

# Programowanie Obiektowe

---

Grzegorz Hałdaś

Technologie Internetowe



# Plan Wykładu

---

- Sprawy organizacyjne
- Spis literaturowy
- Pierwsze programy i krótkie zapoznanie się z środowiskiem programistycznym
- Typy danych
- Instrukcje sterujące i pętle
- Operatory - działanie na wybranych operatorach



## Kontakt

---

Pok. 302 bud. A

Tel. 17 86 51 112

E-mail: [ghaldas@prz.edu.pl](mailto:ghaldas@prz.edu.pl)

Strona domowa: [ghaldas.v.prz.edu.pl](http://ghaldas.v.prz.edu.pl)



## Spis literaturowy

---

- **Jerzy Grębosz , Symfonia C++ standard, Wydawnictwo „Edition 2000”, Kraków 2005.**
- **Stroustrup Bjarne, Język C++. Kompendium wiedzy, Helion, Gliwice 2014.**
- **Tony L. Hansen, C++. Zadania i odpowiedzi, WNT, Warszawa 1994.**



# Podstawowe cechy języka C++

---

- Język C++

- język wolnego formatu
- język programowania zorientowanego obiektowo
- ma możliwość programowania klasycznego: proceduralnego i strukturalnego
- rozróżnienia małe i duże litery



