Vocational high school in computer programming

and innovation

**Team:**

**Viktor Velizarov**

**Ivan Ivanov**

**Krasiyana Kamburova**

**Petar Petkov**

**Stefan Diomov**

**Table of content:**

1. **Authors**
2. **Purpose**
3. **Diagram**
4. **Main steps in the implementation of the project**
5. **Elements**

**5.1**

**5.2 Developer-Oriented**

**5.3 Developer Environment**

1. **DevOps Tools**
2. **Description of used functions**

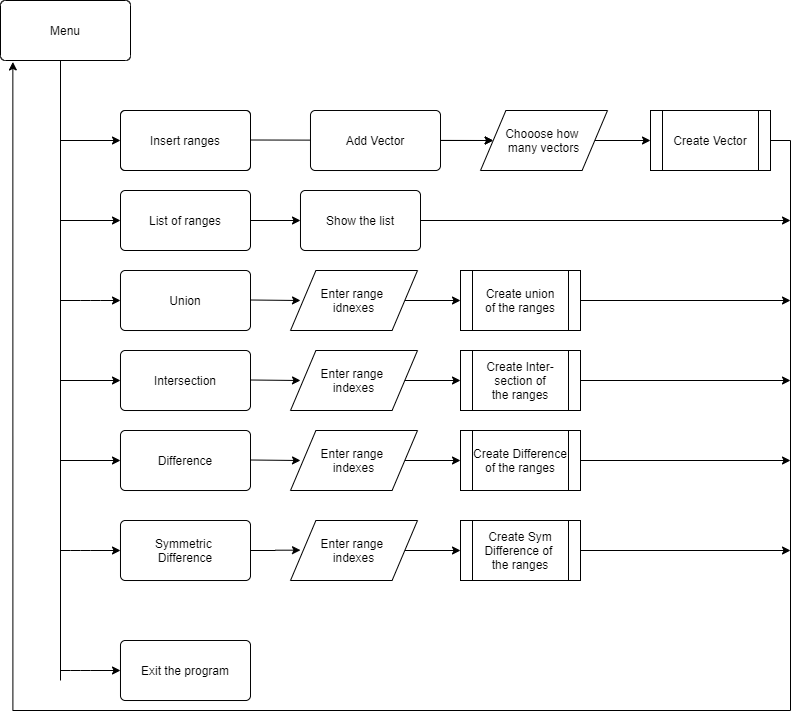
**1. Authors:**

* **Viktor Velizarov -** [**VVVelizarov18@codingburgas.bg**](mailto:VVVelizarov18@codingburgas.bg) **- Class XA**
* **Ivan Ivanov -** [**ITIvanov18@codingburgas.bg**](mailto:ITIvanov18@codingburgas.bg) **- Class XB**
* **Krasiyana Kamburova -** [**KSKamburova18@codingburgas.bg**](mailto:KSKamburova18@codingburgas.bg) **- Class XB**
* **Petar Petkov -** [**PGPetkov@codingburgas.bg**](mailto:PGPetkov@codingburgas.bg) **- Class XG**
* **Stefan Diomov -** [**SGDiomov18@codingburgas.bg**](mailto:SGDiomov18@codingburgas.bg) **- Class XG**

**2. Purpose:**

**Our program is designed to make your work in mathematics easier. The user enters sets into the program, and it can find their difference, cross section, union, and symmetrical difference. The program saves you a lot of time working with sets.**

**3. Diagram:**



**4. Main steps in the implementation of the project:**

* **Defining the main idea**
* **Defining the purpose of the project**
* **Implementing the project by writing code**

**The application is written in C++**

**5. Elements:**

* **The structure of the application is as easy to use as possible. The information is consistent with the selected objectives. The main principles for building an application are:**

**5.1:**

* **Our application designed in seconds to give you the answer to the difficult mathematical sets that have tormented you for so long. It definitely saves a lot of time.**

**5.2 Developer-Oriented:**

* **The biggest hurdle to rapid development is often not your development process, but how much time you spend focusing on the business logic of the function you are working on.**

**5.3 Developer Environment:**

* ***An example of an easy-to-work-with development environment:***

1. **A developer clones a GitHub repo**
2. **He or she runs a couple of commands from a make file**
3. **Tests run**
4. **The application comes up and is accessible**
5. **Code changes are apparent in the running application**

**6. DevOps Tools:**

**In addition to makе it easier to understand and work with the application, one way to improve it is to reduce the time we spend on our own infrastructure. Developers who do not have an environment that is easy to work with from the beginning must always take the time to make the environment easy to work with.**

**7. Description of used functions:**

|  |  |  |
| --- | --- | --- |
| **Funtion Name** | **Description** | **Description of arguments** |
| **createVector** | **function used to initialize the contents of a new vector** | **vector<string> vArray[] - array of vectors**  **int& n – total number of vectors** |
| **addVector** | **function that adds a new vector to the array of vectors** | **vector<string> vArray[] - array of vectors**  **int& n – total number of vectors** |
| **PrintVector** | **function used to print all the vectors on the screen** | **vector<string> vArray[] - array of vectors**  **int& n – total number of vectors** |
| **checkIndex** | **funtion that checks if the inputed index is valid** | **Int firstIndex – first index we check**  **Int secondIndex – second index we check**  **int& n – total number of vectors** |
| **Difference** | **function that finds the difference between two ranges** | **vector<string> vArray[] - array of vectors**  **int& n – total number of vectors** |
| **SymmetricDifference** | **function that finds the symmetric difference between two ranges** | **vector<string> vArray[] - array of vectors**  **int& n – total number of vectors** |
| **Intersection** | **function that finds the Intersection between two ranges** | **vector<string> vArray[] - array of vectors**  **int& n – total number of vectors** |
| **Union** | **function that finds the Union between two ranges** | **vector<string> vArray[] - array of vectors**  **int& n – total number of vectors** |