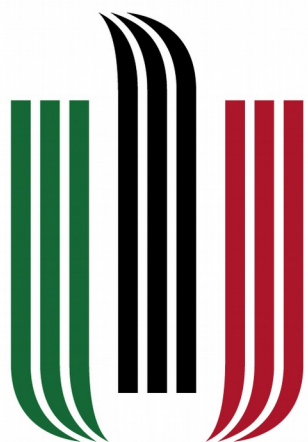




AGH



Podstawy Informatyki – *instrukcje warunkowe pętle, wejście - wyjście*

05.10.218

dr inż. Michał Rad



Plan:

1. Przestrzenie nazw
2. Instrukcje sterujące
 - a) If
 - b) :?
 - c) switch - case
3. Pętle
 - a) do
 - b) while
 - c) for
4. Enum
5. Wejście - wyjście
- 6.

Przestrzenie nazw



- » Pozwalają grupować zmienne pod określoną nazwą
- » Dzielią zakres globalny na mniejsze, nazwane podzakresy

```
namespace moje_zmienne  
{  
    //deklaracja zmiennych  
};
```

Przestrzenie nazw - przykład

```
namespace parameter
{
    int i;
    float s;
};
```

```
...
parameter::i=5;
```

Instrukcja sterująca „if”

- » Uzależnia wykonanie kodu od warunku logicznego
- » Składnia:

```
if( warunek ){  
    //wykonaj jeśli warunek jest prawdą  
} else {  
    //wykonaj jeśli warunek jest fałszem  
    };
```

Przykład



```
if (delta>0) {  
    x1= - b - sqrt(delta) / (2*a);  
    x2= - b + sqrt(delta) / (2*a);  
} else {  
    cout<<"Nie ma pierwiastków"<<endl;  
};
```

Operator ? :



- » Działa podobnie do „if” ale jest operatorem a nie instrukcją – można używać w wyrażeniach

```
warunek ? a_jeśli_tak : b_jeśli_nie;
```

```
wiekszy = a>b?a:b
```

Instrukcja switch - case

```
switch    (wyrażenie)
{
    case wartość1:
        instrukcje1;
        break; //break jest opcjonalny
    case wartość2:
        instrukcje1;
        break;
    default:
        instrukcje_wykonywane_domyślnie;
}
```



Przykład



```
#include <iostream>
using namespace std;
int main()
{
    int a;

    cout<<"Podaj a"<<endl;
    cin>>a;

    switch (a)
    {
        case 1:
            cout<<"Jeden to fajna liczba"<<endl;
        case 2:
            cout<<"Dwa - rowniez dobre"<<endl;
        case 4:
            cout<<"Cztery to Dwa i Dwa"<<endl;
        default:
            cout<<"Niech bedzie "<<a<<endl;
    }
};
```

Pętle



- » Pętle służą do powtarzania określonej części programu
- » Razem z instrukcją warunkową dają możliwość sterowania wykonaniem

Typy pętli:

» „do – while” powtarza się aż warunek przestanie być prawdziwy:

```
do {  
    //kod który będzie powtarzany  
}while(warunek) ;
```

Przykład

```
do {  
a++;  
cout<<a<<endl;  
}while (a<10) ;
```

Typy pętli:

» „while” – warunek sprawdzany jest na początku, w odróżnieniu od poprzedniego rodzaju pętli „do – while”

```
while (a<10) {  
a++;  
cout<<a<<endl;  
};
```

Typy pętli:

