Tallinn University of Technology

School of Engineering

Integrated Engineering

Emma Hansson

243487MVEB21

**RECURSION**

Second homework in course IAX0584

Supervisor: Vladimir Viies

Tallinn 2025

# AUTHORS DECLARATION

I confirm that I have prepared this homework independently and that it has not been previously submitted for defense by someone else. All the works of other authors used in the preparation of the work, important points of view, data from literary sources and elsewhere are cited in the work.

Author: Emma Hansson

Date: 25.04.2025

**CONTENTS**

[AUTHORS DECLARATION 2](#_Toc196481788)

[1. TASK STATEMENT 4](#_Toc196481789)

[1.1. Specific task 4](#_Toc196481790)

[Specific task is option R-21: 4](#_Toc196481791)

[1.2. Workflow 4](#_Toc196481792)

[1.3. The algorithm 5](#_Toc196481793)

[2. THE CODE IN C 7](#_Toc196481794)

[3. EXPLANATION 9](#_Toc196481795)

[4. EXECUTION 10](#_Toc196481796)

[REFERENCES 11](#_Toc196481797)

# TASK STATEMENT

The task is to construct the algorithm of the task and the corresponding program in C language. All raw data is entered from keyboard by the user. The results are displayed on .txt file.

## Specific task

## Specific task is option R-21:

Create an algorithm and the corresponding program (in C ) to:

1.From the keyboard, the real numbers X(⏐X⏐<1) ja ε (0<ε<1) are entered;

2. Using a recursive function, a real array A is formed with elements

A1 = X,

A2 = –1/2 \* X3/3,

A3 = 1/2 \* 3/4 \*X5/5,

A4 = –1/2 \* 3/4 \* 5/6 \* X7/7,

. . .

up to the number of elements L of array A either satisfies the condition ⏐A(L) – A(L – 1) ⏐ ≤ ε or (if this condition is not satisfied) L = 15;

3. The number L of the elements of the array A and the elements itself are output to the file F with indexes.

## Workflow

Simple description of the workflow of programme:

 **Data input:** The program takes two real number inputs from the keyboard:

* X, where 21​<∣X∣<1.
* ε, where 0<ε<1.

 **Array generation (recursive function):** A recursive function is called to generate the elements of a real array A. The elements are calculated according to a pattern. The recursion continues to generate new elements and add them to the array A until one of the following conditions is met:

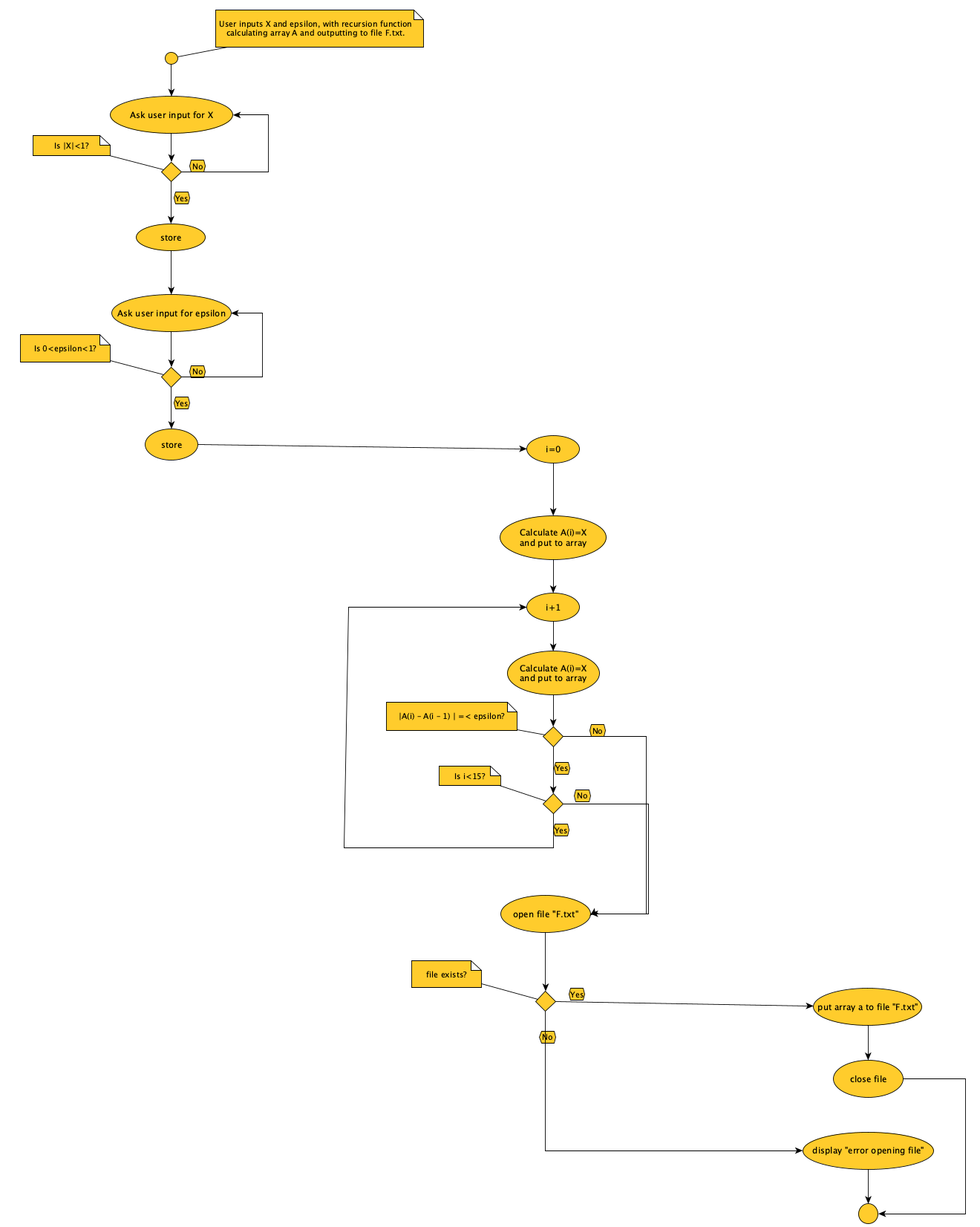
* The absolute difference between the last two generated elements is less than or equal to ε: ∣A(L)​−A(L−1)​∣≤ε.
* The number of elements in the array A reaches a maximum limit of 15 (L=15).

