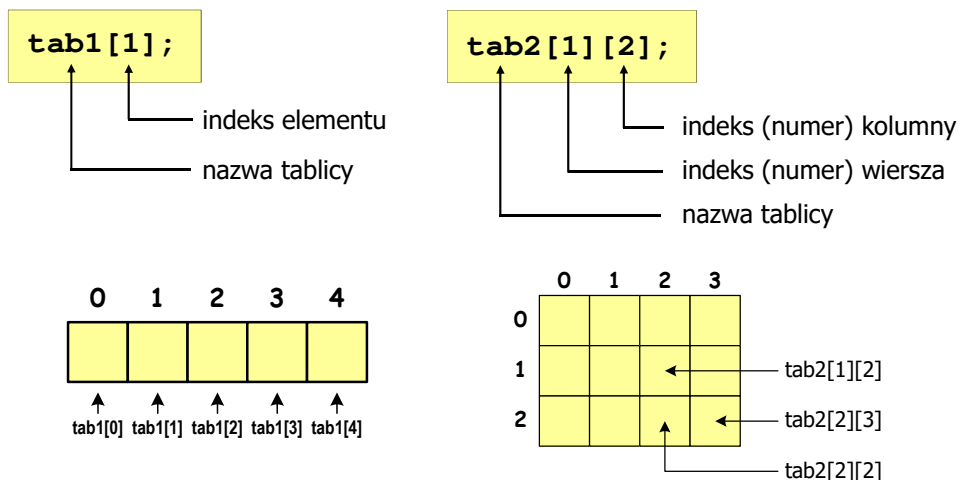
`int tab1[5];`

`int tab1[N];`

`int tab1[n];`

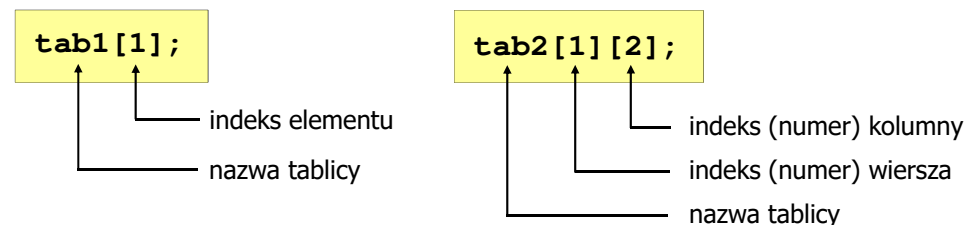
Język C - odwołania do elementów tablicy

[] - dwuargumentowy operator indeksowania



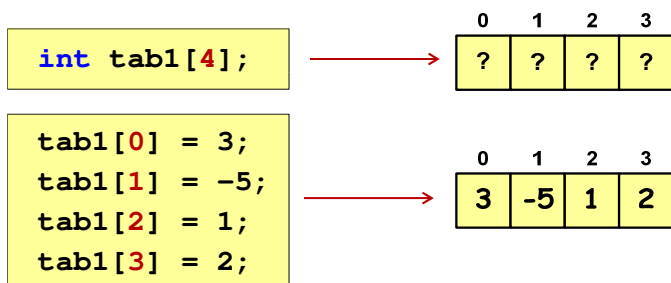
Język C - odwołania do elementów tablicy

[] - dwuargumentowy operator indeksowania

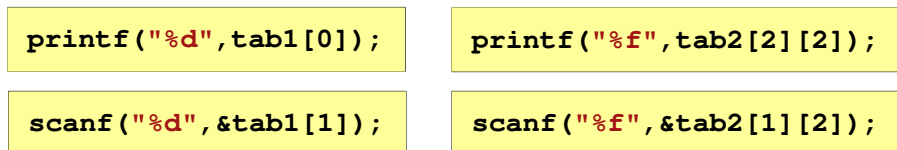


- Indeks:
 - stała liczbowa, np. 0, 1, 10
 - nazwa zmiennej, np. i, idx
 - wyrażenie, np. i*j+5
- Brak sprawdzania poprawności indeksów!