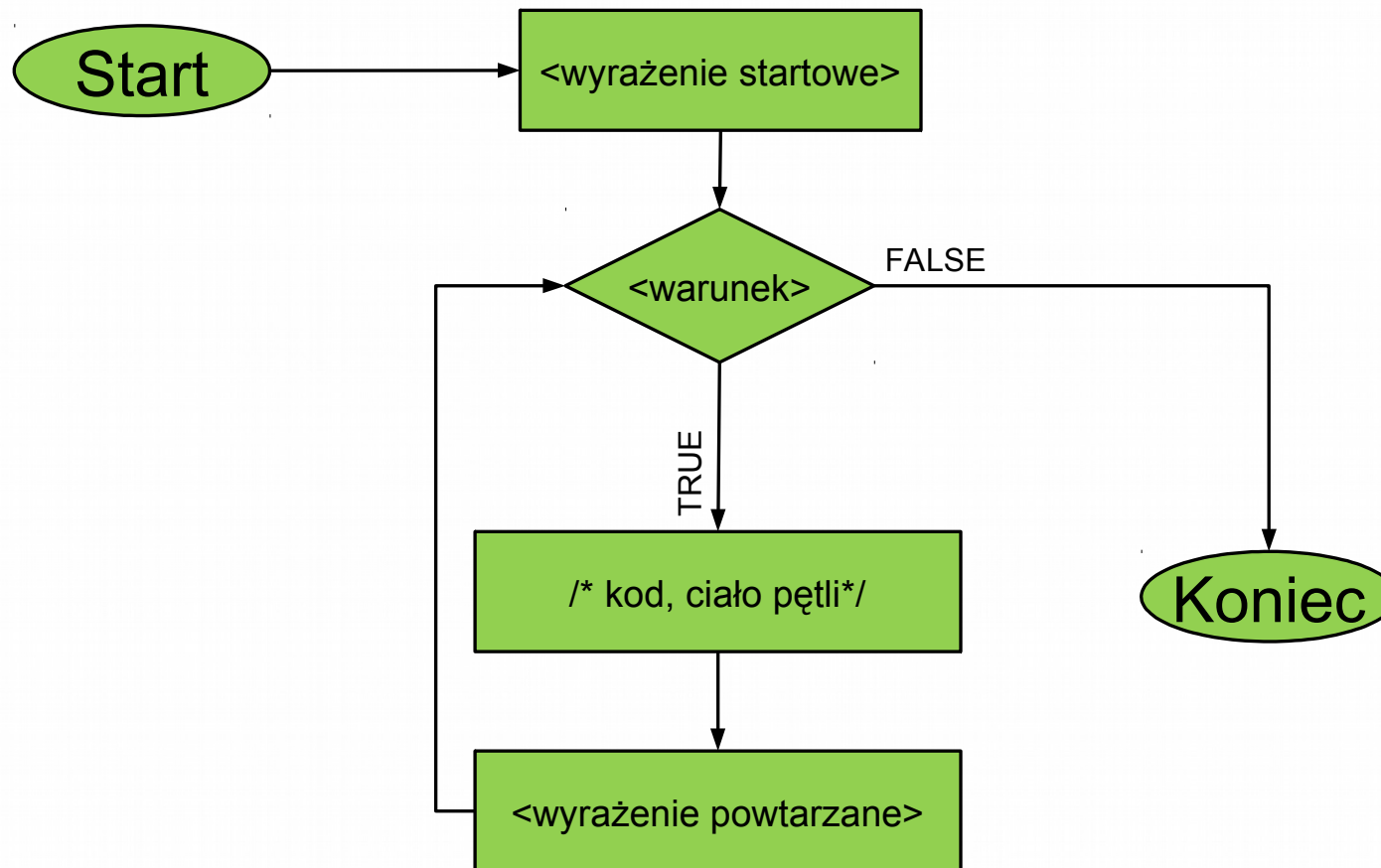


» Pętla „for”

```
> for (wyr_startowe; warunek; wyr_powtazane)
{    // ciało petli
    };
```

```
    for (a=1; a<10; a++) {
cout<<a<<endl;
    };
```

Diagram blokowy

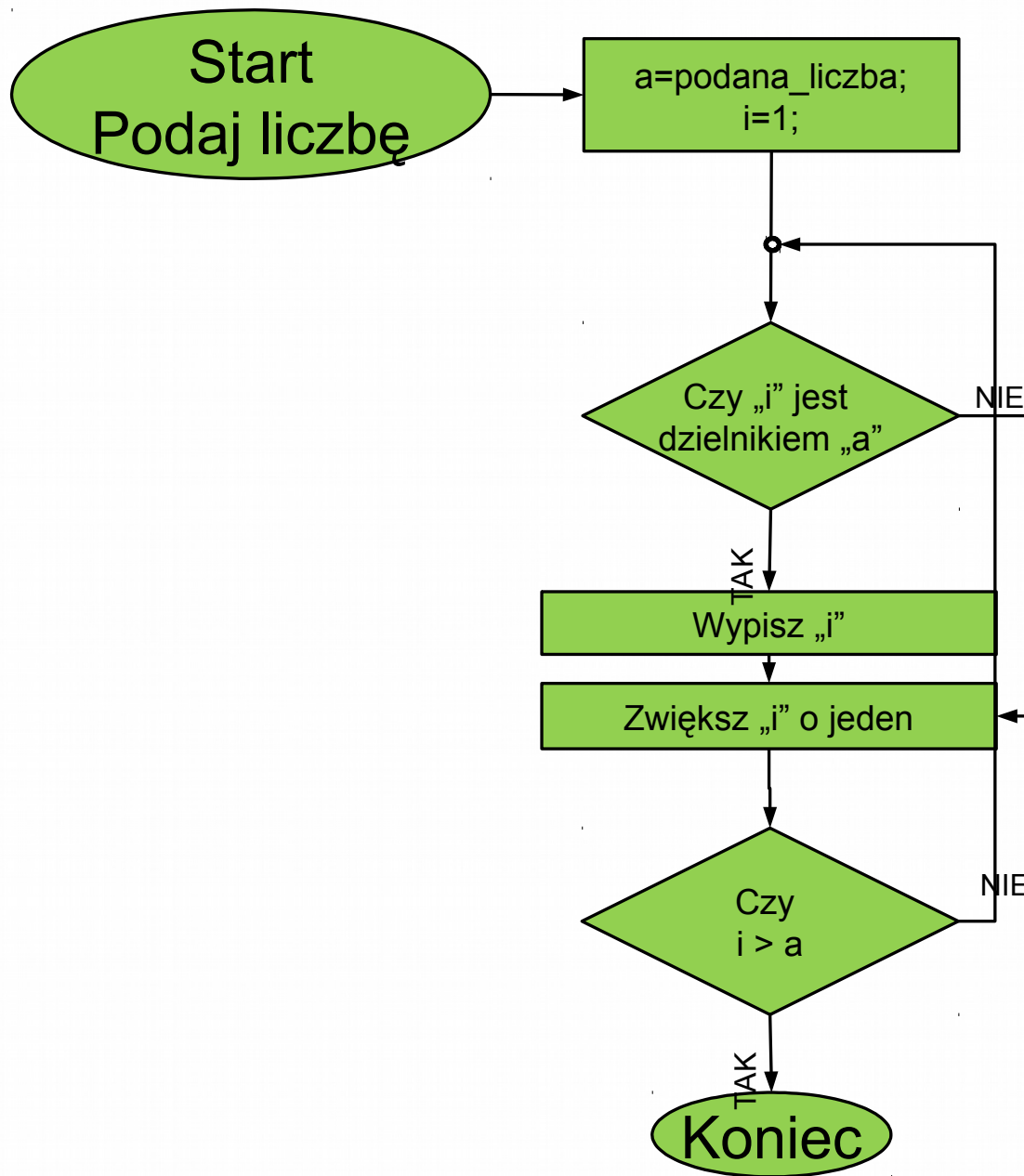


Instrukcje „break” i „continue”

- » „break” przerywa wykonanie bloku programu
- » „continue” przerywa aktualnie wykonywany blok i przechodzi do sprawdzenia warunku

Przykład

- » Znajdowanie wszystkich dzielników podanej liczby
- » Znajdowanie liczb pierwszych



Typ wyliczeniowy enum

```
enum <nazwa>
{
    <tag0> = <wartosc #0>,
    <tag1> = <wartosc #1>,
    <tag2> = <wartosc #2>,
    ...,    // ...
    <tag_n> = <wartosc #n>
};
```

Operacje wejścia - wyjścia

- » W języku C++ mamy możliwość używania dość wygodnego sposobu sterowania wyjściem i wejściem – za pomocą tzw. strumieni
- » Aby korzystać ze strumieni należy załączyć bibliotekę `<iostream>`
- » Wygodnie jest na wstępie programu zadeklarować używanie przestrzeni nazw „std”

Operacje wejścia - wyjścia

- » Domyślnym wyjściem jest konsola (ekran)
- » Domyślnym wejściem jest klawiatura
- » Ale wejściem i wyjściem może być również plik bądź inne urządzenia

Przykład



```
#include <iostream>
using namespace std;
int main() {
    int a;
    cout<<"Podaj wartość a: "<<endl;
    cin>>a;
    cout<<"a^2 = "<<a*a;
}
```

Manipulatory dla strumieni

- Manipulator
- boolalpha
- noboolalpha
- fixed
- scientific
- dec
- hex
- oct
- endl
- Opis
- Typ bool „tekstowy”
- Typ bool „numeryczny”
- Używaj stałej notacji (związane z *precision*)
- Używaj notacji naukowej
- Podstawa dziesiętna
- Podstawa szesnastkowa
- Podstawa ósemkowa
- Koniec lini

Podejście z języka C



- » Funkcje printf() oraz scanf()
- » Znajdują się one w bibliotece <stdio.h>
- » Są też funkcje podobne, do zapisu i odczytu z pliku
- » fprintf(), fscanf()

Przykład

```
#include <stdio.h>

int main()
{
    int zmm1=13;
    float zmm2=2.1213442345;
    printf("Komunikat \n");
    printf("Wartosc int %d \n", zmm1);
    printf("Wartosc float: %6.2f \n", zmm2);
}
```