

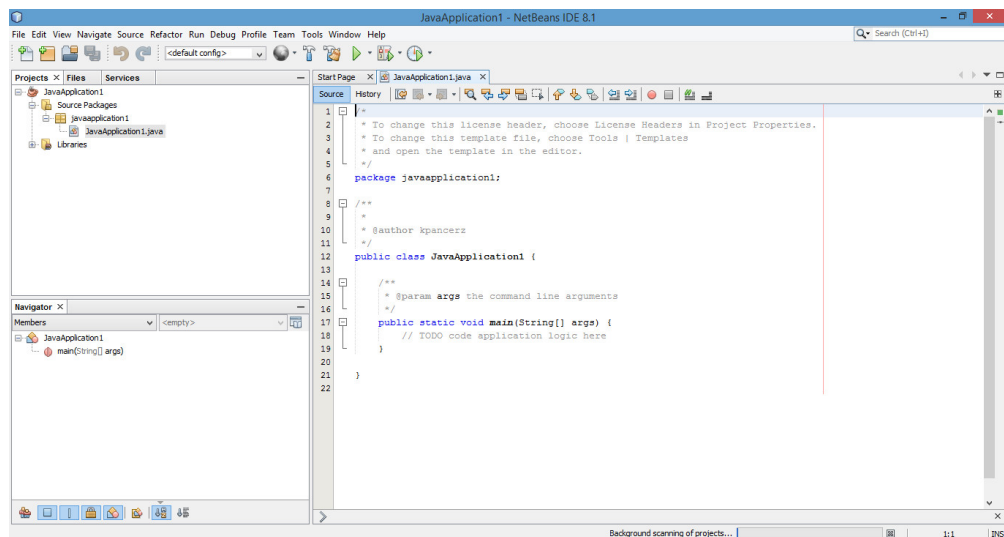
Ćwiczenie 1

Podstawy języka Java. Instrukcje warunkowe

1 Wprowadzenie

1.1 Środowisko programistyczne NetBeans

<https://netbeans.org/>



1.2 Dokumentacja języka Java

<https://docs.oracle.com/javase/8/docs/api/>

1.3 Podstawowa struktura programu

```
public class Glowna
{
    public static void main(String args[])
    {
        // ... instrukcje do wykonania ...
    }
}
```

1.4 Komentarze

```
/*
to jest komentarz blokowy, który może
```

```
zajmować wiele linii
*/
```

```
// to jest komentarz do końca linii
```

1.5 Deklaracja zmiennej

```
typ nazwa;
```

Przykład:

```
int a;
```

1.6 Deklaracja stałej

```
final typ nazwa=wartość;
```

Przykład:

```
final double pi=3.14;
```

1.7 Typy danych

1.7.1 Typy całkowite

Typ	Zakres	Rozmiar
<i>byte</i>	- 128 ... 127	liczba 8-bitowa
<i>short</i>	- 32768 ... 32767	liczba 16-bitowa
<i>int</i>	$-2^{31} \dots 2^{31} - 1$	liczba 32-bitowa
<i>long</i>	$-2^{63} \dots 2^{63} - 1$	liczba 64-bitowa

1.7.2 Typ logiczny

- *boolean* – typ reprezentujący wartości logiczne *true* oraz *false*.

1.7.3 Typ znakowy

- *char* - znaki zajmujące dwa bajty pamięci, uporządkowane zgodnie z zestawem znaków Unicode.

1.7.4 Typy rzeczywiste

Typ	Rozmiar
<i>float</i>	liczba 4-bajtowa
<i>double</i>	liczba 8-bajtowa

1.7.5 Literały

- Literały liczbowe – ciągi znaków reprezentujące liczby, np.: 123, 123e5, 12.52, -5.2, -23.2e10, -12.45e-5
- Literały znakowe - reprezentują znaki alfanumeryczne, np.: 'A', 'c', '+', '@'
- Literały łańcuchowe – służą do reprezentowania ciągów znaków, np.: "tekst", "komunikat"
- Literały logiczne – reprezentują wartości logiczne: true, false

1.8 Operatory

1.8.1 Operatory arytmetyczne jednoargumentowe

Operator	Działanie
+	identyczność
-	zmiana znaku

1.8.2 Operatory arytmetyczne dwuargumentowe

Operator	Działanie
+	dodawanie
-	odejmowanie
*	mnożenie
/	dzielenie
%	reszta z dzielenia całkowitego
++	inkrementacja
--	dekrementacja

