

ANS Elbląg
Instytut Informatyki Stosowanej im. Krzysztofa
Brzeskiego
Programowanie obiektowe I – laboratorium

Studium Stacjonarne, sem. 3, 2022/2023

Sprawozdanie nr : 11,
nr grupy: 1,
dzień: wtorek,
godz. 12:00.

Data wykonania ćwiczenia: 24.01

Data oddania sprawozdania: 30.01

Nazwisko i imię: Kuczawski Kacper

Nr albumu: 20195

Nazwa pliku : lab11_kuczawski_kacper20195

1. Przeanalizować i uruchomić przykładowe programy z wykładu nr 8
 - a) przechwytyjący wyjątek **ArrayIndexOutOfBoundsException**
 - b) przechwytyjący wyjątek **ArithmeticException**
 - c) tworzący własną klasę wyjątku

Treść:

```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.text.*;
public class temp{
    public static void main(String args[]) throws RuntimeException{

        int tab[] = new int[10];
        try{
            tab[10]=1337;
        }
        catch (ArrayIndexOutOfBoundsException e){
            System.out.print("\nBłędny indeks tablicy");
        }
        try{
            int liczba = 20/0;
        }
        catch (ArithmeticException e){
            System.out.print("\nWyjątek arytmetyczny");
            System.out.print("\nKomunikat systemowy: "+e.getMessage());
        }
    }
}
```

Treść punktu c:

```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.text.*;
class szkalujo extends Exception{
    public szkalujo(String errorMessage){
        super(errorMessage);
    }
}
public class temp{
    static void f(double l) throws szkalujo{
        if(l==2137){
            throw new szkalujo("\nNie wolno szkalować świętego Jana Pawła
II");
        }
    }
}
```

```

        else{
            System.out.print("\nPapież bezpieczny wynik: "+1);
        }
    }
    public static void main(String args[]){
        double tab[] = {0, 0, 2000, 100, 30, 7, 0, 0, 0, 0};
        double suma=0;
        for(int i=0; i<10; i++){
            suma=suma+tab[i];
        }
        try{
            f(suma);
        }
        catch(szkalujo e){
            System.out.print("\nAle tak papieża?");
            System.out.print("\nKomunikat systemowy: "+e.getMessage());
        }
    }
}

```

Wyniki:

| | |
|-----------------------------------|---|
| Błędny indeks tablicy | Ale tak papieża? |
| Wyjątek arytmetyczny | Komunikat systemowy: |
| Komunikat systemowy: / by zero | Nie wolno skalować świętego Jana Pawła II |
| Process finished with exit code 0 | Process finished with exit code 0 |

3. W klasie **Student2** (Laboratorium 8) zmodyfikować metodę **wstawOceny**, która będzie wstawiać oceny w pola **ocena1**, **ocena2**, a jeśli jeden z argumentów będzie mniejszy od 1 lub większy od 6, generowany będzie odpowiedni wyjątek (napisać własną klasę tego wyjątku - powinna dziedziczyć z klasy **Exception**). W metodzie **wstawOceny** należy zapewnić obsługę tego wyjątku. W metodzie **main()** klasy publicznej **Lab11z3** utworzyć obiekt klasy **Student2** i wywołać dla tego obiektu metodę **wstawOcen**, najpierw z dopuszczalnymi wartościami ocen i następnie z niepoprawnymi.

Treść:

Wyniki:

4. Zmodyfikować program z zadania 3 w taki sposób, aby obiekt wyjątku przechowywał nieprawidłową wartość oceny (trzeba przekazać tę wartość do konstruktora klasy wyjątku).

Treść:

Wyniki:

5. Zmodyfikować metodę **wypożycz** w klasie **Czytelnik** w taki sposób, aby generowała wyjątek, jeżeli książka jest już wypożyczona lub czytelnik nie może wypożyczyć więcej książek. W metodzie **main()** klasy publicznej **Lab11z3** utworzyć obiekt **czytelnik1** klasy **Lab11z5** i wywołać dla niego metodę dla wariantu 1, gdy książka jest wypożyczona i wariantu 2, gdy czytelnik nie może już wypożyczyć więcej książek.

Treść:

Wyniki:

6. Napisać klasę **Pudelko** z polami **długość**, **szerokość**, **wysokość** z konstruktorem o trzech parametrach. Do klasy **Pudelko** dopisać metodę **sklej**, która pobiera jako argument drugi obiekt klasy **Pudelko** i zwraca nowy trzeci obiekt klasy **Pudelko** będący wynikiem sklejenia dwóch pudełek wzdłuż jednej ze ścian, o ile jest to możliwe (czyli oba pudełka posiadają przynajmniej po jednej ścianie o tych samych wymiarach). Jeśli sklejenie nie jest możliwe, metoda generuje wyjątek klasy **NiesklejalnePudelka**.

Treść:

Wyniki: