

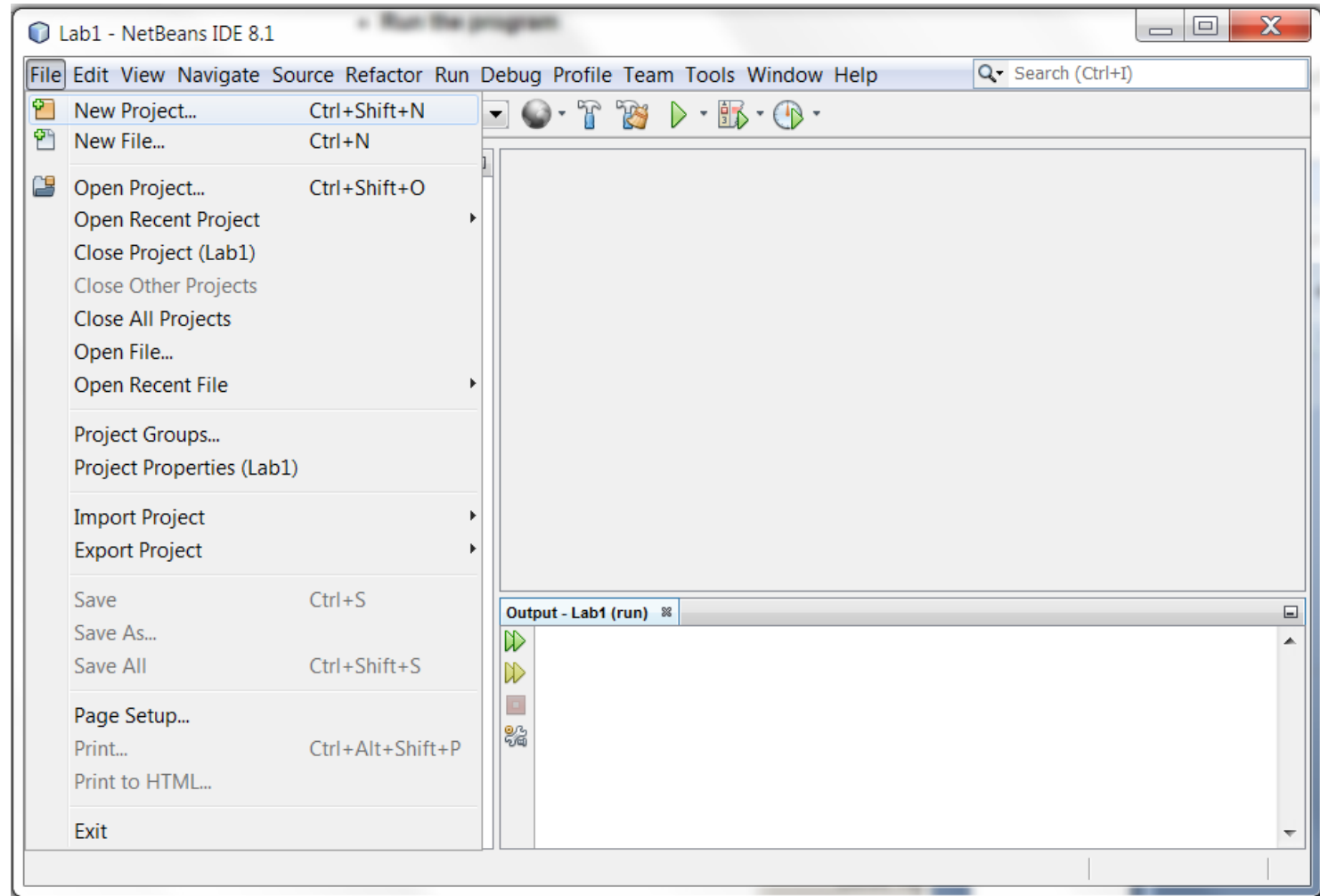
Języki i metody programowania Java

Lab1

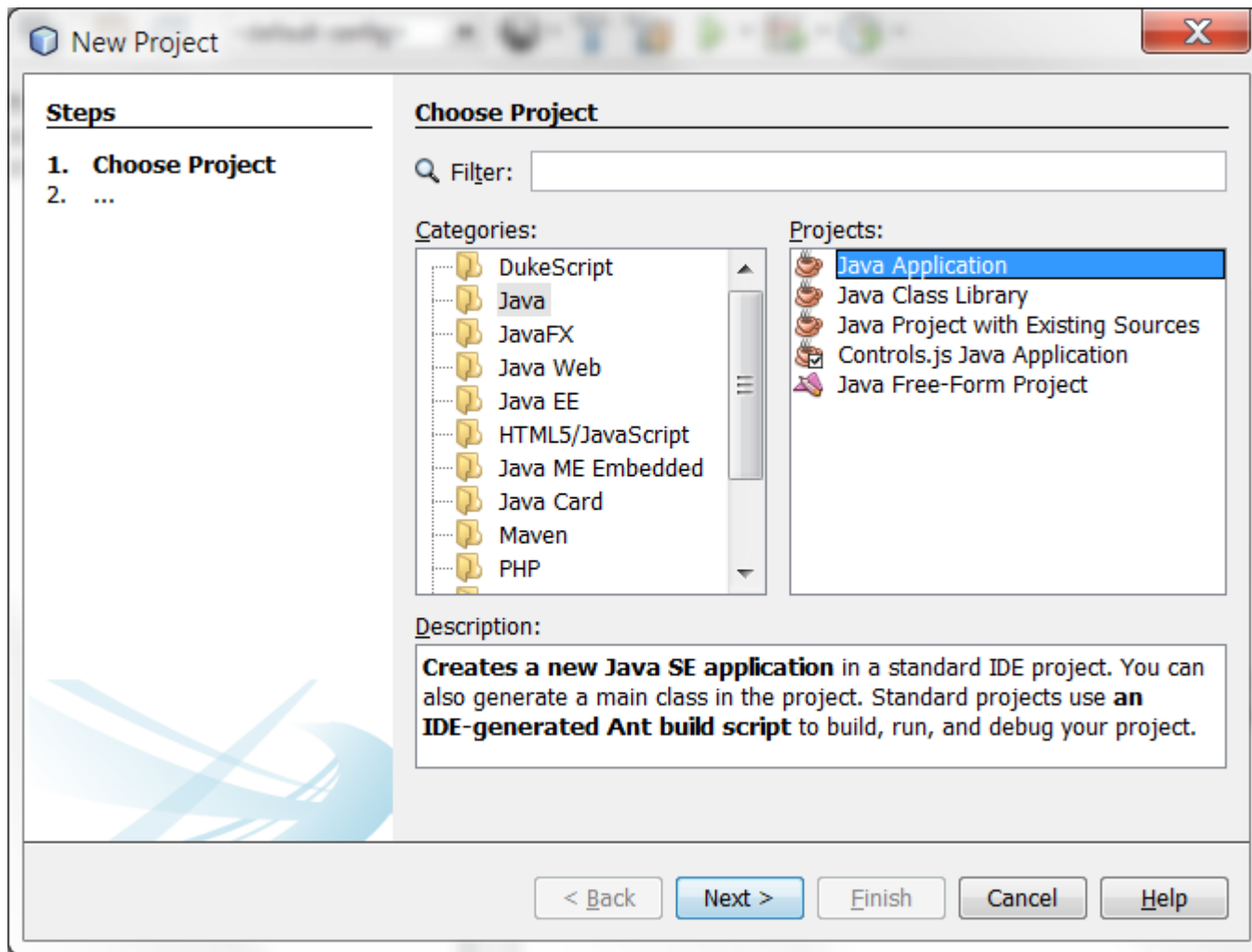
<https://docs.oracle.com/javase/tutorial/>

Zofia Kruczkiewicz

Zadanie 1 . Wykonanie projektu Java SE w środowisku **Netbeans-File/New Project**



W formularzu **New Project** należy wybrać **Java/Java Application** i następnie kliknąć na **Next**



W kolejnym formularzu wpisać nazwę projektu w polu **Project Name**, a w polu **Project Location** wybrać położenie projektu. W polu **Create Main Class** pojawia się informacja, że główna klasa projektu nazywa się tak jak projekt i jest umieszczona w pakiecie o nazwie wywodzącej się z nazwy projektu, napisanej małymi literami – stąd nazwa pakietowa klasy:

witaj_se.Witaj_SE

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: Witaj_SE

Project Location: C:\Studia\Szkola\Java\lab1

Project Folder: C:\Studia\Szkola\Java\lab1\Witaj_SE

Use Dedicated Folder for Storing Libraries

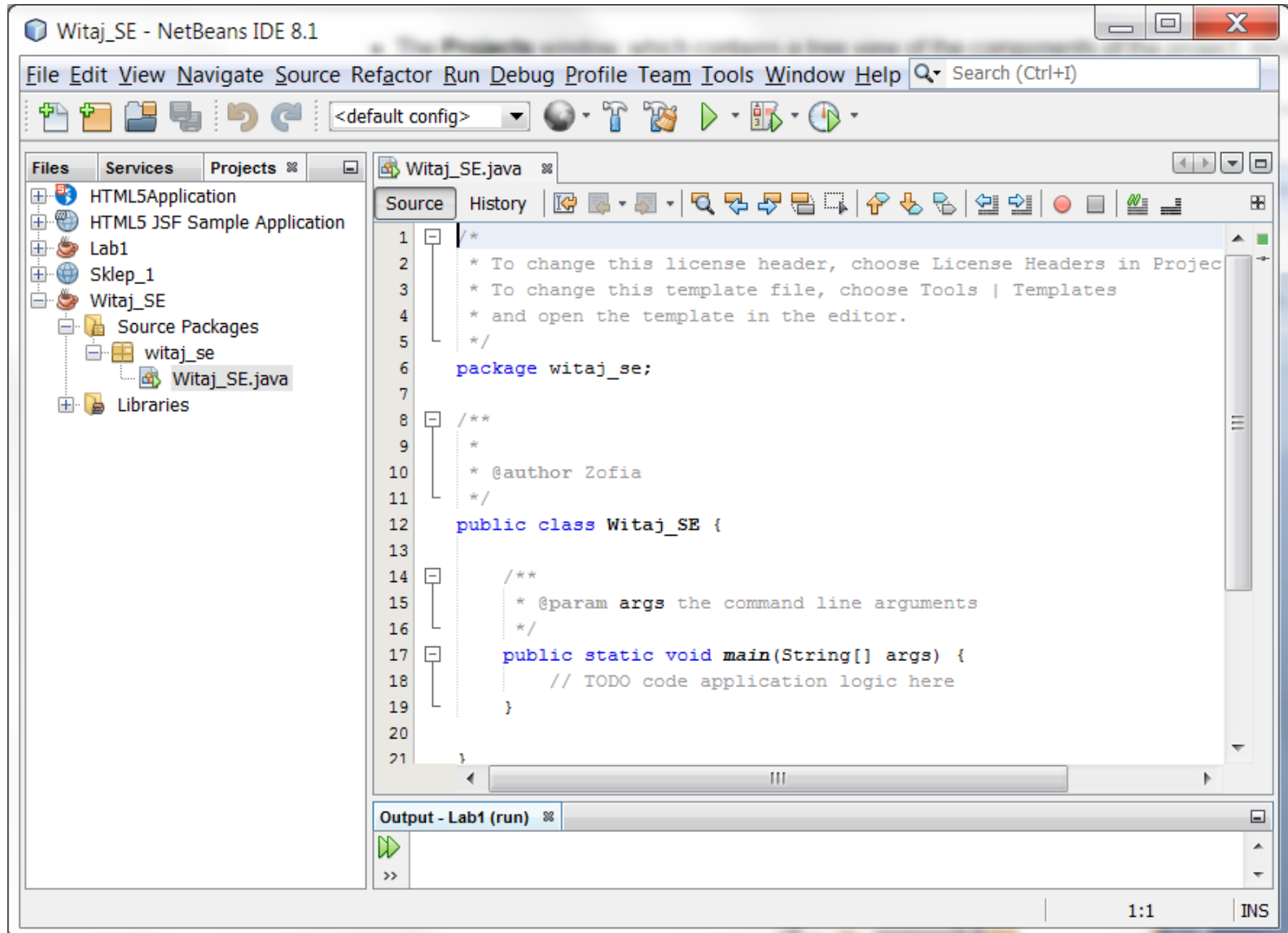
Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

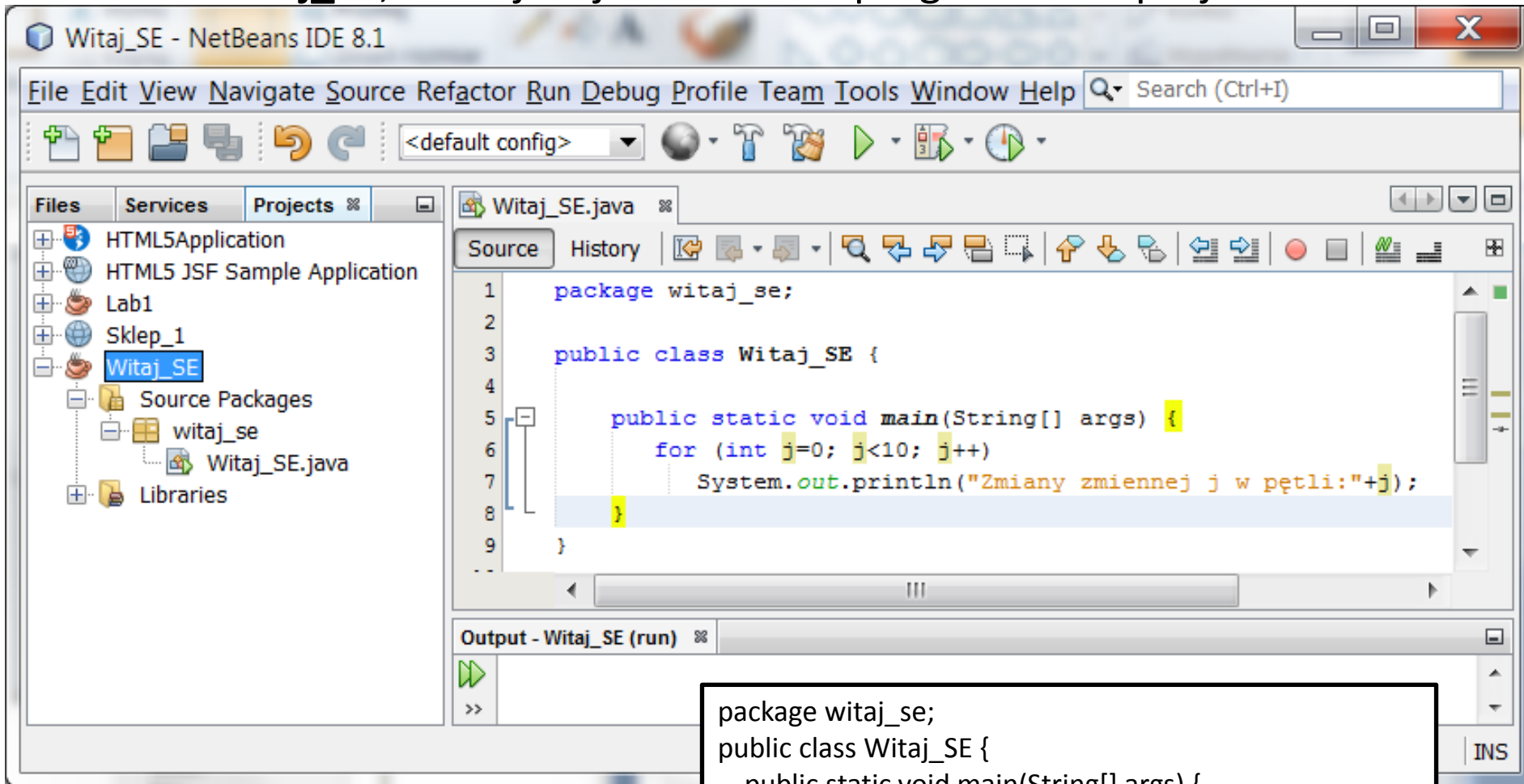
Create Main Class witaj_se.Witaj_SE

< Back Next > **Finish** Cancel Help

Widok głównej klasy w utworzonym projekcie

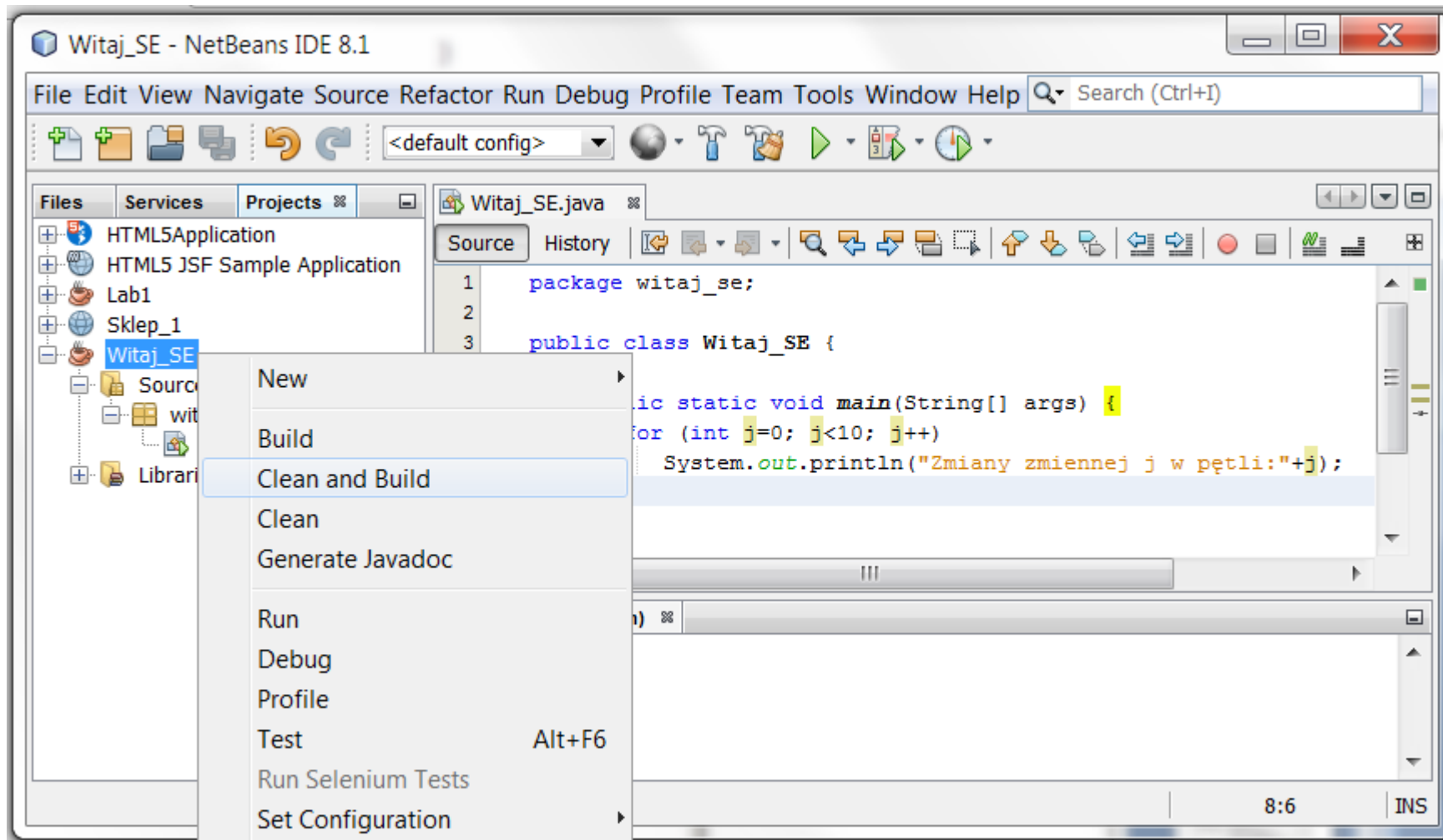


Należy wpisać prosty program typu pętla drukująca na ekranie wartość zmiennej sterującej pętli. Pętlę należy umieścić w metodzie **main** klasy **Witaj_SE**, która jest jednocześnie programem w projekcie.