1.8.3 Operatory logiczne jednoargumentowe

Operator	Działanie
!	negacja

1.8.4 Operatory logiczne dwuargumentowe

Operator	Działanie
&&	koniunkcja (iloczyn logiczny)
	alternatywa (suma logiczna)

1.8.5 Operatory relacji

Operator	Działanie
==	równy
!=	różny
<	mniejszy
>	większy
<=	mniejszy lub równy
>=	większy lub równy

1.9 Instrukcja przypisania

zmienna=wyrażenie;

Przykład:

```
int a;
a=5;
```

1.10 Instrukcja złożona

```
{  
instrukcja_1  
instrukcja_2  
...  
instrukcja_n  
}
```

1.11 Wyprowadzanie danych na ekran

Bez przejścia do nowej linii po wyprowadzeniu danych:

```
System.out.print(wyrażenie);
```

Z przejściem do nowej linii po wyprowadzeniu danych:

```
System.out.println(wyrażenie);
```

1.12 Wprowadzanie danych z klawiatury

Utworzenie obiektu klasy Scanner:

```
Scanner s=new Scanner(System.in);
```

Wykorzystanie jednej z metod *nextXXX*, np.:

```
int zmienna;  
zmienna=s.nextInt();
```

lub

```
float zmienna;  
zmienna=s.nextFloat();
```

1.13 Definiowanie metod

```
typ_zwracany metoda(argumenty)  
{  
//...Treść (ciało) metody...  
}
```

Przykład:

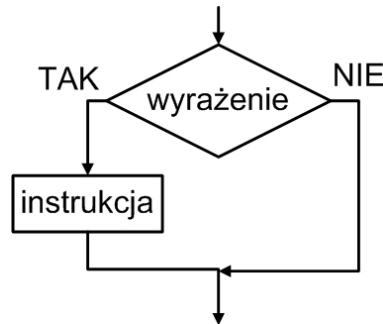
```
int obliczenia(int a, int b)  
{  
return 2*a-b;  
}
```

1.14 Funkcje matematyczne

<https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html>

1.15 Instrukcje warunkowe i wyboru

1.15.1 Instrukcja warunkowa IF

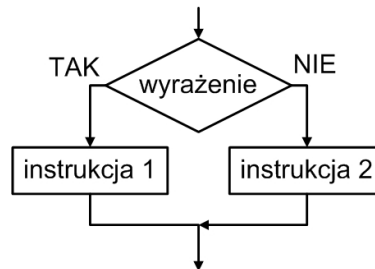


Składnia:

```
if (wyrażenie)
instrukcja
```

- Wartością wyrażenia powinna być wartość logiczna *true* albo *false*.
- Jeśli wartością wyrażenia jest *true*, to instrukcja zostaje wykonana i sterowanie przenoszone jest do kolejnych instrukcji po instrukcji warunkowej.
- Jeśli wartością wyrażenia jest *false*, to instrukcja nie zostaje wykonana, a sterowanie przenoszone jest bezpośrednio do kolejnych instrukcji po instrukcji warunkowej.
- Instrukcja może być instrukcją złożoną.

1.15.2 Instrukcja warunkowa IF-ELSE

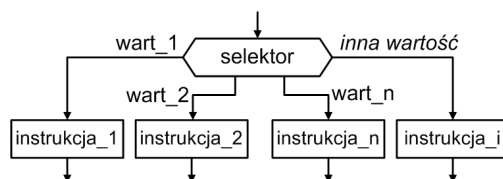


Składnia:

```
if (wyrażenie)
instrukcja1
else
instrukcja2
```

- Wartością wyrażenia powinna być wartość logiczna *true* albo *false*.
- Jeśli wartością wyrażenia jest *true*, to tylko instrukcja1 zostaje wykonana i sterowanie przenoszone jest do kolejnych instrukcji po instrukcji warunkowej.
- Jeśli wartością wyrażenia jest *false*, to tylko instrukcja2 zostaje wykonana i sterowanie przenoszone jest do kolejnych instrukcji po instrukcji warunkowej.
- Instrukcje mogą być instrukcjami złożonymi

1.15.3 Instrukcja wyboru SWITCH



Składnia:

```
switch (selektor)
{
case wart_1: instrukcja_1; break;
case wart_2: instrukcja_2; break;
//...
case wart_n: instrukcja_n; break;
default: instrukcja_i;
}
```

- Selektor jest zmienną lub wyrażeniem typu całkowitego.
- W zależności od wartości wyrażenia wykonywana jest odpowiednia instrukcja.

