

# **Laboratorium Metod Obliczeniowych**

Wydział Elektrotechniki Automatyki i Informatyki

Politechnika Świętokrzyska

Studia: **Stacjonarne I stopnia**

Kierunek: **Informatyka**

Data wykonania: **27.10.2017**

Grupa: **3ID13B**

Ocena:

1. Bartłomiej Osak

Temat:

**Szeregi potęgowe w obliczaniu wartości funkcji**

## 1. Szereg potęgowy $\ln(x)$ – program w środowisku MATLAB.

**Polecenie:** Wyznacz wartość funkcji  $\ln(x)$  dla  $x = 0,5$  korzystając z poniższego szeregu:

$$\ln(x+1) = \sum_{i=1}^{\infty} \frac{(-1)^{i+1}}{i} * x^i, x \in (-1, 1 >$$

**Kod źródłowy algorytmu:**

```
function[wynik] = szereg_ln_x(x,n)
format long g;
wynik = 0;
if(x<=-1) || (x>1)
    disp('Zakres wartosci x zawiera się w przedziale (-1,1>');
end
for i=1 : n
    wynik = wynik + (power((-1),(i+1))/i)*power(x-1,i);
end;
end
```

**Opis funkcji:**

Funkcja szereg\_ln\_x przyjmuje dwa argumenty:

- x – wartość, dla której chcemy policzyć wartość szeregu  $\ln(x)$
- n – ilość iteracji

Funkcja zwraca wartość wynik, która jest określana w pętli za pomocą wzoru:

$$\ln(x) = \sum_{i=1}^{\infty} \frac{(-1)^{i+1}}{i} * x^{i-1}, x \in (-1, 1 >$$

Powyższy wzór różni się od  $\ln(x+1)$ , ponieważ zadaniem było przekształcenie go na postać liczącą szereg dla funkcji  $\ln(x)$ . Wymagało to jedynie odjęcia wartości 1 od wykładnik podstawy potęgi x. Po przejściu n iteracji wynik jest zwracany w command line. Ponadto program został zabezpieczony przed podaniem błędnej wartości argumentu x, ponieważ jest ona odgórnie określona stałym przedziałem  $(-1, 1 >$ . Po wpisaniu wartości spoza podanego zakresu wypisywany jest stosowny komunikat.

**Wywołanie funkcji dla:**

$$\ln(x), \text{ gdzie } x = 0,5$$

- Przypisanie do zmiennej x wartości, dla której obliczamy ile wynosi dla  $\ln(x)$ :**  
`x = 0.5`
- Przypisanie do zmiennej n ilości iteracji:**  
`n = 33`
- Wywołanie funkcji szereg\_ln\_x:**  
`szereg_ln_x(x,n)`
- Wynik zadziałania:**  
`ans = -0.693147180556614`

## 2. Szereg potęgowy $\ln(x)$ – program w języku Java.

**Kod źródłowy algorytmu:**

```
import java.util.Scanner;
public class SzeregLnX {

    private static Double x;
    private static Integer n;
    private static Double wynik = 0D;
    private static Scanner s = new Scanner(System.in);
```

```

public static Double getX() {
    return x;
}

public static void setX(Double x) {
    SzeregLnX.x = x;
}

public static Double getWynik() {
    return wynik;
}

public static void setWynik(Double wynik) {
    SzeregLnX.wynik = wynik;
}

public static Integer getN() {
    return n;
}

public static void setN(Integer n) {
    SzeregLnX.n = n;
}

public static void wprowadzX() {
    Double tmpX;
    Boolean koniec = false;
    while (!koniec) {
        try {
            System.out.print("Podaj wartosc x dla ktorej chcesz policzyc szereg (np.
0.3): ");
            tmpX = Double.parseDouble(s.next());
            if (tmpX <= -1 | tmpX > 1) {
                throw new Exception();
            } else {
                koniec = true;
                setX(tmpX);
            }
        } catch (Exception e) {
            System.out.println("Wartosc x musi nalezec do przedzialu (-1,1>!");
        }
    }
}

public static void wprowadzN() {
    Integer tmpN;
    Boolean koniec = false;
    while (!koniec) {
        try {
            System.out.print("Podaj wartosc ilosc iteracji (np. 10): ");
            tmpN = Integer.parseInt(s.next());
            if (tmpN < 1) {
                throw new Exception();
            }
        }
    }
}