# Operacje we/wy

---

- `iostream`

Standardowe strumienie wyjściowe

- `cout`

- `cerr`

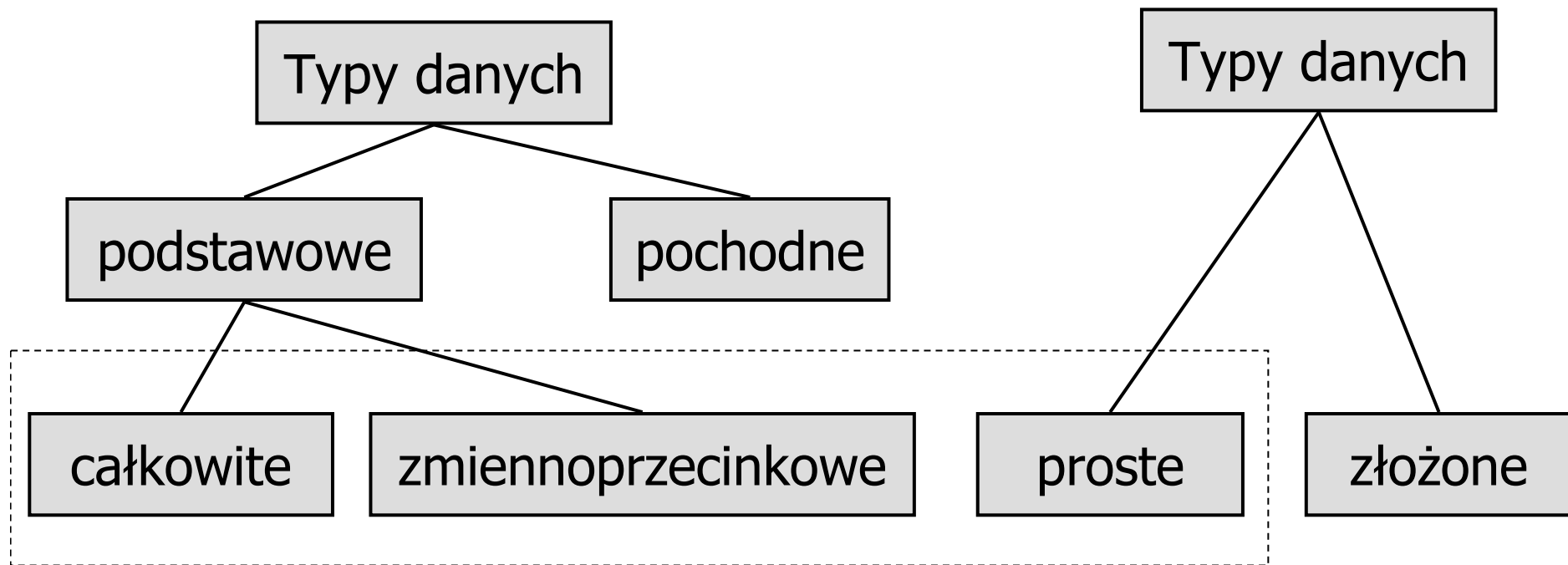
- `clog`

Standardowy strumień wejściowy

- `cin`



# Typy danych





# Typy całkowite

nazwa	liczba bajtów	Zakres liczbowy
char	1	-128..127
short int	2	-32768..32767 ( $-2^{15}..2^{15}-1$ )
int	4	-2 147 483 648..2 147 483 647 ( $-2^{31}..2^{31}-1$ )
long int	4	j.w.
long long	8	$-2^{63}..2^{63}-1$





# Typy zmiennoprzecinkowe

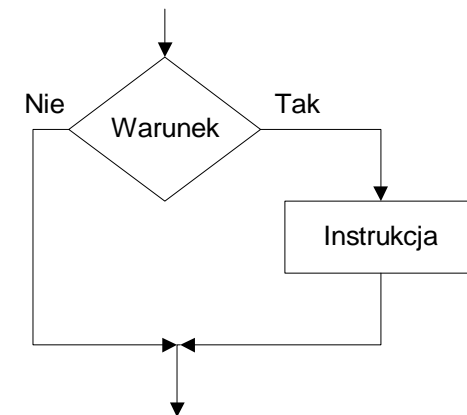
nazwa	liczba bajtów	zakres liczbowy	precyzja
float	4	$\pm 3.4e^{\pm 38}$	7cyfr znaczących
double	8	$\pm 1.7e^{\pm 308}$	15 cyfr
long double	10	$-3.4..1.1 e^{\pm 4932}$	19 cyfr



# Instrukcje sterujące if

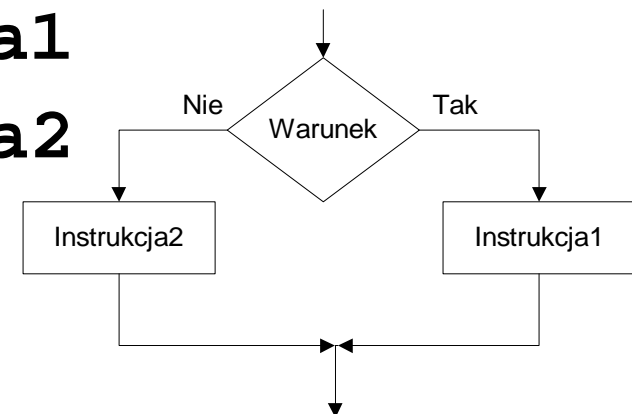
- **if**

**if (warunek) instrukcja**



- **if else**

**if (warunek) instrukcja1  
else instrukcja2**



- **if else if**



# Instrukcja sterująca switch

## ■ switch

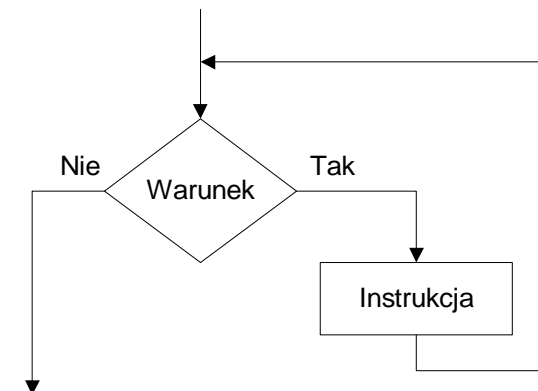
```
switch(wyrażenie)
{
    case wartość_wyrażenia1:
        instrukcja1;
        break;
    case wartość_wyrażenia2:
        instrukcja2;
        break;
    default:
        instrukcja_dla_pozostałych_wartości;
        break;
}
```