2 Zadania

2.1

Napisz program, w którym ustawiane są wartości dwóch zmiennych typu całkowitego i wyprowadzane są na ekran wartości ich sumy, różnicy, iloczynu i ilorazu.

2.2

Zmodyfikuj program z zadania 2.1 tak, aby wartości zmiennych wczytywane były z klawiatury.

2.3

Napisz program, w którym ustawiane są wartości dwóch zmiennych typu rzeczywistego i wyprowadzane są na ekran wartości ich sumy, różnicy, iloczynu i ilorazu.

2.4

Zmodyfikuj program z zadania 2.3 tak, aby wartości zmiennych wczytywane były z klawiatury.

2.5

Napisz program, który oblicza wartości następujących wyrażeń dla zmiennej rzeczywistej x podawanej przez użytkownika:

- $x + \frac{1}{x}$,
- $\sin(2x) + \cos^2(x)$,
- $\sqrt{x^2 + 3x - 8}$.

2.6

Napisz program, który oblicza dla zmiennej całkowitej x podawanej przez użytkownika:

- wartość bezwzględną,
- część całkowitą z dzielenia wartości bezwzględnej przez 2,
- resztę z dzielenia wartości bezwzględnej przez 5.

2.7

Napisz metodę rozwiązywania równania kwadratowego o postaci $ax^2 + bx + c = 0$, gdzie a , b i c są stałymi podawanymi jako argumenty metody, ale przyjmij, że stała a będzie zawsze różna od zera. Przetestuj utworzoną metodę.

2.8

Rozszerz metodę napisaną w zad. 2.1 o przypadek kiedy stała a może być równa 0. Przetestuj utworzoną metodę.

2.9

Napisz metodę obliczania podatku dochodowego przy danej podstawie obliczenia podatku.

Podstawa		Podatek
ponad	do	
	85 528	18% podstawy minus kwota 556 zł 02 gr
85 528		14 839 zł 02 gr plus 32% nadwyżki ponad 85 528 zł.

Przetestuj utworzoną metodę.

2.10

Napisz metodę wyznaczania wartości funkcji *signum* dla argumentu rzeczywistego x . Funkcja *signum* określona jest wzorem:

$$\text{signum}(x) = \begin{cases} 1 & \text{dla } x > 0 \\ 0 & \text{dla } x = 0 \\ -1 & \text{dla } x < 0 \end{cases}$$

Przetestuj utworzoną metodę.

2.11

Napisz metody wyznaczania wartości poniższych funkcji dla argumentu rzeczywistego x .

- $a(x) = \begin{cases} 2x & \text{dla } x \geq 0 \\ \frac{x}{2} & \text{dla } x < 0 \end{cases}$
- $b(x) = \begin{cases} \sin(x) & \text{dla } x > 0 \\ x & \text{dla } x = 0 \\ \cos(x) & \text{dla } x < 0 \end{cases}$

$$\bullet c(x) = \begin{cases} \sqrt{x+1} & \text{dla } x > 0 \\ 0 & \text{dla } x = 0 \\ \sqrt{|x-1|} & \text{dla } x < 0 \end{cases}$$

Przetestuj utworzone metody.

2.12

Napisz metodę sprawdzającą podzielność podanej liczby naturalnej x przez podaną liczbę naturalną y . Przetestuj utworzoną metodę.

2.13

Napisz metodę zamieniającą podaną ocenę w postaci cyfrowej na ocenę w postaci słownej:

- 5 → bardzo dobry
- 4 → dobry
- 3 → dostateczny
- 2 → niedostateczny

Przetestuj utworzoną metodę.

2.14

Napisz metodę, która dla podanego miesiąca w postaci liczbowej (tj. 1 - styczeń, 2 - luty, itd.) zwracała będzie liczbę dni w tym miesiącu. Przetestuj utworzoną metodę.