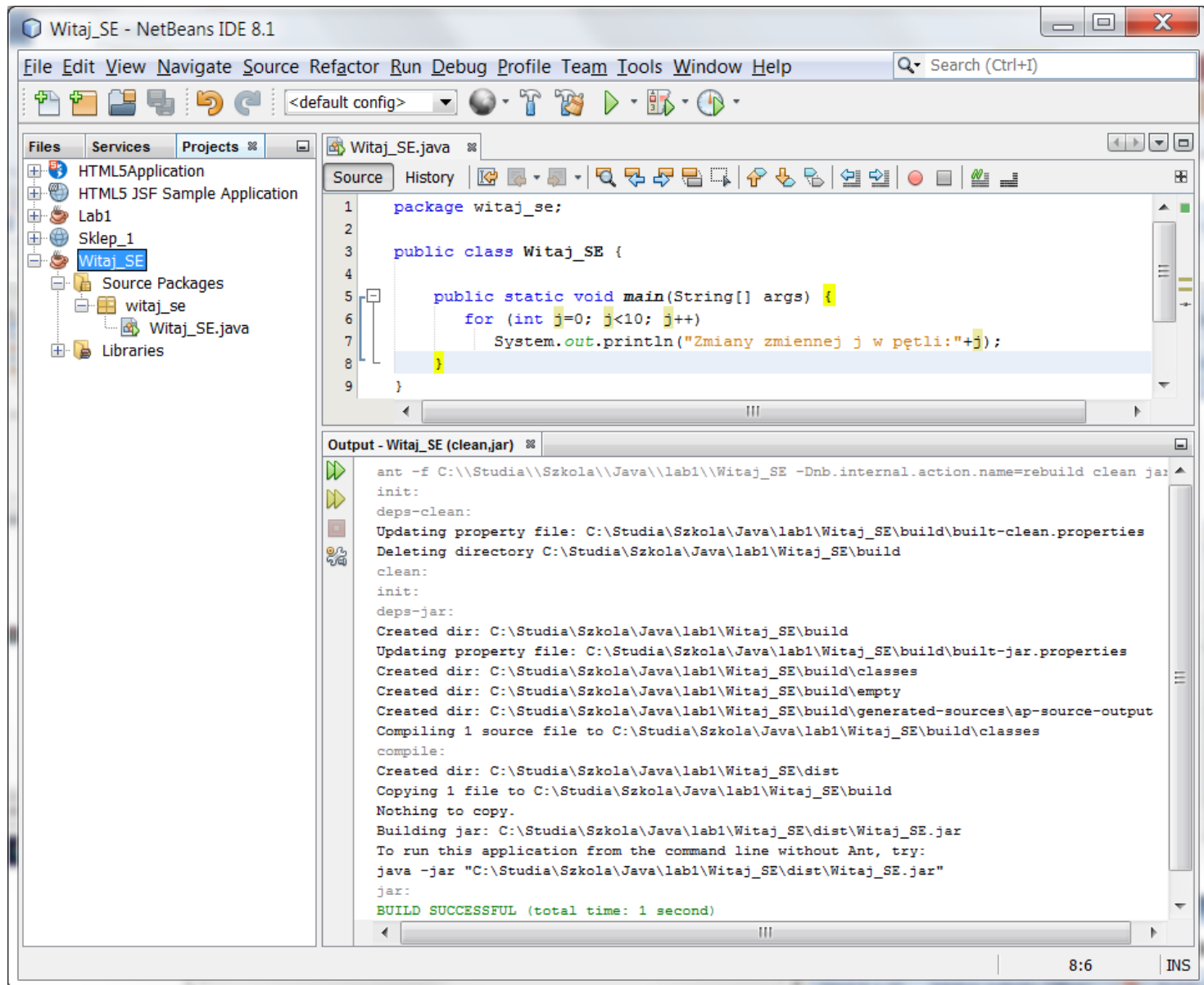


```
package witaj_se;
public class Witaj_SE {
    public static void main(String[] args) {
        for (int j = 0; j < 10; j++)
            System.out.println("Zmiany zmiennej j w pętli:"+j);
    }
}
```

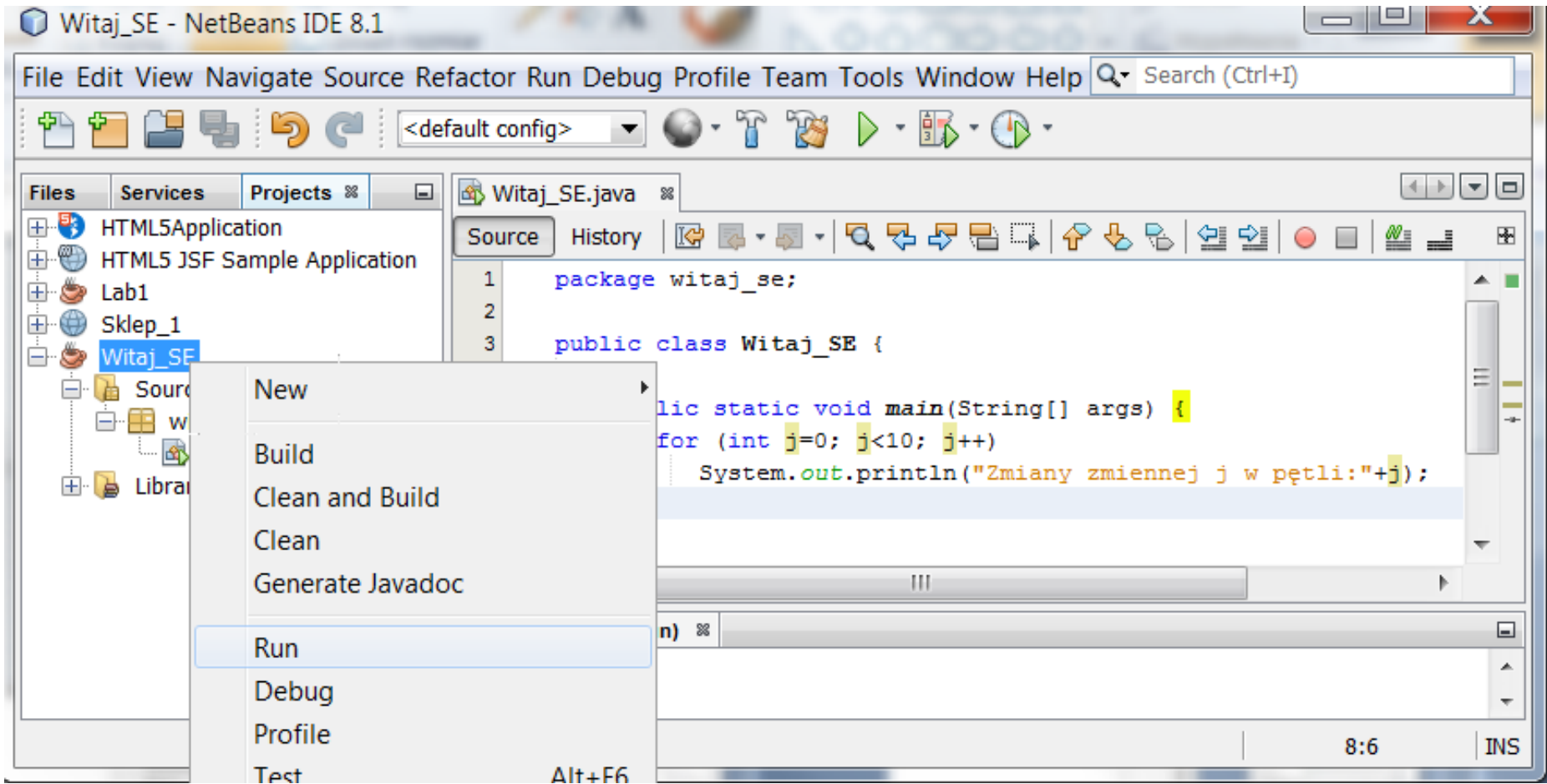
Program zawierający klasę z metoda **main** należy uruchomić w następujący sposób: po kliknięciu prawym klawiszem na nazwę projektu w okienku **Projects**, należy wybrać pozycję **Build** lub **Clean and Build** po wprowadzeniu zmian w projekcie po ostatniej kompilacji.



Wynik przebiegu kompilacji jest prezentowany w okienku zakładki **Output**



Po kompilacji należy wybrać pozycję **Run**, podobnie jak pozycje **Build** lub **Clean and Build**



Wynik działania programu - metoda **System.out.println** wyświetla w trybie konsolowym informację na konsoli w zakładce **Output**.

