Program that calculates an expression

KH-221b.e

Vladislav Storozhenko

Software Requirements Specification Document

Version: 1.0 **Date: 11.15.2021**

Table of Contents

1. Introduction

1.1 Purpose

Purpose of this document is to provide a description of a computer program that has been developed by a student during laboratory training, and outline its capabilities and requirements.

1.2 Scope

- The final product will represent an executable file of any chosen name.
- The program calculates a mathematical expression
- The software may be used in order to solve a particular mathematical expression, other relevant use cases include:
    - using it as a foundation for future laboratory trainings.
    - checking a student's knowledge by asking to explain how the program operates.

1.3 Definitions, Acronyms, and Abbreviations

This document does not use any special terms that may require definitions, not acronyms or abbreviations.

1.4 References

This document does not reference any external entities directly. Nevertheless, the methodical recommendation in which all the tasks were obtained worths mentioning, it can be found in the google classroom at this address: https://classroom.google.com/u/0/w/MTQ3MTQzMzg3MjA4/tc/MTczNjU0NTkxNzYy

1.5 Overview

Potential users should pay more attention to section №2 of this document, developers and testers should focus on section №3.

2. The Overall Description

The program calculates a specific mathematical expression, you can see it on Picture 1.1

$$y=\begin{cases}\sum_{i=1}^{n-1}\sum_{j=1}^{n}\frac{j}{j^2+j}, x<0\\ x-\sum_{i=1}^{n-1} i, x\geqslant 0\end{cases}$$

Picture 1.1 - the task of the previous lab

You input unknown variables *n, x* and obtain the result (y).

2.1 Product Perspective

The program is independent and fully self-contained, it does not require any perspective.

2.1.1 System Interfaces

There are no external systems that the program interacts with.

2.1.2 Interfaces

The program uses the command line as an interface, it is designed for the general population, and has no special requirements. Accessibility issues are addressed by the operating system that the software runs on and beyond the scope of this document.

2.1.3 Hardware Interfaces

The hardware interface is the command line/terminal that can be accessed on any computer.

2.1.4 Software Interfaces

The system has no software interfaces.

2.1.5 Communications Interfaces

The system has no communications interfaces.

2.1.6 Memory Constraints

The system requires so little memory, that it can be neglected.

2.1.7 Operations

The system does not require any operations

2.1.8 Site Adaptation Requirements

The system does not require any site adaptations

2.2 Product Functions

The only function of the product is to get input from the user and perform a calculation in order to solve mathematical expression (see Picture 1.1)

2.3 User Characteristics

Intended users of the program are students and teachers, because they may benefit from calculation of an expression, and from carrying out such type of work during laboratory training.

2.4 Constraints

The system has no constraints

2.5 Assumptions and Dependencies

This document assumes that the program is built for the operating system it is running on.

2.6 Apportioning of Requirements

The system has no apportioning of requirements

3. Specific Requirements

The system shall:

- get input from the user
    - input *x*
    - input *n*
- calculate expression based on user input
- display answer

Activity and use-case diagrams may be found in Appendix A

3.1 External Interfaces

Keyboard that the user can use in order to input values.

3.2 Functions

- Validity checks of the input values
- Calculation
- Answer output

3.3 Performance Requirements

There are no performance requirements for the system.

3.4 Logical Database Requirements

There are no logical database requirements for the system.

3.5 Design Constraints

3.5.1 Standards Compliance

There are no standards that the system has to comply with.

3.6 Software System Attributes

Program should run smoothly in the console window.

3.6.1 Reliability

The program is reliable enough if it is just working and not crashing

3.6.2 Availability

The system should run infrequently, on-demand only.

3.6.3 Security

The software does not require any security features.

3.6.4 Maintainability

The software does not require any additional maintainability features.

