

Informatyka 1

Politechnika Białostocka - Wydział Elektryczny
Elektrotechnika, semestr II, studia niestacjonarne I stopnia
Rok akademicki 2017/2018

Wykład nr 7 (27.04.2018)

dr inż. Jarosław Forenc

Plan wykładu nr 7

- Język C - tablice jednowymiarowe (wektory)
 - deklaracja tablicy
 - odwołania do elementów
 - inicjalizacja tablicy
 - generator liczb pseudolosowych
 - operacje na wektorze

Język C - operacje na dużej ilości danych

```
#include <stdio.h>

int main(void)
{
    double U1, U2, U3, U4, U5;
    double I1, I2, I3, I4, I5;
    double R1, R2, R3, R4, R5;

    U1 = 5.0;
    U2 = 10.0;
    U3 = 15.0;
    U4 = 20.0;
    U5 = 25.0;

    I1 = 0.16;
    I2 = 0.21;
    I3 = 0.27;
    I4 = 0.33;
    I5 = 0.36;
```

Język C - operacje na dużej ilości danych

```
R1 = U1/I1;  
R2 = U2/I2;  
R3 = U3/I3;  
R4 = U4/I4;  
R5 = U5/I5;
```

```
printf("R1 = %f\n", R1);  
printf("R2 = %f\n", R2);  
printf("R3 = %f\n", R3);  
printf("R4 = %f\n", R4);  
printf("R5 = %f\n", R5);
```

```
return 0;
```

```
}
```

```
R1 = 31.250000  
R2 = 47.619048  
R3 = 55.555556  
R4 = 60.606061  
R5 = 69.444444
```

Język C - operacje na dużej ilości danych (tablica)

```
#include <stdio.h>

int main(void)
{
    double U[5] = { 5.0, 10.0, 15.0, 20.0, 25.0 };
    double I[5] = { 0.16, 0.21, 0.27, 0.33, 0.36 };
    double R[5];
    int i;

    for (i=0; i<5; i++)
        R[i] = U[i]/I[i];

    for (i=0; i<5; i++)
        printf("R%d = %f\n", i+1, R[i]);

    return 0;
}
```

```
R1 = 31.250000
R2 = 47.619048
R3 = 55.555556
R4 = 60.606061
R5 = 69.444444
```