The screenshot displays the NetBeans IDE 8.1 interface. The main editor window shows the source code for `Witaj_SE.java`:

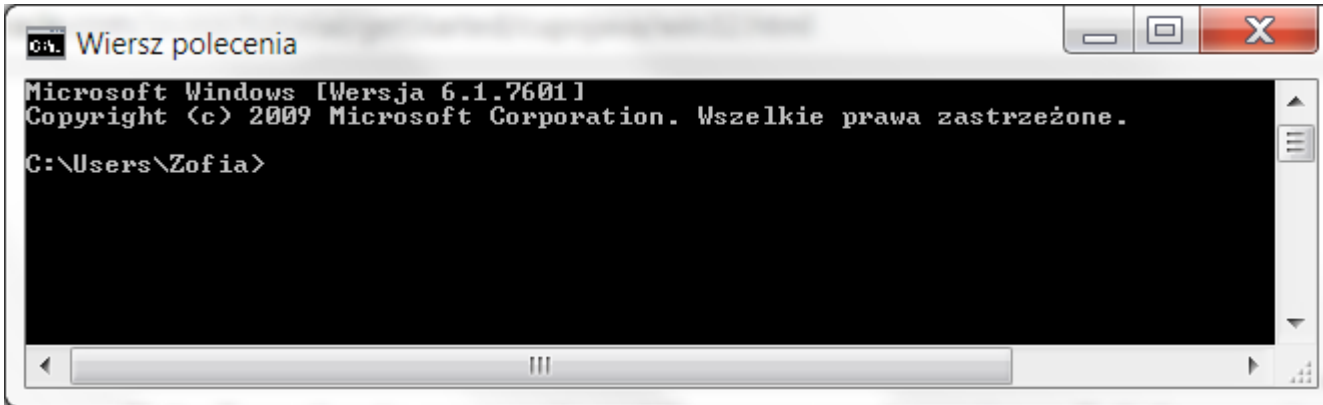
```
1 package witaj_se;
2
3 public class Witaj_SE {
4
5     public static void main(String[] args) {
6         for (int j=0; j<10; j++)
7             System.out.println("Zmiany zmiennej j w pętli:"+j);
8     }
9 }
```

The `main` method contains a `for` loop that iterates from `j=0` to `j<10`, printing the value of `j` in a Polish sentence. The output window, titled "Output - Witaj_SE (run)", shows the following output:

```
run:
Zmiany zmiennej j w pętli:0
Zmiany zmiennej j w pętli:1
Zmiany zmiennej j w pętli:2
Zmiany zmiennej j w pętli:3
Zmiany zmiennej j w pętli:4
Zmiany zmiennej j w pętli:5
Zmiany zmiennej j w pętli:6
Zmiany zmiennej j w pętli:7
Zmiany zmiennej j w pętli:8
Zmiany zmiennej j w pętli:9
BUILD SUCCESSFUL (total time: 0 seconds)
```

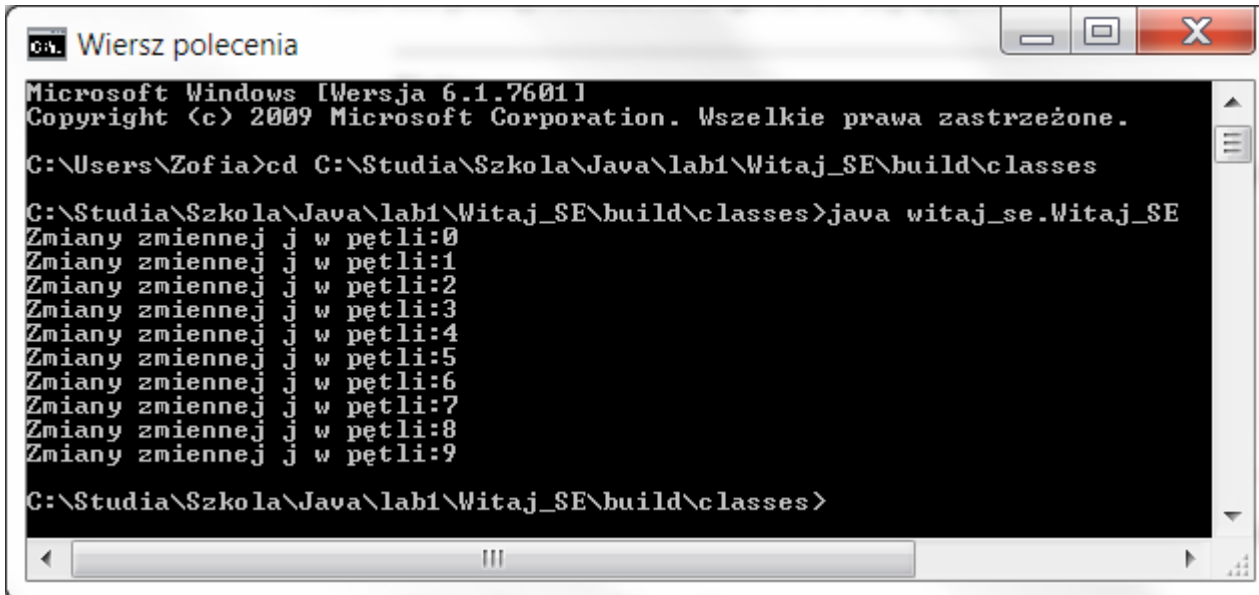
The status bar at the bottom right of the IDE shows the time `8:6` and the keyboard indicator `INS`.

Uruchomienie programu w linii poleceń



```
Wiersz polecenia
Microsoft Windows [Wersja 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\Zofia>
```

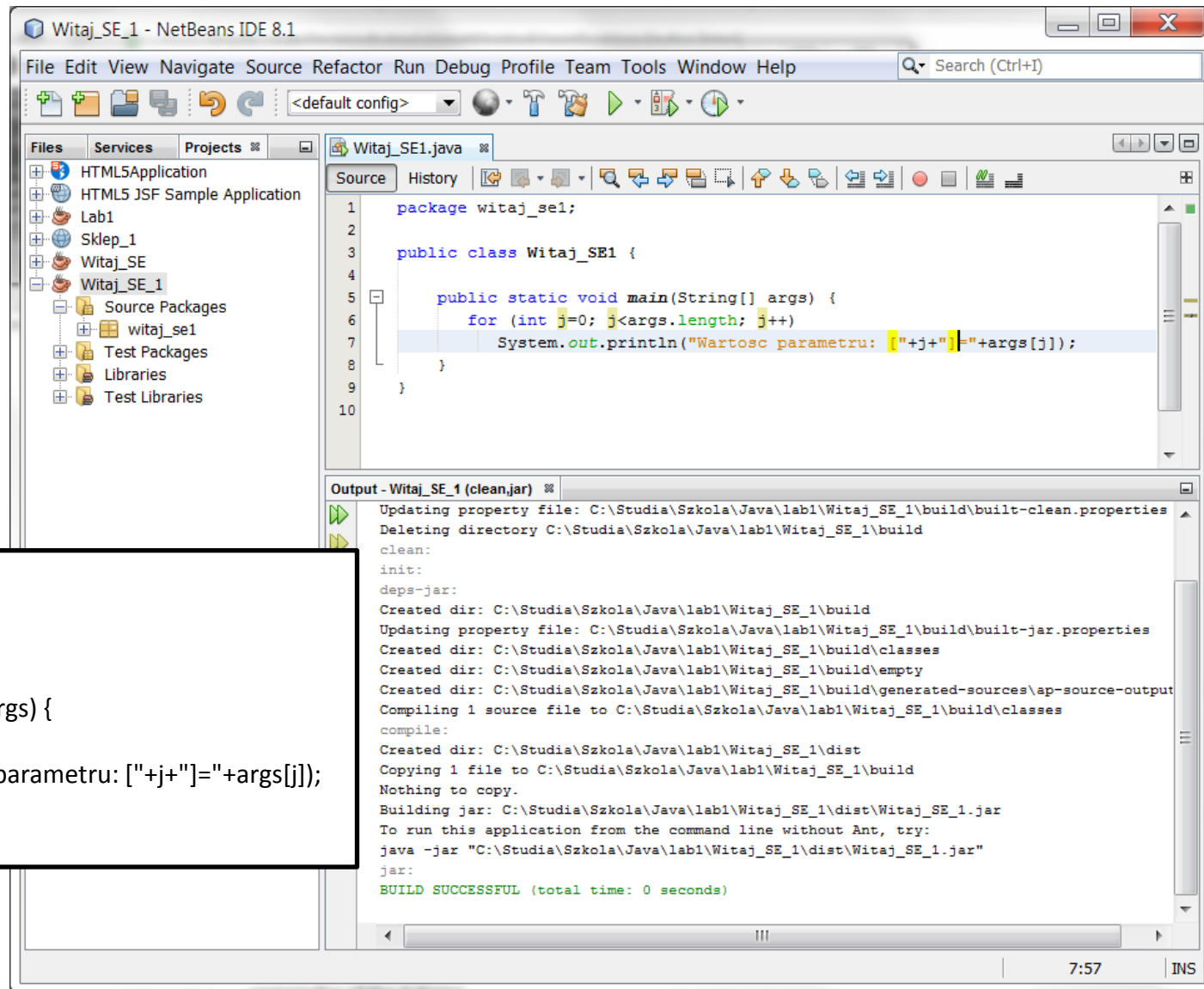


```
Wiersz polecenia
Microsoft Windows [Wersja 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\Zofia>cd C:\Studia\Szkola\Java\lab1\Witaj_SE\build\classes
C:\Studia\Szkola\Java\lab1\Witaj_SE\build\classes>java witaj_se.Witaj_SE
Zmiany zmiennej j w petli:0
Zmiany zmiennej j w petli:1
Zmiany zmiennej j w petli:2
Zmiany zmiennej j w petli:3
Zmiany zmiennej j w petli:4
Zmiany zmiennej j w petli:5
Zmiany zmiennej j w petli:6
Zmiany zmiennej j w petli:7
Zmiany zmiennej j w petli:8
Zmiany zmiennej j w petli:9

C:\Studia\Szkola\Java\lab1\Witaj_SE\build\classes>
```

Przekazywanie parametrów do programu z linii poleceń przez parametr **String []args** metody **main**



The screenshot shows the NetBeans IDE 8.1 interface. The main editor displays the source code for `Witaj_SE1.java`. The code defines a package `witaj_se1` and a public class `Witaj_SE1` with a `main` method that iterates over command-line arguments and prints them.

```
1 package witaj_se1;
2
3 public class Witaj_SE1 {
4
5     public static void main(String[] args) {
6         for (int j=0; j<args.length; j++)
7             System.out.println("Wartosc parametru: ["+j+"]="+args[j]);
8     }
9 }
10
```

The Output window shows the successful compilation of the program into a JAR file.