3.6.5 Portability

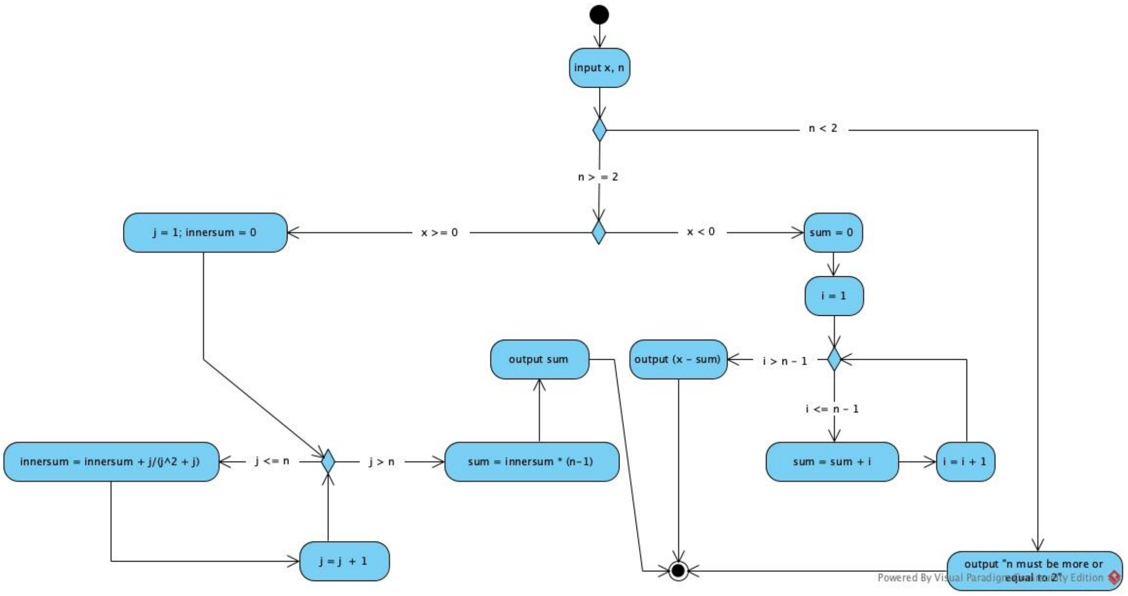
The software does not require any portability features.

4. Change Management Process

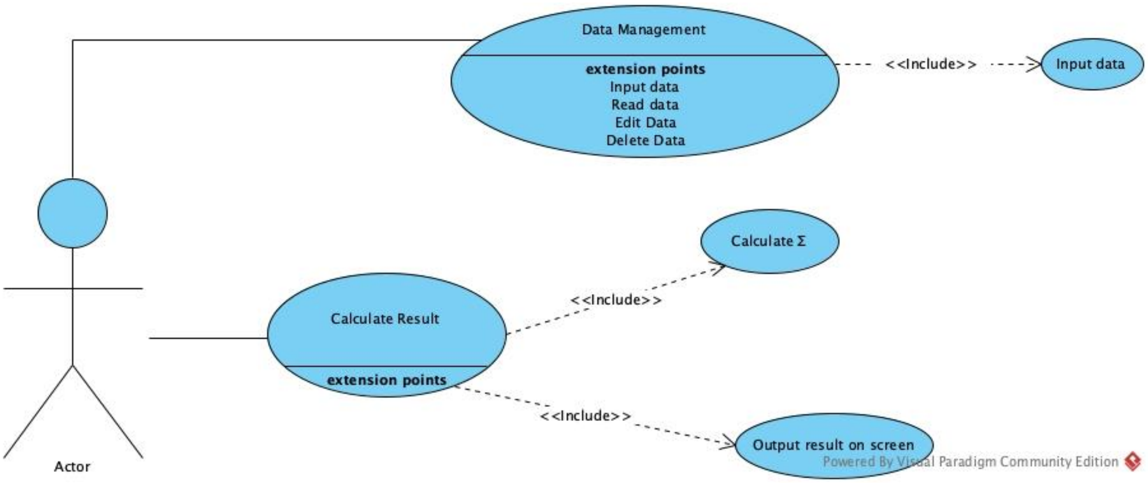
The program will not change at all, change management is obsolete

5. Supporting information

Appendix A



Picture 5.1 - Activity Diagram



Picture 5.2 - Use-Case Diagram