**Dokumentacja Projektu II**

**Przedmiot: Wprowadzenie do programowania**

Autor: Katarzyna Bąk

Informatyka, Semestr I

Numer albumu: 48678

**Spis treści**

[Wstęp 3](#_Toc532158604)

[Badanie zbieżności szeregu 3](#_Toc532158605)

[Sprawdzenie wyników 4](#_Toc532158606)

[1. Podanie wartości dla 5 wyrazów 4](#_Toc532158607)

[Zmienne, ich typy oraz zastosowanie 5](#_Toc532158608)

[1. Funkcjonalność A – zastosowane zmienne 5](#_Toc532158609)

[2. Funkcjonalność B – zastosowane zmienne 7](#_Toc532158610)

# Wstęp

Program, który wykonałam w ramach Projektu II na przedmiot Wprowadzenie do programowania umożliwia wyprowadzenie obliczeń dwóch następujących funkcjonalności:

1. Obliczenie sumy szeregu potęgowego z dokładnością do Eps
2. Tablicowanie wartości szeregu potęgowego

Szereg potęgowy, który zastosowałam w obliczeniach mojego projektu wygląda następująco :

Program umożliwia obliczenie dla wartości dla x zawierających się w przedziale zbieżności szeregu oraz dla Eps(dokładność obliczeń) z przedziału 0<Eps<1.

Program oblicza sumę szeregu oraz wyświetla, ile wyrazów zostało zsumowanych.

Tablicowanie wartości sumy szeregu odbywa się dla podanego przez użytkownika przedziału: od dolnej do górnej granicy tego przedziału(muszą być zawarte w przedziale zbieżności szeregu), wartość x zwiększa się o krok H, który również zostaje podany przez użytkownika. Wyniki wyprowadzane są w różnych formatach.

Szczegółowe informacje o warunkach nakładanych na dane wejściowe znajdują się w dalszej części dokumentacji.

# Badanie zbieżności szeregu

Do policzenia sumy szeregu potrzebne jest nam zbadanie jego zbieżności, ponieważ dopiero po stwierdzeniu, że szereg jest zbieżny istnieje możliwość wykonania takiego działania. Szereg jest zazwyczaj zbieżny tylko dla wybranego przedziału. Poniżej znajdują się obliczenia, które wykazują zbieżność szeregu, który został wykorzystany przeze mnie w tym projekcie wraz z przedziałem jego zbieżności.

=

=

To działanie pokazuje, że zbieżność szeregu zależna jest wyłącznie od czynnika. Należy, więc sprawdzić dla jakich x szereg jest zbieżny (tzn. że czynnik jest ułamkiem), ponieważ czynnik jest równy 1.

\* dla 0 szereg ma stałą wartość 0, a dla 1 szereg ma stałą wartość 1.