 **Output to file:** Once the array A is generated and the number of elements L is determined, the program writes the following information to a file named "F":

* The total number of elements in the array, L.
* Each element of the array A along with its index (starting from 1). The output format for each element should clearly indicate the index and the corresponding value.

## The algorithm

Based on the workflow, I conducted an algorithm, using “yEd” graph editor. Alrogithm has 3 separate parts: input, process and output, which are shown visually by columns.

****

Breakdown of the algorithm:

**1. Get Input X:**

* The program prompts the user to enter a real number, X.
* It checks if X is within the valid range (-1 < X < 1).
* If X is not valid, it displays an error message and repeats the input prompt.

**2. Get Input Epsilon:**

* The program prompts the user to enter a real number, epsilon (e).
* It checks if e is within the valid range (0 < e < 1).
* If e is not valid, it displays an error message and repeats the input prompt.

**3. Generate Array A:**

* The program initializes an array A.
* It uses a recursive function to calculate the elements of A:
  + The first element A[0] is set to X.
  + Subsequent elements A[i] are calculated using a specific formula (involving powers of X and factorials).
* The recursion continues until one of these conditions is met:
  + The absolute difference between two consecutive elements |A[i] - A[i-1]| is less than or equal to e.
  + The array A has 15 elements.

**4. Write Array to File:**

* The program opens a file named "F.txt" in write mode.
* If the file cannot be opened, it displays an error message and stops.
* If the file is opened successfully, it writes the following to the file:
  + The number of elements in array A (which is L + 1, because the array index starts at 0).
  + Each element of array A, along with its index (e.g., "A[0] = value").
* The program closes the file.

**5. End:**

* The program is done.

# THE CODE IN C

A screen shot of a computer program

AI-generated content may be incorrect.

**A screenshot of a computer program

AI-generated content may be incorrect.**

**A screen shot of a computer program

AI-generated content may be incorrect.**

# EXPLANATION

This program takes two numbers as input, uses them to create a list of numbers according to a specific mathematical rule, and then saves that list into a file.

Program Structure:

The program is organized into several parts (functions):

* main() Function: This is the starting point of the program. It controls the order in which the other parts of the program run.
* getInputX() Function: This function gets a number called "X" from the user. It also checks if the number is within a valid range (between -1 and 1) and asks the user to enter it again if it's not.
* getInputEpsilon() Function: This function gets another number called "epsilon" from the user. It checks if the number is within a valid range (between 0 and 1) and asks the user to re-enter it if it's not.
* generateArray() Function: This is the core of the program. It calculates a series of numbers and stores them in a array. The calculation involves a repeating pattern, and the function uses recursion to do this.
* writeArrayToFile() Function: This function takes the list of numbers created by generateArray() and saves it to a file named "F.txt" on computer.

How the Program Works Step-by-Step:

1. Get Input:
   * The main() function starts by using the getInputX() function to get the value of "X" from the user. The program makes sure that X is a valid number.
   * The main() function then uses the getInputEpsilon() function to get the value of "epsilon" from the user. The program also validates this input.
2. Generate Number List:
   * The main() function calls the generateArray() function to create a list of numbers.
   * The generateArray() function calculates the numbers in the list one at a time. The value of each number depends on the previous numbers in the list and the value of "X".
   * The program stops adding numbers to the list when either:
     + The difference between two consecutive numbers in the list is less than epsilon, or
     + The list contains 15 numbers.
3. Save to File:
   * The main() function calls the writeArrayToFile() function to save the generated list of numbers to a file.
   * The writeArrayToFile() function creates a file named "F.txt" (or opens it if it already exists) and writes the numbers from the list into that file.

# EXECUTION

Input: X = 0,9 and epsilon = 0,1

Output:

A screenshot of a computer

AI-generated content may be incorrect.

Input: X = epsilon = 0,5

Output:

A screenshot of a computer

AI-generated content may be incorrect.

# REFERENCES

Vladimir Viies. (2022). Homework formatting requirements. <https://www.tud.ttu.ee/im/Vladimir.Viies/materials/IAX0583_Programmeerimine%20I/pR0583en/Homeworks%20formatting.pdf> (25.04.2025).

https://www.tud.ttu.ee/im/Vladimir.Viies/materials/IAX0584PROGRAMMEERIMINE%20II/PR2\_iax0584/PR2en/PR2\_en25/help13\_25preex1HWdef/home2Pr2/ (24.04.2025).

Google. (2023). https://gemini.google.com/app (24.04.2025).