```

```

        } else {
            koniec = true;
            setN(tmpN);
        }
    } catch (Exception e) {
        System.out.println("Ilosc iteracji musi byc >=1!");
    }
}

}

public static void obliczWartoscSzeregu() {
    for (int i = 1; i < getN(); i++) {
        setWynik(getWynik() + (Math.pow((-1), (i + 1)) / i) * Math.pow(x - 1, i));
    }

    System.out.println("Wynik szeregu ln(" + getX() + ")=sum[i=1,n=" + getN() +
        "] [((-1)^(i+1)/" + getN() + ") * ("
        + getX() + "-1)^i] = " + getWynik());
}

public static void main(String[] args) throws Exception {
    SzeregLnX.wprowadzX();
    SzeregLnX.wprowadzN();
    SzeregLnX.obliczWartoscSzeregu();
}
}

```

### Opis programu:

W programie występuje jedna klasa publiczna o nazwie SzeregLnX. Posiada ona cztery pola statyczne:

- x – pole typu Double określające wartość, dla której użytkownik chce policzyć szereg  $\ln(x)$ ,
- n – pole typu Integer określające ilość iteracji – wartość jest również określana przez użytkownika,
- wynik – pole typu Double zainicjowane wartością początkową 0, w której będzie przechowywany wynik szeregu
- s – pole typu Scanner odpowiadające za inicjalizację Scannera celem umożliwienia dla użytkownika wpisywania danych z poziomu konsoli

Ponadto w programie występują trzy metody nie wliczając getterów i setterów. Są to:

- wprowadzX() – metoda odpowiedzialna za możliwość wprowadzania przez użytkownika wartości x, dla której chce policzyć szereg. Metoda nie zwraca żadnej wartości, lecz zapisuje wartość wpisaną do pola x.
- wprowadzN() – metoda odpowiedzialna za możliwość wprowadzania przez użytkownika wartości n, czyli ilości iteracji celem uzyskania jak najbardziej poprawnego wyniku szeregu funkcyjnego dla zadanej wartości x. Metoda nie zwraca żadnej wartości, lecz zapisuje wartość wpisaną do pola n.
- obliczWartoscSzeregu() – metoda odpowiedzialna za obliczenie wyniku na podstawie danych zapisanych w polach x oraz n poprzez wywołanie dwóch metod opisanych powyżej. W funkcji zaimplementowano pętlę, która pracuje od wartości numer 1 do wartości n, czyli maksymalnej liczby iteracji. W pętli pole wynik jest ustawiane za pomocą settera zgodnie ze wzorem szeregu:  $\sum_{i=1}^{\infty} \frac{(-1)^{i+1}}{i} * x^i, x \in (-1, 1]$ . Jednakże wzór ten jest odpowiednio przekształcony celem wyliczenia wartości  $\ln(x)$  a nie jak to ma domyślnie zadeklarowane  $\ln(x+1)$ . W tym celu odejmujemy wartość 1 od wykładnika potęgi liczby x. Uzyskujemy zatem wzór:  $\sum_{i=1}^{\infty} \frac{(-1)^{i+1}}{i} * x^{i-1}, x \in (-1, 1]$   
Po zakończeniu działania pętli wypisywana jest zawartość pola wynik wraz z ogólnym wyglądem wzoru szeregu dla podanych danych.

Program został również zabezpieczony przed podaniem błędnych danych przez użytkownika, niezgodnych ze wzorem.

**Uruchomienie programu dla funkcji:**

$\ln(x)$ , gdzie  $x = 0,5$

**a) Wprowadzanie wartości argumentu x, dla której program ma policzyć szereg:**

Podaj wartosc x dla ktorej chcesz policzyc szereg (np. 0.3): 0.5

**b) Wprowadzenie wartości określającej ilość iteracji celem uzyskania poprawnego wyniku:**

Podaj wartosc ilosc iteracji (np. 10): 33

**c) Wypisanie wyniku:**

Wynik szeregu  $\ln(0.5)=\sum_{i=1,n=33} [((-1)^{(i+1)/33}) * (0.5-1)^i] =$   
-0.6931471805530861