\*\* we wzorach „H” oznacza wykorzystanie twierdzenia del’Hospitala

# Sprawdzenie wyników

## Podanie wartości dla 5 wyrazów

Załóżmy, że x = 1,1

* n = 1

w1= (1,1 – 1)1 \* sin 1/ = 0,1 \* sin 1 = 0,1 \* 0,84171 = 0,084171

* n = 2

w2 = (1,1 – 1)2 \* sin 1/ = 0,01 \* sin / 2= 0,01 \* 0,75128 = 0,0075128

* n = 3

w3= (1,1 – 1)3 \* sin 1/ = 0,001 \* sin /3= 0,001 \* 0,545806 = 0,000545806

* n = 4

w4= (1,1 – 1)4 \* sin 1/ = 0,0001 \* sin 1/2= 0,0001 \* 0,479426 = 0,0000479426

* n = 5

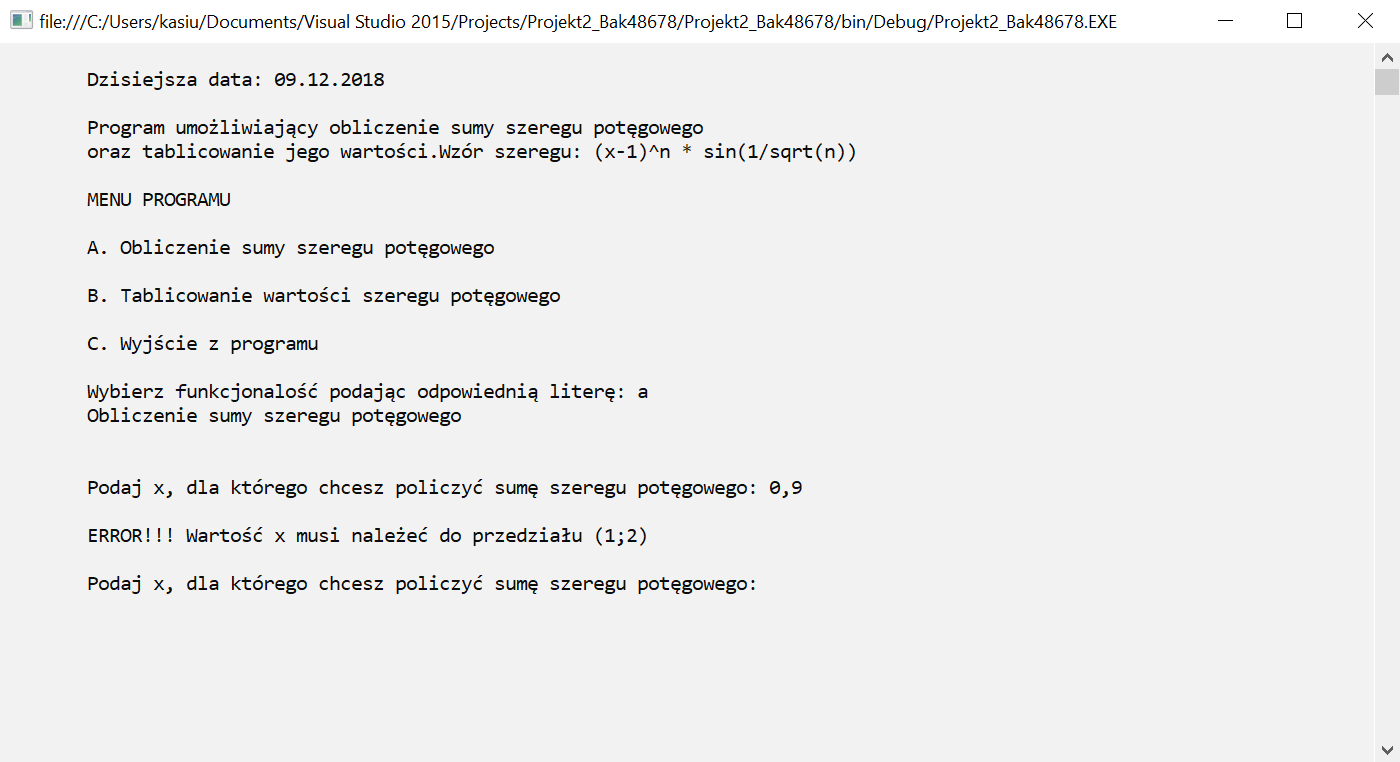
w5= (1,1 – 1)5 \* sin 1/ = 0,00001 \* sin /5= 0,00001 \* 0,432455 = 0,00000432455

