**Постановка задачи**

Разработать детальные требования, тест план и программу для следующей задачи:

Проверить, есть ли в двух заданных строках одинаковые символы. Функция должна  
возвращать true, если есть одинаковые символы, false – в противном случае

* ***напишите функцию main***, в которой будут вводиться данные,  
  выделяться память под строки, вызываться функция, обрабатываться исключения, освобождаться  
  память и выводиться результаты.
* Максимальную длину строки (строк) и сами символы строки (строк) надо вводить из файла.  
  Данные для всех тестов могут быть в одном файле. Если в задаче требуется формирование строкирезультата, то ее надо вывести в файл-результата.
* Строки в стиле С разместить в динамической памяти, максимальный размер строки-результата  
  определить на основе введенных данных.  
  Для исключения повторных проходов по строкам, можно использовать дополнительную память.
* Результаты всех выполняемых тестов нужно вывести в один файл.

1. **Пример:**

Дана строка: aaaab. Дана вторая строка: alsdfgj. Тогда совпавшим элементом будет «a», следовательно выведется «true».

1. **Детальные требования**
   1. Вводимый файл должны быть.
      1. Вводимые данные любые.
         1. Если файла нет, то вывести сообщение «The file was not open!», завершив программу

**Тест-план**

|  |  |  |  |
| --- | --- | --- | --- |
| Требование | Детальные требования | Данные | Ожидаемый результат |
| 2.1. Вводимые данные должны быть. | | | |
| Вводимый файл должны быть. | 2.1.1.1 Если файла нет, то вывести сообщение «The file was not open!», завершив программу |  | Сообщение «The file was not open!», |

**Код:**

#include <iostream>

#include <fstream>

const char\* ERROR\_FILE\_NOT\_OPEN = "The file was not open!";

bool Comparison(char array1[], char array2[], int count1, int count2)

{

for (int i = 0; i < count1; i++)

{

for (int j = 0; j < count2; j++)

{

if (array1[i] == array2[j])

return 1;

}

}

return 0;

}

int main()

{

int max\_size = 0;

std::ifstream input("C:/Users/PetrK/Documents/Visual Studio Files/ConsoleApplication15/input.txt");

try

{

if (!input.is\_open())

{

throw ERROR\_FILE\_NOT\_OPEN;

}

}

catch (const char\* error)

{

std::cerr << error;

return -1;

}

while (!input.eof())

{

char object = 'a';

input >> max\_size;

char\* array1 = new char[max\_size];

char\* array2 = new char[max\_size];

int i = 0;

while ((i < max\_size) && (object != '\0'))

{

input >> object;

//std::cout << object;

array1[i] = object;

i++;

}

//std::cout << std::endl;

int count1 = i;

i = 0;

object = 'a';

while ((i < max\_size) && (object != '\0'))

{

input >> object;

//std::cout << object;

array2[i] = object;

i++;

}

//std::cout << std::endl;

int count2 = i;

if (Comparison(array1, array2, count1, count2) == 1)

{

std::cout << "true" << std::endl;

}

else

{

std::cout << "false" << std::endl;

}

delete[] array1;

delete[] array2;

return 0;

}

}