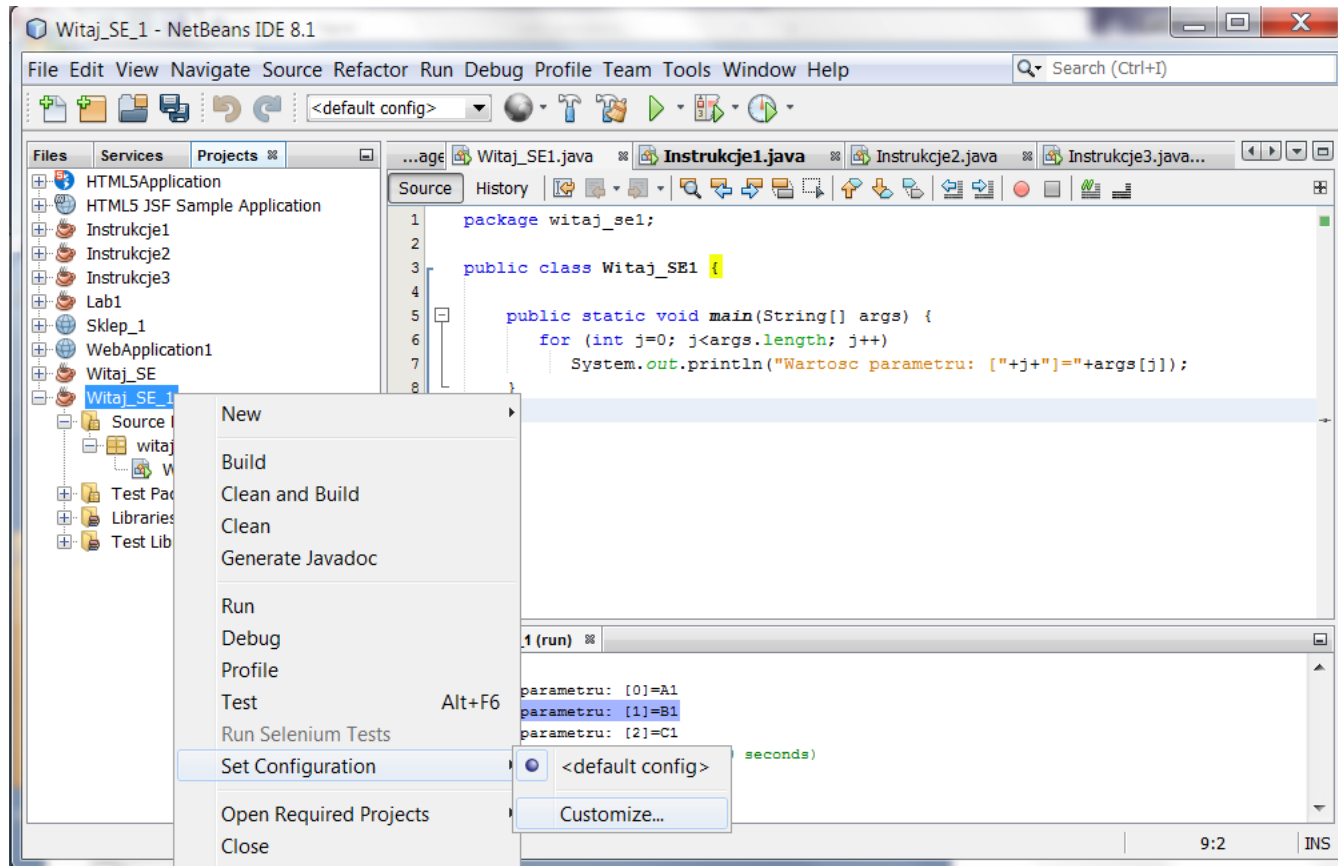
```
Updating property file: C:\Studia\Szkola\Java\lab1\Witaj_SE1\build\build-clean.properties
Deleting directory C:\Studia\Szkola\Java\lab1\Witaj_SE1\build
clean:
init:
deps-jar:
Created dir: C:\Studia\Szkola\Java\lab1\Witaj_SE1\build
Updating property file: C:\Studia\Szkola\Java\lab1\Witaj_SE1\build\build-jar.properties
Created dir: C:\Studia\Szkola\Java\lab1\Witaj_SE1\build\classes
Created dir: C:\Studia\Szkola\Java\lab1\Witaj_SE1\build\empty
Created dir: C:\Studia\Szkola\Java\lab1\Witaj_SE1\build\generated-sources\ap-source-output
Compiling 1 source file to C:\Studia\Szkola\Java\lab1\Witaj_SE1\build\classes
compile:
Created dir: C:\Studia\Szkola\Java\lab1\Witaj_SE1\dist
Copying 1 file to C:\Studia\Szkola\Java\lab1\Witaj_SE1\build
Nothing to copy.
Building jar: C:\Studia\Szkola\Java\lab1\Witaj_SE1\dist\Witaj_SE1.jar
To run this application from the command line without Ant, try:
java -jar "C:\Studia\Szkola\Java\lab1\Witaj_SE1\dist\Witaj_SE1.jar"
jar:
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
package witaj_se1;

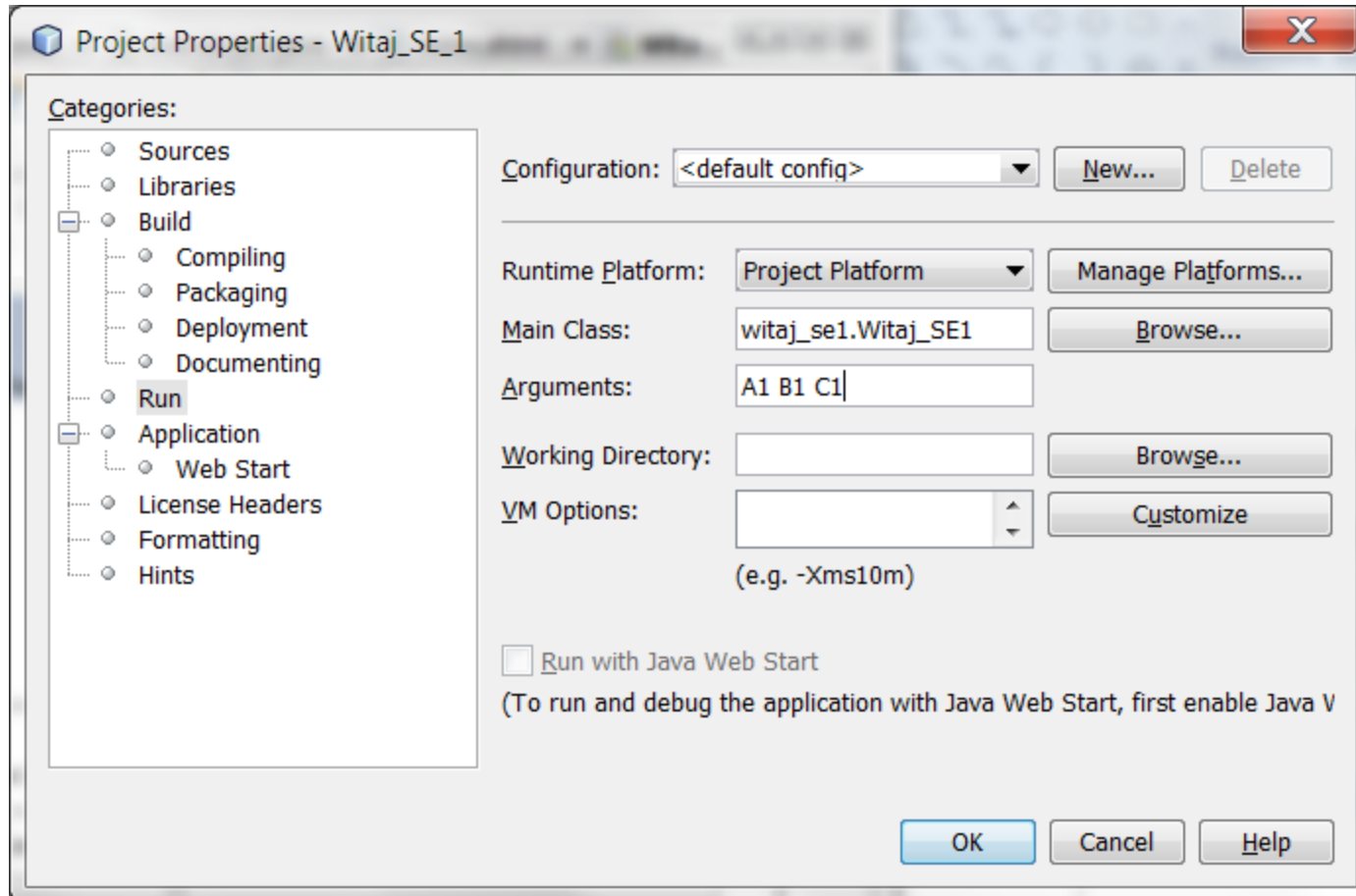
public class Witaj_SE1 {

    public static void main(String[] args) {
        for (int j=0; j<args.length; j++)
            System.out.println("Wartosc parametru: ["+j+"]="+args[j]);
    }
}
```