	0	1	2	3	4
U	5.0	10.0	15.0	20.0	25.0
I	0.16	0.21	0.27	0.33	0.36
R					

Język C - tablica elementów

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

wektor

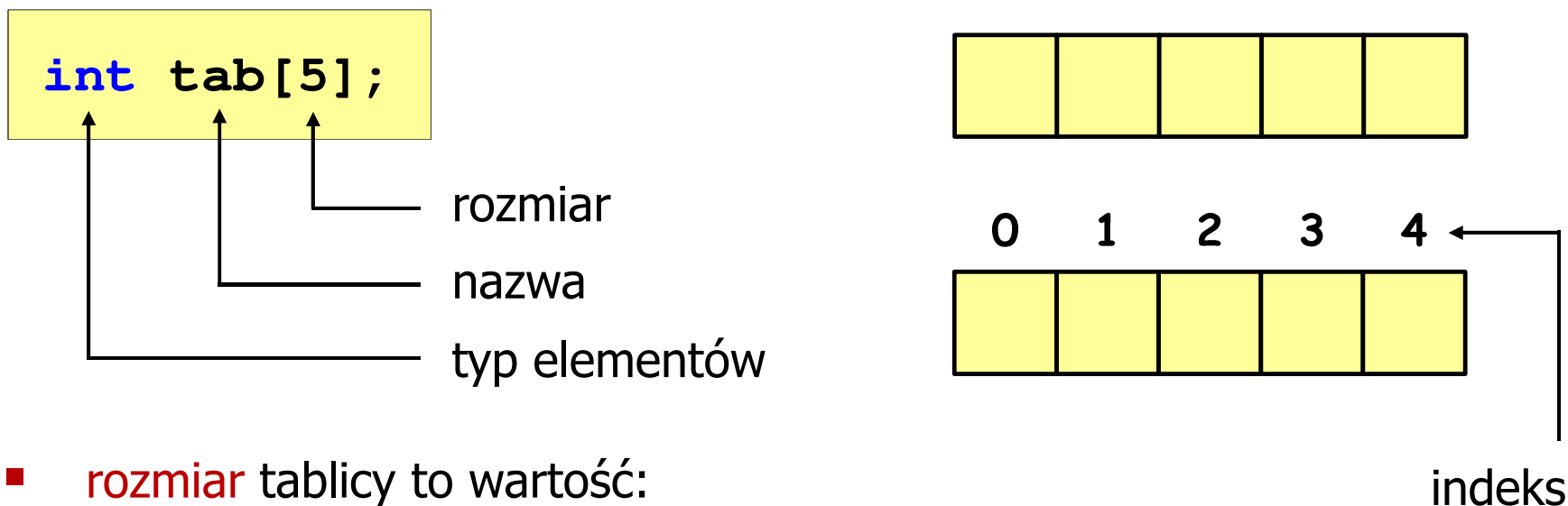
5	3	-2	1	-4
---	---	----	---	----

macierz

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

1.2	2.5	2.0	10.0
-0.1	4.3	6.2	-5.1
0.0	12.2	4.1	-2.2

Język C - deklaracja tablica jednowymiarowej



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

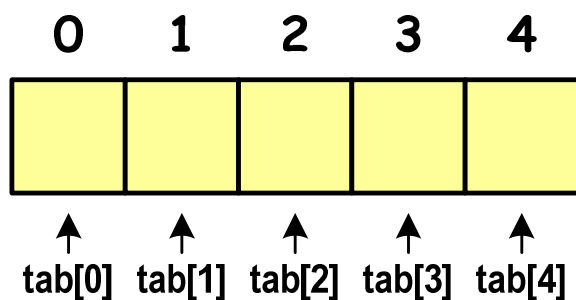
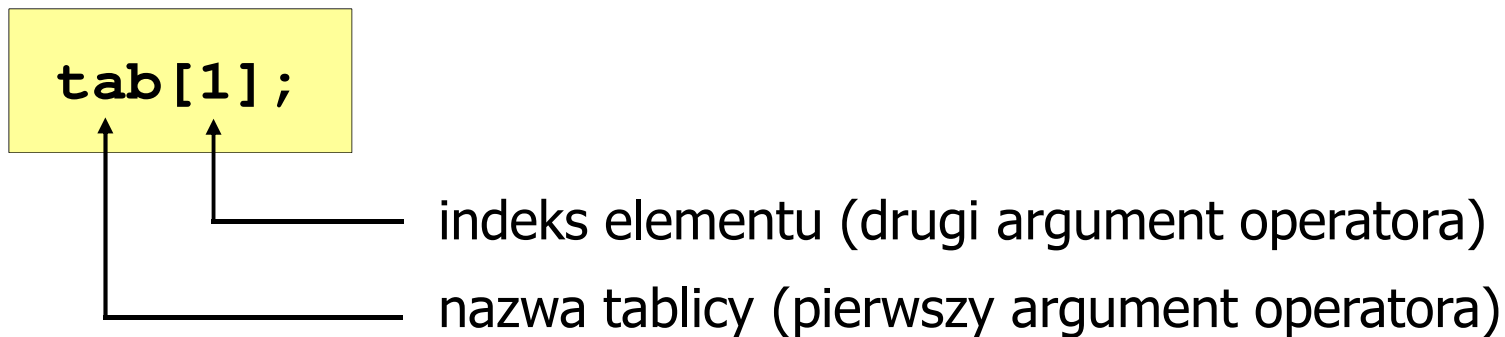
```
int tab[5];
```

```
int tab[N];
```

```
int tab[n];
```

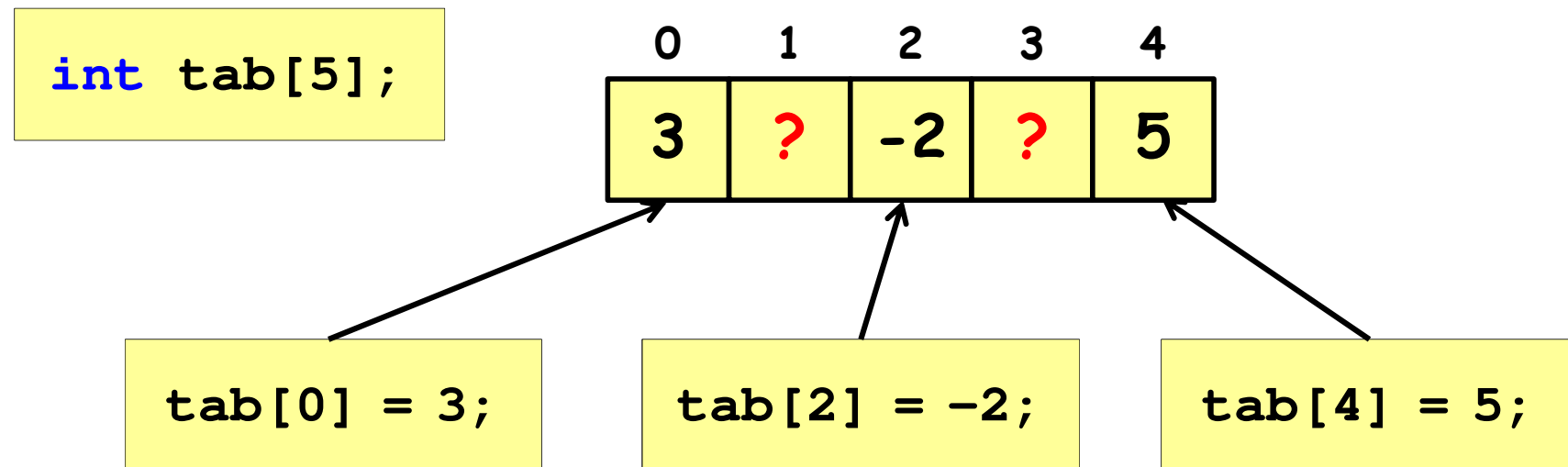
Język C - odwołania do elementów tablicy

