Министерство образования Республики Беларусь

УО «Брестский государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №2**

По дисциплине: “ООТПиСП”

Тема: “Потоковые классы”

**Вариант №8**

**Выполнил**: студент 2 курса группы ПО-7 Крупенков Михаил Дмитриевич

**Проверил:** Хацкевич М.В.

Брест 2022

**Цель.** Научиться программировать ввод и вывод в С++, используя объекты потоковых классов стандартной библиотеки С++.

**Постановка задачи.** В файле большого размера записан некоторый текст. Требуется записать в другой файл его копию в обратном порядке символов. Для обработки текста использовать "окно" в памяти компьютера размером 1024 байт.

**Код программы**

#include **<iostream>**#include **<fstream>**#include **<algorithm>  
  
using namespace** std;  
  
**const char** FILENAME[] = **"..\\files\\Lorem.txt"**;  
*//const char FILENAME[] = "..\\files\\Alphabet.txt";  
//const char FILENAME[] = "..\\files\\Digits.txt";***const char** FILENAME\_NEW[] = **"..\\files\\NewFile.txt"**;  
**const long long** BUFFER\_SIZE = 1023;  
**char** buffer[BUFFER\_SIZE + 1];  
  
  
**int** main() {  
 ifstream file(FILENAME, ios\_base::ate | ios\_base::binary);  
 ofstream fileNew(FILENAME\_NEW, ios\_base::trunc | ios\_base::binary);  
  
 **if** (!file.is\_open() **or** !fileNew.is\_open()) {  
 cout << **"Some of files are not found"**;  
 **return** -1;  
 }  
  
 **long long** cursor = file.tellg() - BUFFER\_SIZE; *// Курсор - номер позиции в файле = BUFFER\_SIZE с конца* **while** (cursor > 0) {  
 file.seekg(cursor, ios\_base::beg); *// Перенос указателя в потоке файла на курсор* file.read(buffer, BUFFER\_SIZE); *// Чтение в буфер BUFFER\_SIZE символов* reverse(buffer, buffer + BUFFER\_SIZE); *// Поворот буфера* fileNew << buffer; *// Запись в новый файл* cursor -= BUFFER\_SIZE; *// Смещение курсора* }  
 cursor += BUFFER\_SIZE; *// Теперь курсор - это конец: запись с 0 до cursor* file.seekg(ios\_base::beg); *// Перенос указателя в потоке файла на начало* file.read(buffer, cursor); *// Чтение в буфер cursor символов* buffer[cursor] = **'\0'**; *// Заполяется не весь буфер, поэтому надо отделить* reverse(buffer, buffer + cursor);  
 fileNew << buffer;  
  
 file.close();  
 fileNew.close();  
}

**Результаты программы**

|  |  |  |
| --- | --- | --- |
| Название | Оригинальный файл | Новый файл |
| Lorem.txt | Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.  Nunc viverra imperdiet enim. Fusce est. Vivamus a tellus.  Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci.  Aenean nec lorem. In porttitor. Donec laoreet nonummy augue. | .im a sumaviV .sucnohr gnicsipida etna lev angam non sarC .susir lev angam eugua etatupluv cen ,ucra supmet sore maid ,rotcua satsege ainical tilev ,rotittrop euqsetnelleP  .silef sittigas susruc euqsetnelleP .silef ,di nasmucca ,ca rotcua ,lev etatupluv ,rolod oido saneceaM .angam ca cnuN  .satsege siprut ca semaf adauselam te suten te sutcenes euqitsirt ibrom tnatibah euqsetnelleP .angam ropmet mauqila euqsiuQ .eugua cen niorP  .euqitsirt atrop cnun ni merol ta deS .taptulov tare mauqilA .iud tege maitE |
| Alphabet.txt | abcdefghijklmnopqrstuvwxyz  ABCDEFGHIJKLMNOPQRSTUVWXYZ | ZYXWVUTSRQPONMLKJIHGFEDCBA  zyxwvutsrqponmlkjihgfedcba |

**Пояснение**

Сперва бинарно открываются два файла: для чтения с конца и дня записи. Они открываются и производится проверка на это. Далее создается переменная-курсор с номером длины буфера с конца.

В буфер циклично берется текст из основного файла с позиции курсора, переворачивается и ложится в другой файл. Курсор смещается в сторону начала и длину буфера.

В количество текста останется меньше буфера, он будет браться с начала до курсора. После этого файлы закрываются и программа завершает работу

**Вывод:** я преисполнился.