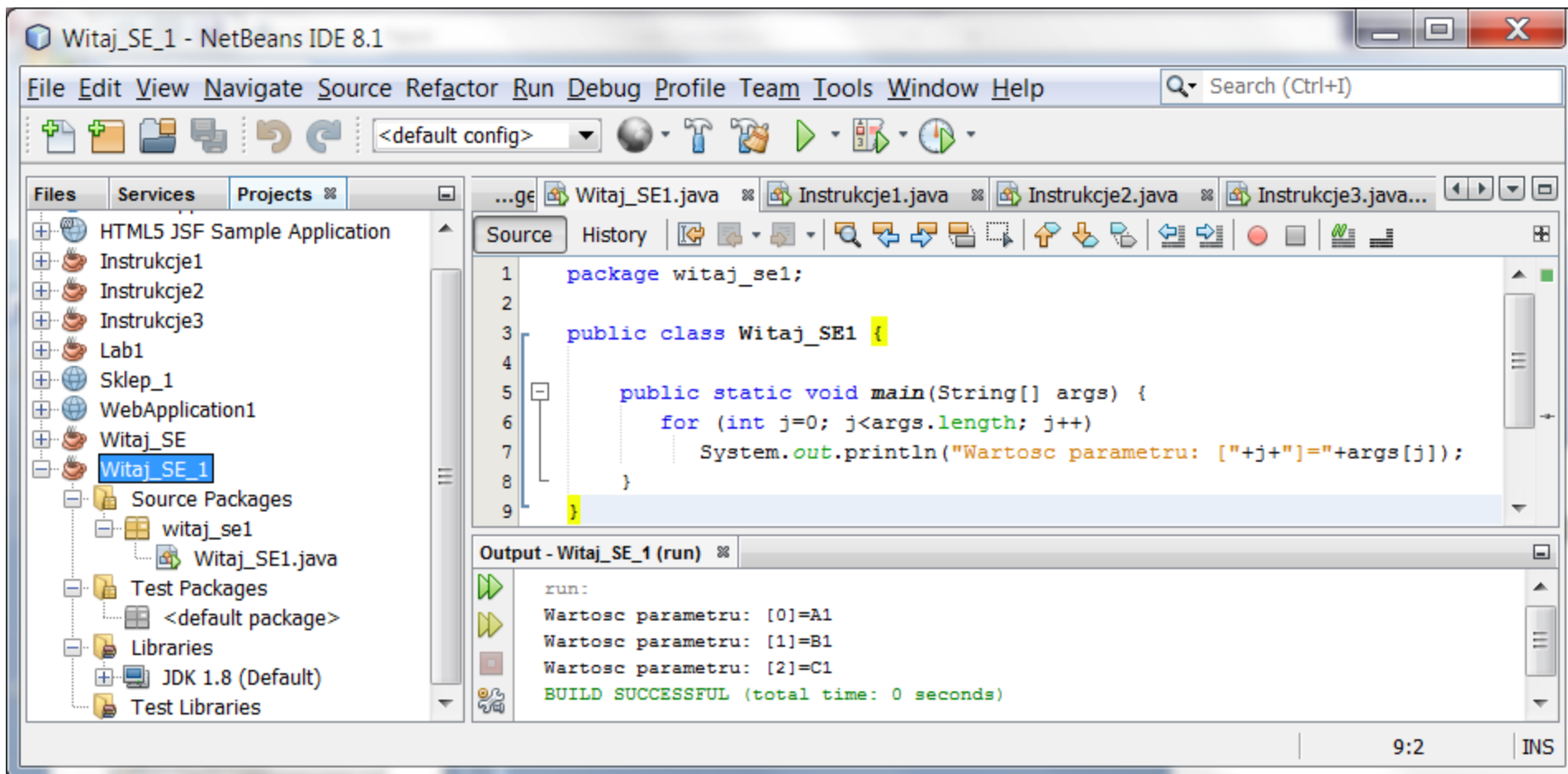
Uruchamianie programu z parametrami w środowisku **NetBeans**
– po kliknięciu prawym klawiszem na nazwę projektu należy
wybrać kolejno pozycje **Set Configuration/Customize**



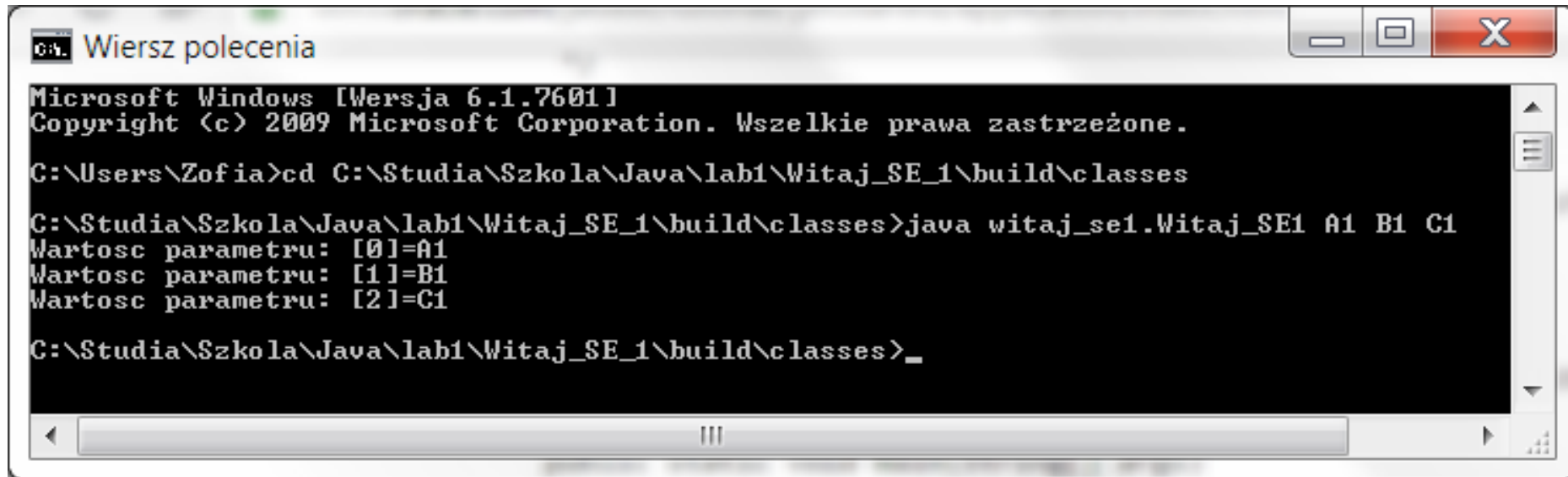
Uruchamianie programu z parametrami w środowisku **NetBeans** – wpisanie do pola **Arguments** dowolnej liczby parametrów i zatwierdzić **OK**



Uruchamianie programu z parametrami w środowisku **NetBeans** – po uruchomieniu programu podane parametry z linii poleceń są wykorzystane w programie



Uruchomienie programu przekazującego parametry do programu z linii poleceń przez parametr **String []args** metody **main** – przetwarzanie jednowymiarowej tablicy obiektów **String**

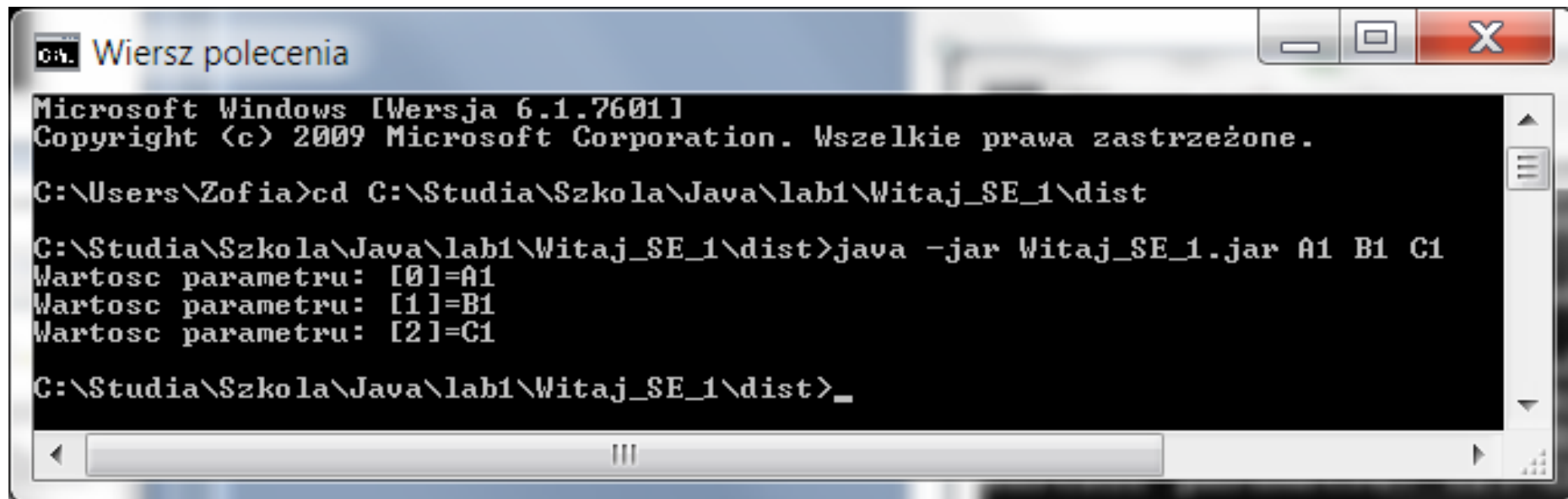


```
Wiersz polecenia
Microsoft Windows [Wersja 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\Zofia>cd C:\Studia\Szkola\Java\lab1\Witaj_SE_1\build\classes

C:\Studia\Szkola\Java\lab1\Witaj_SE_1\build\classes>java witaj_se1.Witaj_SE1 A1 B1 C1
Wartosc parametru: [0]=A1
Wartosc parametru: [1]=B1
Wartosc parametru: [2]=C1

C:\Studia\Szkola\Java\lab1\Witaj_SE_1\build\classes>_
```



```
Wiersz polecenia
Microsoft Windows [Wersja 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\Zofia>cd C:\Studia\Szkola\Java\lab1\Witaj_SE_1\dist

C:\Studia\Szkola\Java\lab1\Witaj_SE_1\dist>java -jar Witaj_SE_1.jar A1 B1 C1
Wartosc parametru: [0]=A1
Wartosc parametru: [1]=B1
Wartosc parametru: [2]=C1

C:\Studia\Szkola\Java\lab1\Witaj_SE_1\dist>_
```


Zadanie 2 – Należy dokonać analizy kodu programu rysującego trójkąt. Należy wykonać program rysujący kwadrat lub prostokąt – dalej pokazano, jak dodać w programie import potrzebnych kodów klas

```
package instrukcje1;
```

```
import javax.swing.JOptionPane;
```

```
public class Instrukcje1 {
```

```
public static void main(String[] args) {
```

```
    int i, j, liczba_wierszy, liczba_spacji, liczba_znakow;
```

```
    String s, rysunek = "";
```

```
    char z;
```

```
    boolean warunek;
```

```
    do {
```

```
        s = JOptionPane.showInputDialog(null,
```

```
        "Podaj liczbę wierszy, czyli wysokość trójkąta zawartą 3<=Liczba wierszy<=10");
```

```
        liczba_wierszy = Integer.parseInt(s);
```

```
        warunek = !(liczba_wierszy >= 3 && liczba_wierszy <= 10);
```

```
    } if (warunek) {
```

