

Постановка задачи

Разработать детальные требования и тест план для следующей задачи:

permutationArray. Написать программу, принимающую на вход количество элементов в массиве и сам массив и выводящую в стандартный поток вывода массив с попарно заменёнными элементами

Детальные требования

1. Количество элементов в массиве (size) должно быть задано корректно:
 - 1.1. size — целое число. Если size не является целым числом, то выводится сообщение об ошибке и программа завершается
 - 1.2. size — положительное число. Если size не является положительным числом, то выводится сообщение об ошибке и программа завершается
2. Элементы массива должны быть заданы корректно:
 - 2.1. Каждый элемент массива — целое число. Если элемент массива не является целым числом, то выводится сообщение об ошибке и программа завершается
3. size и элементы массива заданы корректно:
 - 3.1. Если size и элементы массива заданы корректно, то программа должна завершиться с выводом в консоль сообщения, содержащего массив с попарно заменёнными элементами и кодом возврата 0

Тест-план

Проверка детальных требований с помощью тест-плана:

#	Описание	Результат
1.1	size — целое число. Если size не является целым числом, то выводится сообщение об ошибке и программа завершается	Input: 1.5 Expected: Input error!
1.2	size — положительное число. Если size не является положительным числом, то выводится сообщение об ошибке и программа завершается	Input: -12 Expected: Incorrect size of array!
2.1	Каждый элемент массива — целое число. Если элемент массива не является целым числом, то выводится сообщение об ошибке и программа завершается	Input: 5 1 a 3 4 5 Expected: Input error!
		Input: 2 1.12 37 Expected: Input error!
3.1	Если size и элементы массива заданы корректно, то программа должна завершиться с выводом в консоль сообщения, содержащего массив с попарно заменёнными элементами и кодом возврата 0	Input: 5 1 2 3 4 5 Expected: 2 1 4 3 5
		Input: 4 2 1 4 3 Expected: 1 2 3 4

Исходные тексты программы

Файлы с исходными текстами лабораторной работы (полагаем <ROOT> для папки в котором располагаются исходные тексты):

./<ROOT>/main.cpp

```
#include <iostream>
#include <cstdint>
#include "arrayFunction.hpp"

int main()
{
    size_t size = 0;
    std::cin >> size;
    if (!std::cin)
    {
        std::cerr << "Incorrect size of array! \n";
        return 1;
    }

    int * arr = nullptr;
    try
    {
        arr = new int[size];
    }
    catch (const std::exception & e)
    {
        delete [] arr;
        std::cerr << e.what();
        return 2;
    }

    try
    {
        inArray(arr, size);
    }
    catch (const std::exception & e)
    {
        delete [] arr;
        std::cerr << e.what();
        return 1;
    }

    permutationArray(arr, size);
    outArray(arr, size);

    delete [] arr;
    return 0;
}
```

./<ROOT>/arrayFunction.cpp

```
#include "arrayFunction.hpp"
#include <iostream>

void inArray (int * arr, size_t size)
{
    for (size_t i = 0; i < size; ++i)
    {
        std::cin >> arr[i];
        if (!std::cin)
        {
            throw std::logic_error("Input error!");
        }
    }
};

void outArray (int * arr, size_t size)
{
    for (size_t i = 0; i < size; ++i)
    {
        std::cout << arr[i] << " ";
    }
};

void permutationArray (int * arr, size_t size)
{
    for(size_t i = 0; i < (size - 1); i += 2)
    {
        arr[i] += arr[i + 1];
        arr[i + 1] = arr[i] - arr[i + 1];
        arr[i] -= arr[i + 1];
    }
};
```

./<ROOT>/arrayFunction.hpp

```
#ifndef ARRAYFUNCTION_HPP
#define ARRAYFUNCTION_HPP
#include <cstdint>

void inArray (int * arr, size_t size);
void outArray (int * arr, size_t size);
void permutationArray (int * arr, size_t size);
#endif
```