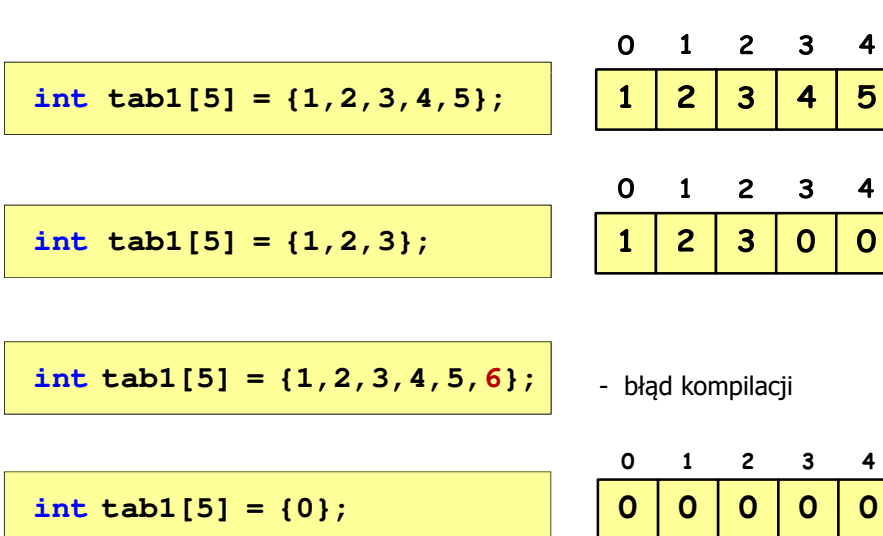
Język C - odwołania do elementów tablicy



- Każdy element tablicy traktowany jest jak zmienna określonego typu



Język C - inicjalizacja elementów tablicy



Język C - inicjalizacja elementów tablicy

```
int tab2[2][3] = {{1,2,3},{4,5,6}};
```

	0	1	2
0	1	2	3
1	4	5	6

```
int tab2[2][3] = {1,2,3,4,5,6};
```

```
int tab2[2][3] = {1,2,3,4};
```

	0	1	2
0	1	2	3
1	4	0	0

```
int tab2[2][3] = {{1},{4,5}};
```

	0	1	2
0	1	0	0
1	4	5	0

Język C - generowanie pseudolosowe liczb

- `rand()` - zwraca liczbę pseudolosową - zakres: `0 ... RAND_MAX` (`0 ... 32767`)
- `srand()` - inicjalizuje generator liczb pseudolosowych
- plik nagłówkowy: `stdlib.h` (`time.h`)

```
int x, y, z;  
srand((unsigned int) time(NULL));  
x = rand(); // zakres <0,32767>  
y = rand() % 100; // zakres <0,99>  
z = rand() % (b-a+1)-a; // zakres <a,b>
```

Język C - operacje na wektorze

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

```
#define N 10
```

```
int main(void)
```

```
{
```

```
    int tab1[N], i;
```

```
    /* generowanie elementów tablicy */
```

```
    srand((unsigned int) time(NULL));
```

```
    for (i=0; i<N; i++)  
        tab1[i] = rand() % 20;
```

0	1	2	3	4	5	6	7	8	9
7	12	1	16	1	11	14	5	19	8

Język C - operacje na wektorze

```
/* wyświetlenie elementów tablicy */
```

```
printf("Elementy tablicy:\n");  
for (i=0; i<N; i++)  
    printf("%d ", tab1[i]);  
printf("\n");
```

```
Elementy tablicy:  
7 12 1 16 1 11 14 5 19 8
```

0	1	2	3	4	5	6	7	8	9
7	12	1	16	1	11	14	5	19	8

N = 10

Język C - operacje na wektorze

```
/* wyszukanie elementu o najmniejszej wartości */  
  
int min;  
  
min = tabl[0];  
for (i=1; i<N; i++)  
    if (tabl[i]<min)  
        min = tabl[i];  
printf("Wartosc elementu najmniejszego: %d\n",min);
```

Wartosc elementu najmniejszego: 1

0	1	2	3	4	5	6	7	8	9
7	12	1	16	1	11	14	5	19	8

N = 10

Język C - operacje na wektorze

```
/* suma i średnia arytmetyczna elementów tablicy */  
  
int suma = 0;  
float srednia;  
  
for (i=0; i<N; i++)  
    suma = suma + tabl[i];  
srednia = (float) suma/N;  
printf("Suma: %d, srednia: %g\n", suma, srednia);
```