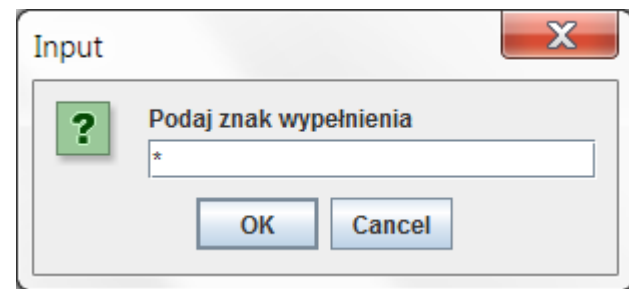
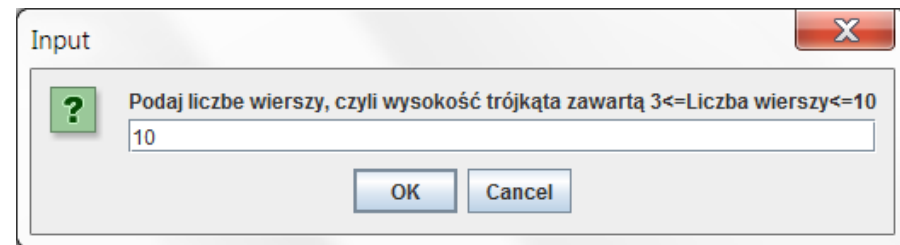
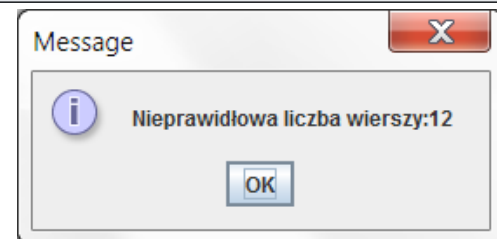
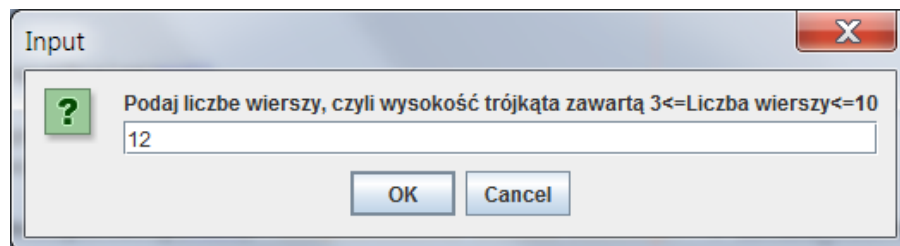
```
        JOptionPane.showMessageDialog(null,
```

```
        "Nieprawidłowa liczba wierszy:" + liczba_wierszy);
```

```
    }
```

```
    } while (warunek);
```

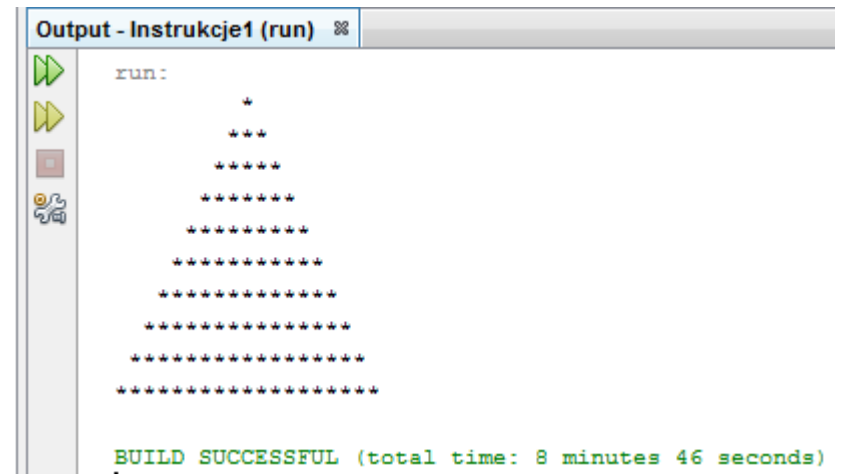
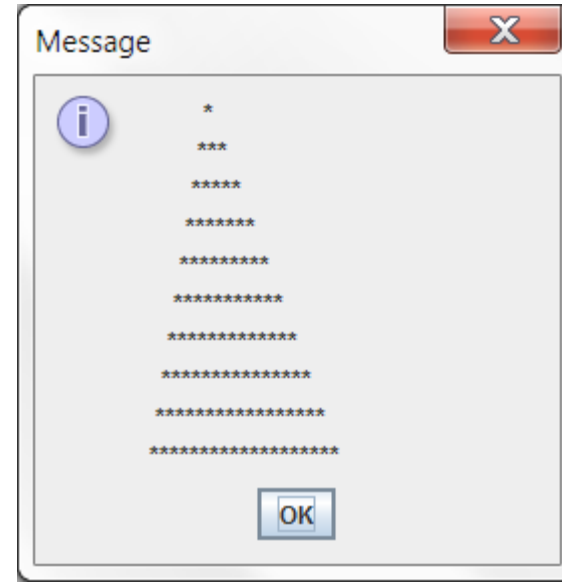
```
    s = JOptionPane.showInputDialog(null, "Podaj znak wypełnienia");
```



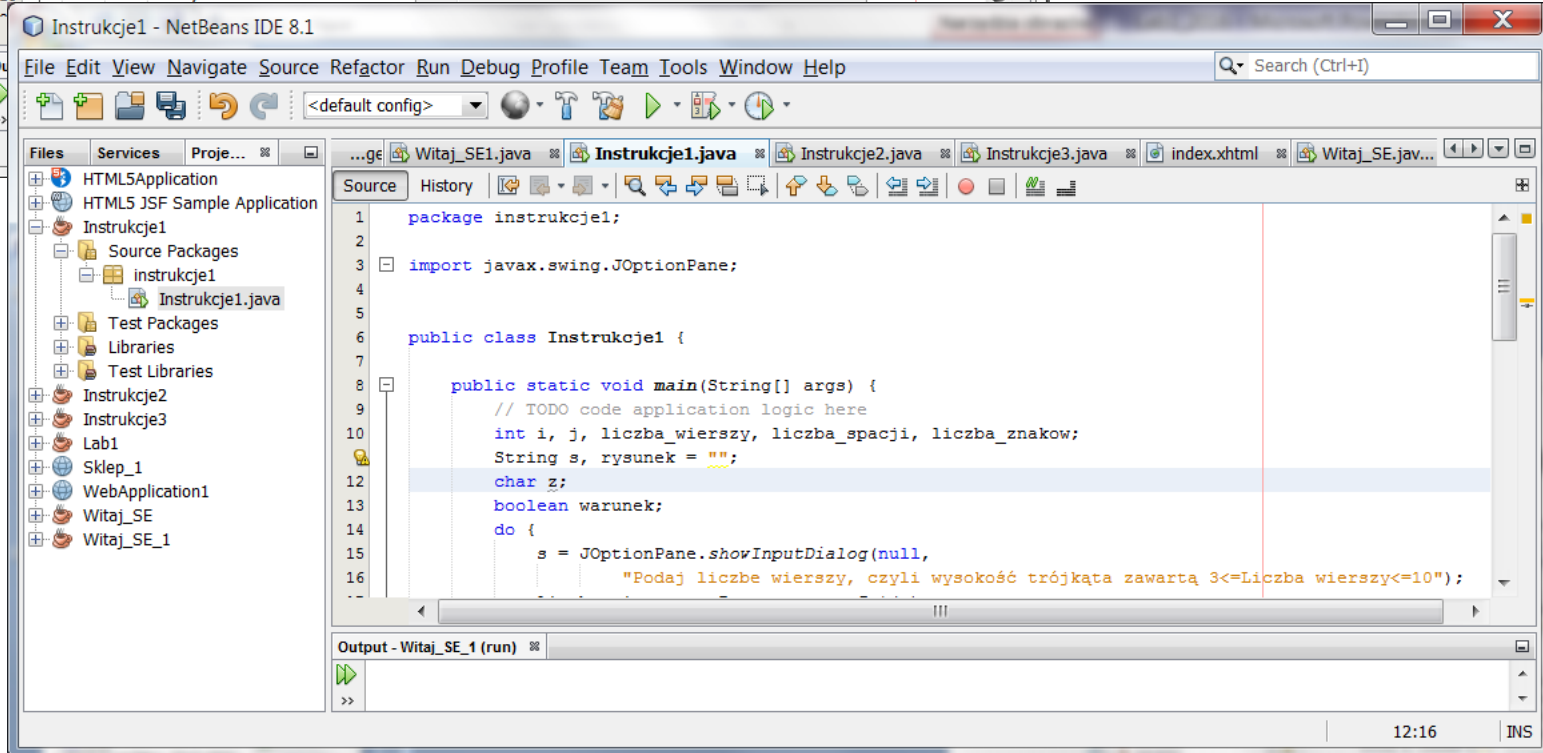
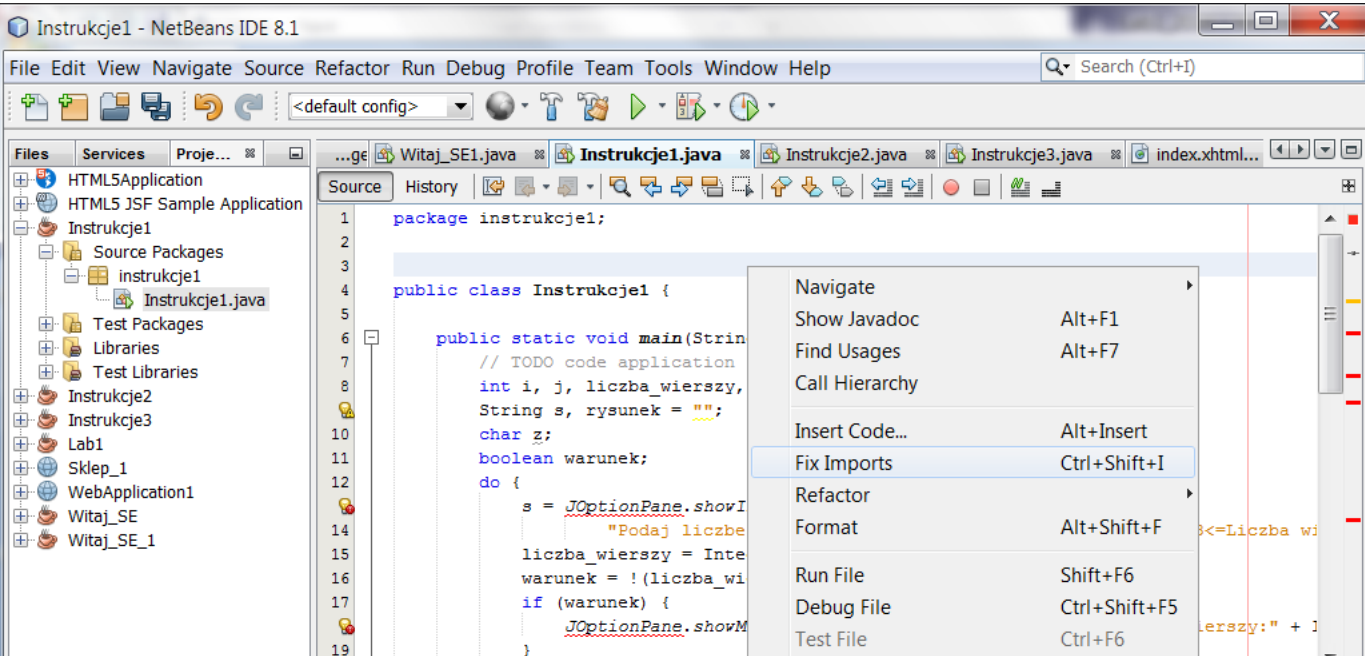
```

//rysowanie
rysunek = "";
for (j = 0; j < liczba_wierszy; j++) {
    liczba_spacji = liczba_wierszy - j - 1;
    for (i = 0; i < liczba_spacji; i++) {
        rysunek += " ";
    }
    liczba_znakow = 2 * j + 1;
    for (i = 0; i < liczba_znakow; i++) {
        rysunek += s;
    }
    rysunek += "\r\n";
}
OptionPane.showMessageDialog(null, rysunek);
System.out.println(rysunek);
System.exit(0);
}
}