[] - dwuargumentowy operator indeksowania



- indeks:
 - stała liczbowa, np. **0**, **1**, **10**
 - nazwa zmiennej, np. **i**, **idx**
 - wyrażenie, np. **i*j+5**

Język C - odwołania do elementów tablicy



- Każdy element tablicy traktowany jest tak samo jak zmienna typu `int`

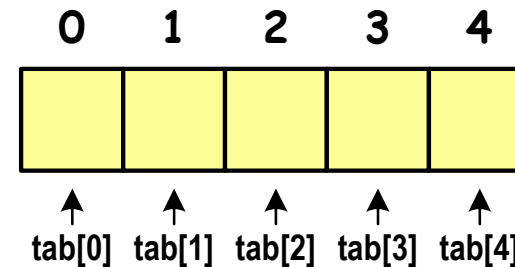
```
printf("%d", tab[0]);
```

```
scanf("%d", &tab[1]);
```

Język C - odwołania do elementów tablicy

- Przy odwołaniach do elementów tablicy kompilator nie sprawdza poprawności indeksów

```
int tab[5];  
tab[5] = 10;
```



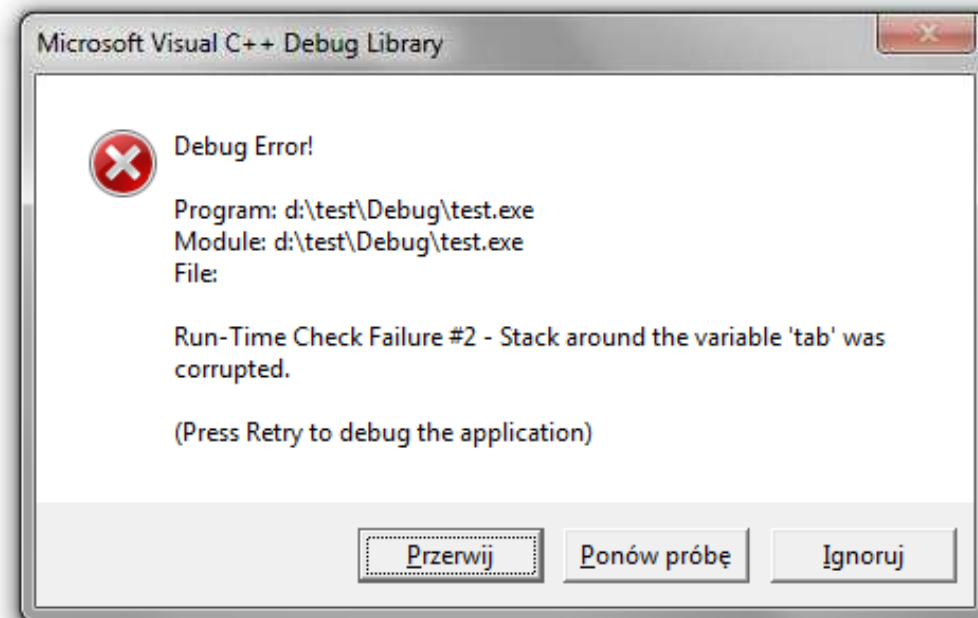
- **błąd!!!** - nie istnieje element **tab[5]**

- Kompilator nie zasygnalizuje błędu
- Program wykona operację
- Środowisko programistyczne może zasygnalizować problem

Język C - odwołania do elementów tablicy

- Przy odwołaniach do elementów tablicy kompilator nie sprawdza poprawności indeksów

```
int tab[5];  
tab[5] = 10;
```



Język C - odwołania do elementów tablicy

- Zapisanie wartości **1** do wszystkich elementów tablicy

```
int tab[5];
```

```
tab[0] = 1;
```

```
tab[1] = 1;
```

```
tab[2] = 1;
```

```
tab[3] = 1;
```

```
tab[4] = 1;
```

0	1	2	3	4
1	1	1	1	1

```
int tab[5], i;
```

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

```
    tab[i] = 1;
```

Język C - inicjalizacja tablicy jednowymiarowej

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

0	1	2	3	4
1	2	3	4	5

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

0	1	2	3	4
1	2	3	0	0

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

- błąd kompilacji

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

0	1	2	3	4
1	2	3	4	5

Język C - generator liczb pseudolosowych

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

```
int x, y;
srand( (unsigned int) time(NULL) );
x = rand();           // zakres <0, 32767>
y = rand() % 100;     // zakres <0, 99>
```

Język C - operacje na wektorze

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

```
#define N 10
```

```
int main(void)
```

```
{
```

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

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

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

```
        tab[i] = rand() % 100;
```

0	1	2	3	4	5	6	7	8	9
79	44	15	5	39	81	6	41	53	17

Język C - operacje na wektorze

```
for (i=0; i<N; i++)  
    printf("%4d", tab[i]);  
printf("\n");  
  
return 0;  
}
```

79 44 15 5 39 81 6 41 53 17

0	1	2	3	4	5	6	7	8	9
79	44	15	5	39	81	6	41	53	17

N = 10

Koniec wykładu nr 7

Dziękuję za uwagę!
(następny wykład: 11.05.2018)