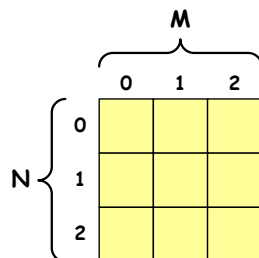
Suma: 94, srednia: 9.4

0	1	2	3	4	5	6	7	8	9
7	12	1	16	1	11	14	5	19	8

N = 10

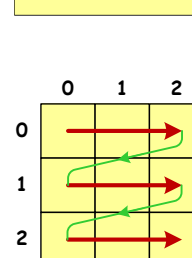
Język C - operacje na macierzy

```
#include <stdio.h>  
#include <stdlib.h>  
#include <time.h>  
  
#define N 3 /* liczba wierszy */  
#define M 3 /* liczba kolumn */  
  
int main(void)  
{  
    int tab2[N][M];  
    int i, j;
```

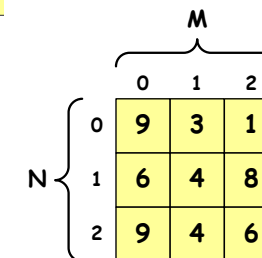


Język C - operacje na macierzy

```
/* generowanie pseudolosowe elementów macierzy */  
  
srand((unsigned int) time(NULL));  
  
for (i=0; i<N; i++)  
    for (j=0; j<M; j++)  
        tab2[i][j] = rand() % 10;
```



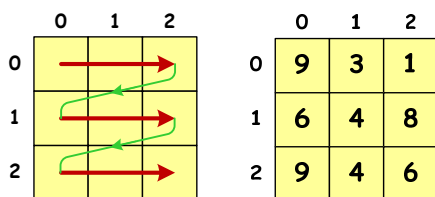
kolejność zapisywania
wartości elementów
macierzy



Język C - operacje na macierzy

```
/* wyświetlenie elementów macierzy */  
for (i=0; i<N; i++)  
{  
    for (j=0; j<M; j++)  
        printf("%3d", tab2[i][j]);  
    printf("\n");  
}
```

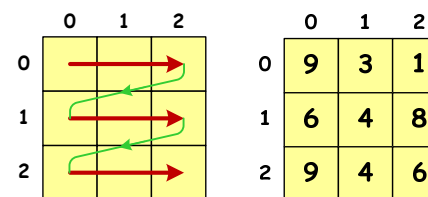
```
9 3 1  
6 4 8  
9 4 6
```



Język C - operacje na macierzy

```
/* poszukiwanie elementu o wartości minimalnej */  
int min = tab2[0][0];  
for (i=0; i<N; i++)  
    for (j=0; j<M; j++)  
        if (tab2[i][j] < min)  
            min = tab2[i][j];  
printf("Wartosc min: %d\n", min);
```

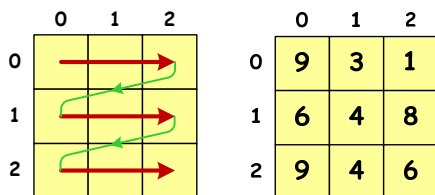
```
Wartosc min: 1
```



Język C - operacje na macierzy

```
/* sumy elementów w poszczególnych wierszach */  
for (i=0; i<N; i++)  
{  
    suma = 0;  
    for (j=0; j<M; j++)  
        suma = suma + tab2[i][j];  
    printf("Suma wiersza %d = %d\n", i, suma);  
}
```

```
Suma wiersza 0 = 13  
Suma wiersza 1 = 18  
Suma wiersza 2 = 19
```



Język C - operacje na macierzy

```
/* sumy elementów w poszczególnych kolumnach */  
for (j=0; j<M; j++)  
{  
    suma = 0;  
    for (i=0; i<N; i++)  
        suma = suma + tab2[i][j];  
    printf("Suma kolumny %d = %d\n", j, suma);  
}
```

```
Suma kolumny 0 = 24  
Suma kolumny 1 = 11  
Suma kolumny 2 = 15
```