```



W przypadku braku dostępu do kodu w programie należy kliknąć na powierzchnię edytora programu prawym klawiszem myszy i wybrać pozycję **Fix Imports**. Po kliknięciu nastąpi import kodu wymaganej klasy



Zadanie 3 – Należy dokonać analizy kodu programu typu kalkulator. Należy dodać operacje odejmowania i mnożenia.

```
package instrukcje2;
```

```
import javax.swing.JOptionPane;
```

```
public class Instrukcje2 {
```

```
    public static void main(String[] args) {
```

```
        char op;
```

```
        float a, b, c;
```

```
        String s, wynik;
```

```
        JOptionPane.showMessageDialog(null, "Prosty kalkulator\n");
```

```
        do {
```

```
            s = JOptionPane.showInputDialog(null,
```

```
                "Podaj jeden z operatorów:\n"
```

```
                + "+ albo d - dodawanie\n"
```

```
                + "/ albo \ - dzielenie\n"
```

```
                + "lub k - koniec programu");
```

```
            op = s.charAt(0);
```

```
            if (op == 'k')
```

```
                break;
```

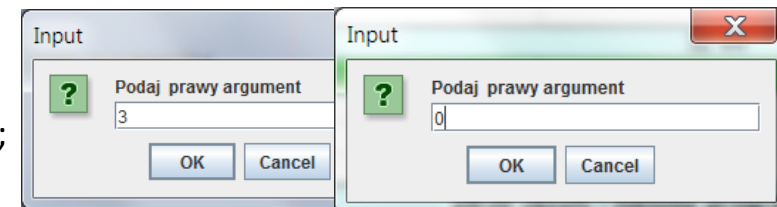
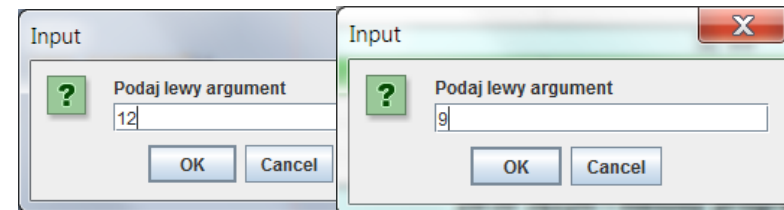
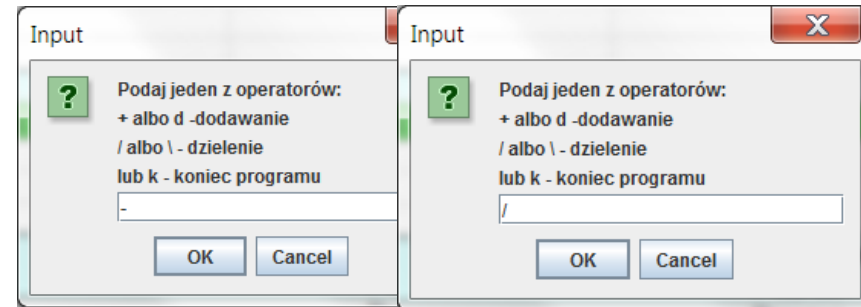
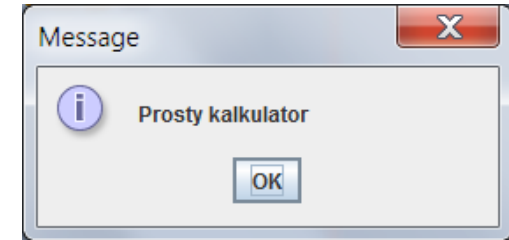
```
                //wyjście z pętli do while
```

```
            s = JOptionPane.showInputDialog(null, "Podaj lewy argument");
```

```
            a = Float.parseFloat(s);
```

```
            s = JOptionPane.showInputDialog(null, "Podaj prawy argument");
```

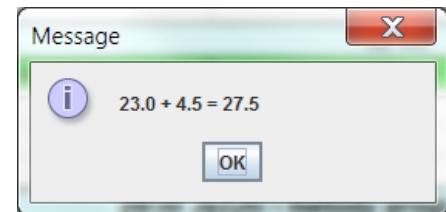
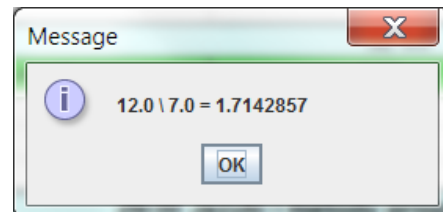
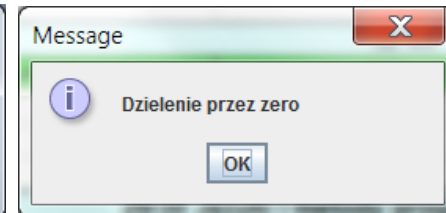
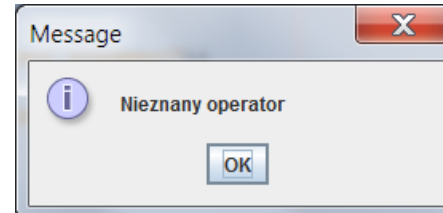
```
            b = Float.parseFloat(s);
```



```

switch (op) {
    case '+':
        //brak instrukcji break powoduje przejście do kolejnego case
    case 'd':
        c = a + b;
        wynik = a + " " + "+" + " " + b + " = " + c;
        break;
        //break powoduje zakończenie instrukcji switch, gdy op='+' lub op='d'
    case '/':
    case '\\':
        if (b != 0) {
            c = a / b;
            wynik = a + " " + op + " " + b + " = " + c;
        } else {
            wynik = "Dzielenie przez zero";
        }
        break;
        //break powoduje zakończenie instrukcji switch, gdy op='\' lub op = '/'
    default:
        wynik = "Nieznany operator";
}
OptionPane.showMessageDialog(null, wynik);
} while (true);
wynik = "Koniec programu";
OptionPane.showMessageDialog(null, wynik);
System.exit(0);
}
}

```



Zadanie 4* - Zadani e dodatkowe – wyznaczenie zestawu podanych banknotów, których wartość jest równa podanej kwotę. Należy dodać jeszcze jeden nominał banknotów w celu wyznaczenia podanej kwoty

```
package instrukcje3;
import javax.swing.JOptionPane;
public class Instrukcje3 {
    public static void main(String[] args) {
        int l10, l5, kwota, liczba_sposobow=0;
        String s, wynik = "";
//wprowadzanie danych o banknotach i kwocie
        do {
            s = JOptionPane.showInputDialog(null,"Podaj liczbe baknotów po 10 tysiecy");
            l10 = Integer.parseInt(s);
            s = JOptionPane.showInputDialog(null, "Podaj liczbe baknotów po 5 tysiecy");
            l5 = Integer.parseInt(s);
            s = JOptionPane.showInputDialog(null, "Podaj kwotę");
            kwota = Integer.parseInt(s);
        } while (l10 <= 0 || l5 <= 0 || kwota <= 0);
//wyznaczenie sposobów wyznaczenia kwoty z dostępnych banknotów
        for (int i = 0; i <= l10; i++)
            for (int j = 0; j <= l5; j++)
                if (5 * j + 10 * i == kwota) {
                    liczba_sposobow++;
                    wynik += "kwota:" + kwota + "tys. zł = ";
                    if (j > 0)
                        wynik += j + " * 5 tys. ";
                    if (i > 0)
                        wynik += i + " * 10 tys. ";
                    wynik += "\r\n"; //dodany znak końca linii
                }
        wynik += " liczba sposobów " + liczba_sposobow + "\r\n";
        JOptionPane.showMessageDialog(null, wynik);
        System.out.println(wynik);
        System.exit(0);
    }
}
```

