Vocational School of Computer Programming and Innovations

Project Bank Deposits

Participant:

Georgi Zhechev, GTZhechev18@codingburgas.bg

Burgas, 2020

# Subject: Sets operations

# Documentation

## Title / cover page

PGKPI Burgas

# Project name: Sets of elements

Georgi Zhechev GTZhechev18@codingburgas.bg

## Content

Contents

[Vocational School of Computer Programming and Innovations 1](#_Toc39138341)

[Sets of elements 1](#_Toc39138342)

[1 Subject: Sets operations 2](#_Toc39138343)

[2 Documentation 2](#_Toc39138344)

[2.1 Title / cover page 2](#_Toc39138345)

[3 Project name: Sets of elements 2](#_Toc39138346)

[3.1 Content 2](#_Toc39138347)

[3.2 Short description of our project and objectives 2](#_Toc39138348)

[3.3 Diagram describing the menu 3](#_Toc39138349)

[3.4 Basic functions description 4](#_Toc39138350)

## Short description of our project and objectives

The purpose of our project is to perform operations with sets. We calculate a set of element and then realize them by arrays. We print them in a main menu with a list of all the functions and operations.

First we input a number of elements, set of elements and stores them into an array.

After we view if the array has element and if so prints them on the screen.

The function Present is an auxiliary function and return true if the element el is an element of the array set with brSet.

The function UnionSet combines two sets into a new set called Union Set and calls ViewSet to print the result.

The function SectSet finds the section of two sets and calls ViewSet to print the result.

DiffSet subtrackts two sets and creates a new set called DiffSet and calls ViewSet to print the result.

FeaturesSet prints set features and creates a new set called FeatureSet and calls ViewSet to print the result.

## Basic functions description

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Description** | **Arguments** | **Returned Value** |
| input\_depositor() | Enters a new depositor. The user can enter name, family name, amount and currency of the deposit. | * Array of depositors * Array size | None |
| input\_deposit() | It inputs a new deposit. When entering a new deposit, the program requires the user to enter the depositor’s number. If such number doesn’t exist, the program returns to the main menu, otherwise it asks for the currency type. If the currency exists the system asks for amount and adds it to the deposit, otherwise it creates new deposit. | Array of depositors  Array size | None |
| withdraw\_deposit() | It withdraws a deposit from the current account partially or completely. When withdrawing a deposit the program requires the user to enter the depositor’s number. If such number doesn’t exist, the program returns to the main menu, otherwise it asks for the currency type. If the user enters wrong type, the system returns to the main menu. If the currency is correct the system asks for the amount. If the amount is greater than the deposit, the system outputs a message and returns to the main menu. If the amount is less than the deposit’s amount the system withdraws the amount. If the amount equals to the deposit, the system deletes the amount. | Array of depositors  Array size | None |
| view | It shows the list of all deposits and all depositors. If there is no depositors it outputs a message. | Array of depositors  Array size | None |