# Pętle while i do while

## ■ while

`while(warunek) instrukcja`

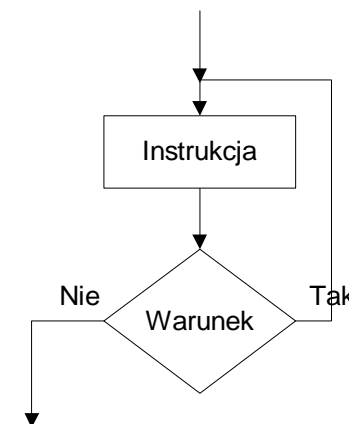


## ■ do while

`do`

`instrukcja`

`while(warunek);`

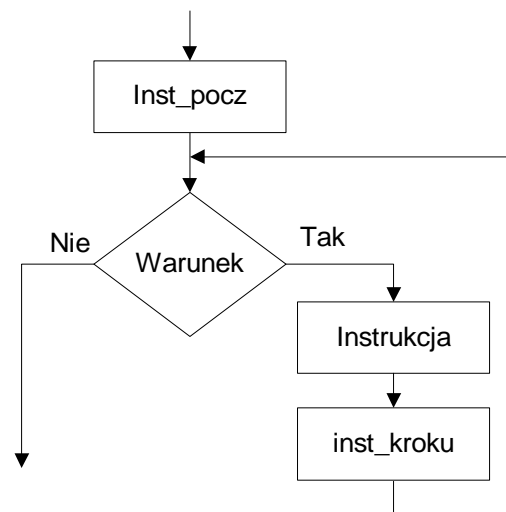




# Pętla for

**for**

```
for( inst_pocz ; warunek ; inst_kroku ) instrukcja
```





# Operatory

## ■ arytmetyczne

**+, -, \*, /, %, ++, --**

przykłady:

```
delta=b*b-4.0*a*c; // delta=((b*b)-((4.0*a)*c));
```

7/2      wynosi 3 (bo dzielenie całkowite)

7.0/2   wynosi 3.5 (dzielenie zmiennoprzecinkowe)

7%3      wynosi 1 (reszta z dzielenia)

```
int i=1, j;
```

```
i++;            =>    i=i+1;
```

```
i=1;
```

```
j=++i;    => i=i+1; j=i;            czyli i=2, j=2
```

```
i=1;
```

```
j=i++;    => j=i; i=i+1;            czyli i=2, j=1;
```



# Operator

- relacji

>, <, >=, <=, !=, ==

- logiczne

&&, ||, !

np.

`a > 1.0 && a <= 3.0`

`a < -12 || a > 10`

`!a ==> a == 0`



# Operator

## ■ bitowe

**&, |, ^, ~, >>, <<**

```
short int a=1234, b=3456, c; // a=0x04d2, b=0x0d80
```

```
rozpisując na bity // a=002322, b=006600
```

```
a => 0000 0100 1101 0010 //0b00000010011010010
```

```
b => 0000 1101 1000 0000 //0b0000110110000000
```

---

```
c=a&b; // 0000 0100 1000 0000 (0x0480 = 1152)
```

```
c=a|b; // 0000 1101 1101 0010 (0x0dd2 = 3538)
```

```
c=a^b; // 0000 1001 0101 0010 (0x0952 = 2386)
```

```
c=a<<1; //0000 1001 1010 0100 (0x09a4 = 2468)
```

```
c=a>>2; //0000 0001 0011 0100 (0x0134 = 308)
```





# Operator

- przypisania

`=,`

`+=, -=, *=, /=, %=`

`&=, |=, ^=, >>=, <<=`

`np.`

`a=b=c; //=> (a=(b=c));`

`a+=b; //=> a=a+b;`

`a>>=2; //=> a=a>>2;`