# Zmienne, ich typy oraz zastosowanie

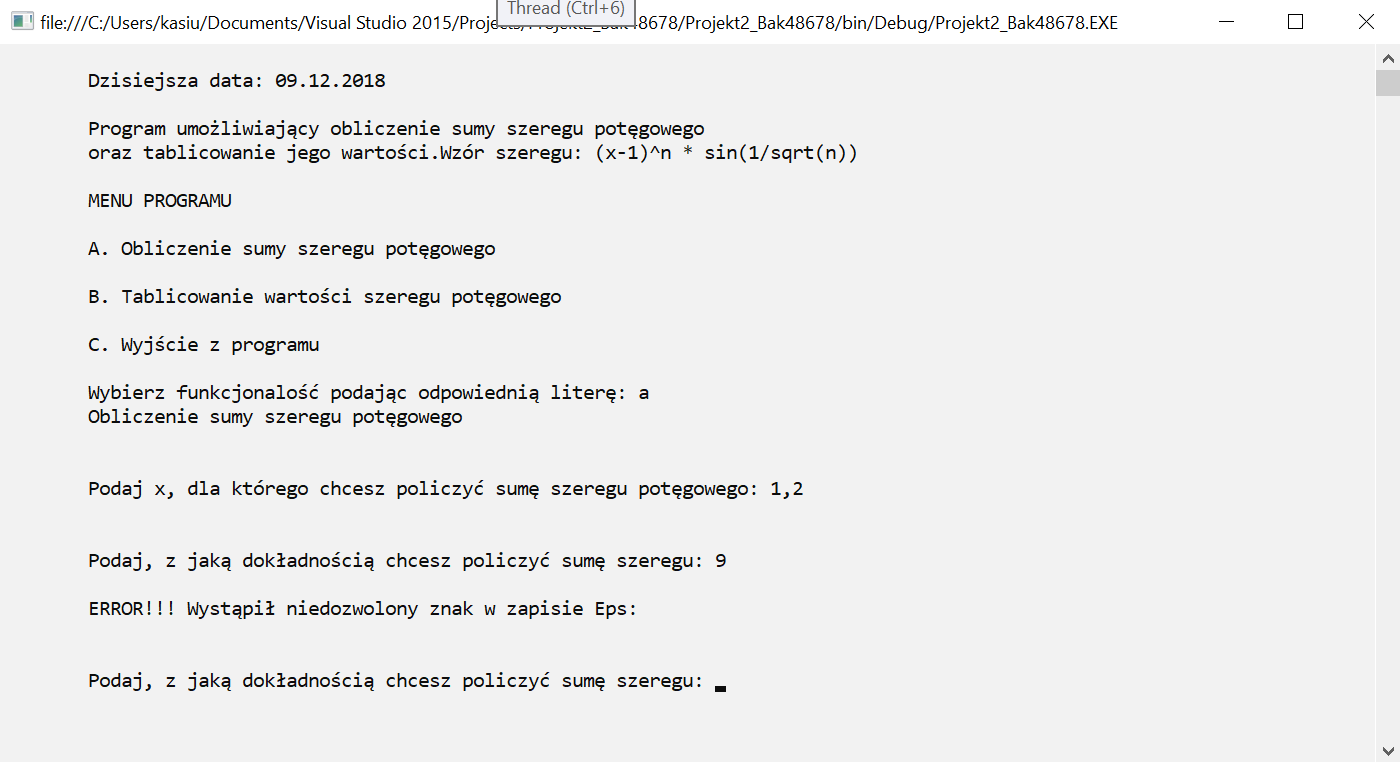
## Funkcjonalność A – zastosowane zmienne

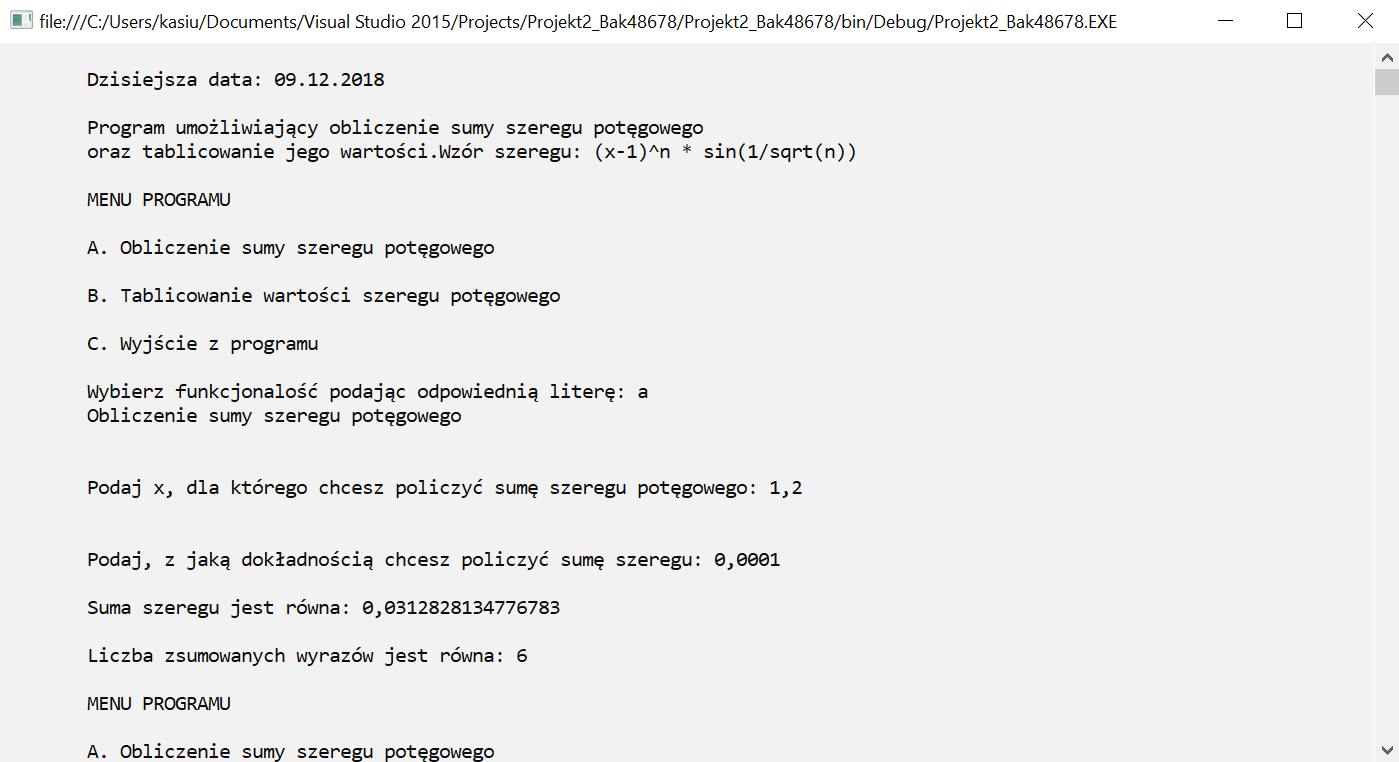
Dane wejściowe:

* x – zmienna liczbowa typu *float*, dla tej zmiennej liczona zostaje suma szeregu, zostaje ona podstawiona za x we wzorze, musi być zawarta w przedziale zbieżności szeregu 1<x<2. W innym przypadku program zasygnalizuje błąd:



* Eps - zmienna liczbowa typu *float*, musi być zawarta w przedziale 0<Eps<1. Jest to dokładność obliczeń, z jaką liczona będzie suma szeregu. W innym przypadku program zasygnalizuje błąd:



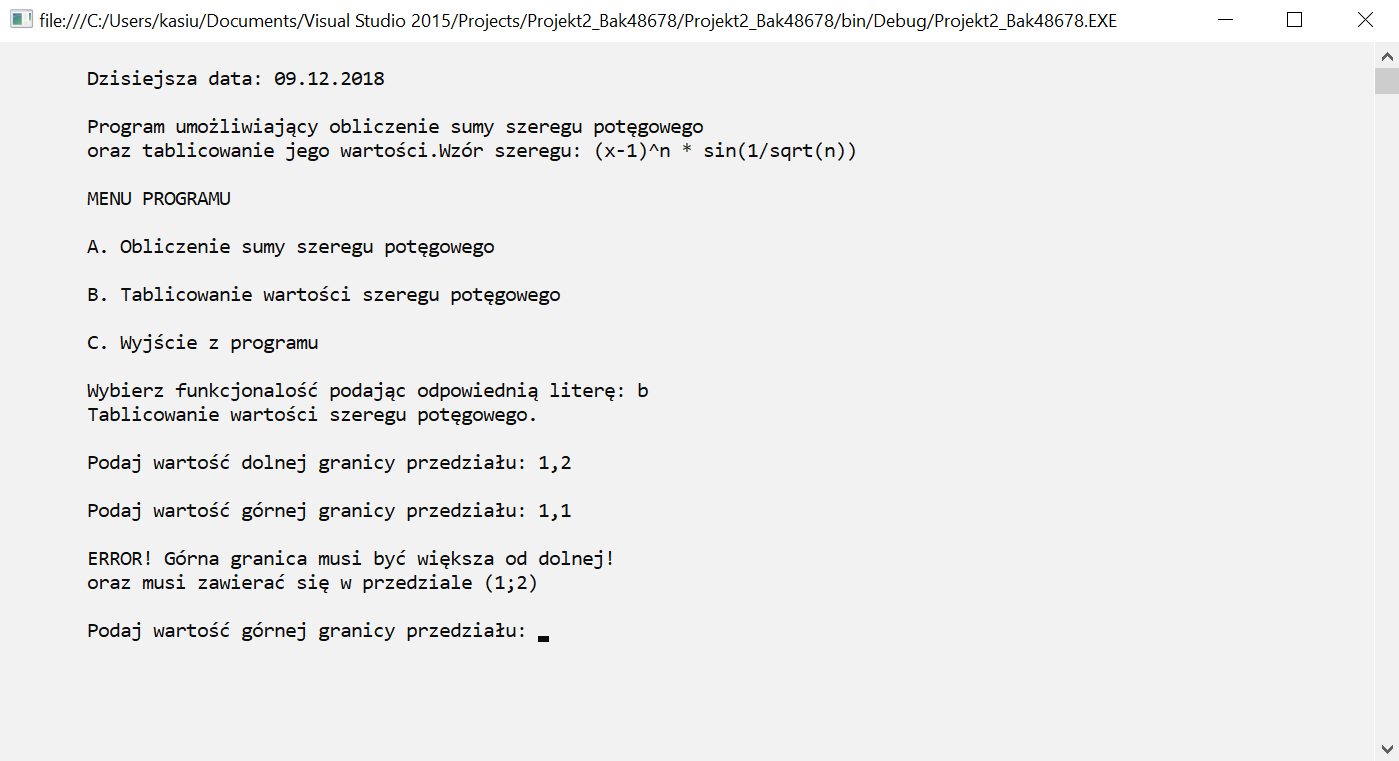
Gdy zostaną wprowadzone poprawne dane wejściowe, program obliczy sumę szeregu. Została ona policzona za pomocą funkcjonalności Suma Szeregu, w której suma zostaje liczona przez dodawanie do siebie kolejnych wyrazów szeregu do momentu, w którym wartość bezwzględna dwóch kolejnych wyrazów będzie mniejsza od podanej wartości epsilon.

Wynik jest zmienną typu *double* ze względu na występujące we wzorze szeregu pierwiastki. Program podaje również, ile wyrazów zostało zsumowanych.

## Funkcjonalność B – zastosowane zmienne

* Dolna granica przedziału Xd – zmienna liczbowa typu *float*, jest to liczba początkowa, pierwszy wyraz, dla którego zostanie policzona suma, musi być zawarta w przedziale zbieżności szeregu 1<x<2. W innym przypadku program zasygnalizuje błąd.
* Górna granica przedziału Xg - zmienna liczbowa typu *float*, jest to ostatni wyraz, dla którego zostanie policzona suma, musi być zawarta w przedziale zbieżności szeregu 1<x<2. W innym przypadku program zasygnalizuje błąd.

Ponadto górna granica przedziału musi być większa od dolnej. W innym przypadku następuje sygnalizacja błędu i obie granice muszą być podane od nowa.



* Krok H - zmienna liczbowa typu *float,* wyznacza ona, o ile ma zwiększać się wartość x, dla którego program ma liczyć kolejne sumy, musi być mniejsza od połowy sumy granic przedziału. W innym przypadku następuje sygnalizacja błędu:

