МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



**Дніпровський національний університет  
залізничного транспорту імені академіка В. Лазаряна**

Кафедра «Комп'ютерні інформаційні технології»

**Лабораторна робота №8**

**з дисципліни «Основи програмування»**

**на тему: «**Файловий ввід-вивід. Робота з текстовими файлами. Функції обробки символьних рядків.**»**

Виконав:

студент гр.ПЗ1911

Сафонов Д.Є.

Прийняла:

Нежуміра О. І.

Дніпро, 2020

**Тема.** Файловий ввід-вивід. Робота з текстовими файлами. Функції обробки символьних рядків.

**Мета.** Формування навиків роботи:

1. з текстовими файлами засобами мови С++;
2. з функціями обробки символьних рядків засобами стандартної бібліотеки.

# **Постановка завдання (завдання до лабораторної роботи, індивідуальне завдання).**

Розробити багатофайлову програму для роботи з текстовими файлами для заданого індивідуального завдання і обраного рівня складності.

Вимоги до програми:

* для обробки файлів використати рядки в стилі С;
* максимально задіяти функції стандартної бібліотеки (забороняється дублювати дії функцій стандартної бібліотеки власними функціями);
* імена файлів ввести з клавіатури;
* передбачити перевірку всіх операцій роботи з файлами (відкриття, читання, запису);
* управління виконанням програми здійснити на основі текстового меню користувача;
* передбачити обробку необмеженої кількості файлів;
* результати роботи програми вивести у файл та продублювати на екрані.

Вимоги до тексту програми:

* специфікації програми і її функцій;
* коментарі щодо призначення блоків програми, дій окремих операторів для пояснення алгоритму;
* самодокументованість коду: всі ідентифікатори повинні мати назви, що відповідають суті змінних.

Рівень III (90-100 балів):

Завдання. Програма обробки текстового файлу по реченням. Реченням вважається довільна послідовність слів, яка обмежена крапкою, після якої йде рядкова літера або кінець файлу. Попередньо кожне речення слід переформатувати так, щоб воно починалося з окремого рядка. Виконати один з варіантів завдань роботи з текстовими файлами.

1. Редактор тексту з командами додавання і видалення речень, з прокруткою тексту в обох напрямках. (+бонус)

# **Зовнішні специфікації програми (вхідні та вихідні дані, функціональні вимоги до програми).**

* 1. Вхідні дані:

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| Адреса файла | fileInName | Рядок довжиною меньше 255 символів з дійсною адресою текстового файла. | D:\test.txt |
| Вибір пункта меню | choice | Ціле число від 1 до 6 включно | 2 |
| Нове речення | sentence | Масив символів довжиною до 255 символів закінчується на ‘.’. | Test. |

Також вхідними данними треба вважати файл за адресою fileInName.

* 1. Вихідні дані:

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| Зчитанне речення | sentence | Масив символів довжиною до 255 символів закінчується на ‘.’. | Test. |

Також меню, повідомлення про помилки та ін.

* 1. Функціональні вимоги:
* імена файлів ввести з клавіатури.
* передбачити перевірку всіх операцій роботи з файлами (відкриття, читання, запису).
* управління виконанням програми здійснити на основі текстового меню користувача.
* передбачити обробку необмеженої кількості файлів.
* результати роботи програми вивести у файл та продублювати на екрані.

# **Модульна структура програми.**

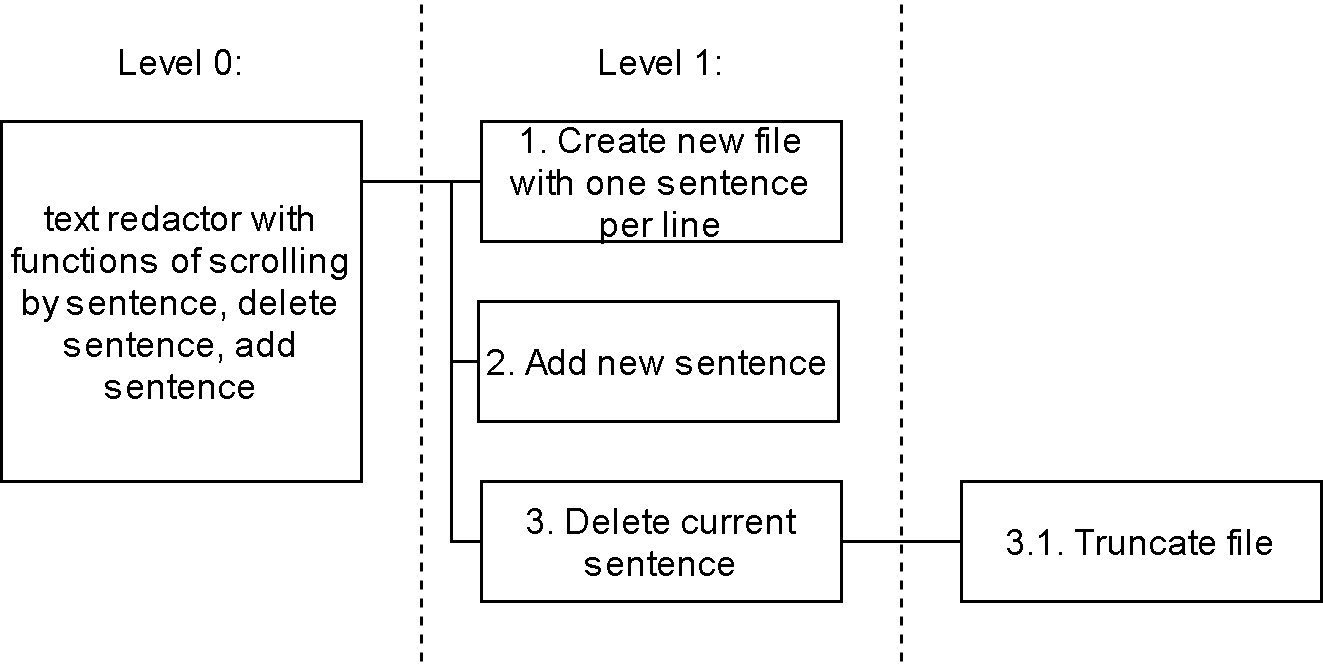


Рисунок 1

# **Алгоритми для розв’язання завдання всіх модулів програми.**

Рисунок 2

# 

Рисунок 3

# **Набори тестів для перевірки правильності виконання всіх модулів програми.**

Для усіх тестів текстовий файл однаковий:

“Aethe was the best archer, and no one could beat him. He \n

had a student, called Rethe, who was almost as skilled as \n

him. When he founded the first school, and Rethe was his \n

right hand. But one day they argued and arranged a fight to \n

determine by virtue of skill who deserved to lead the school. \n

Rethe came unarmed, so Aethe shot an arrow at her heart, \n

mortally wounding her. Rethe didn't die immediately. In the \n

three days it took her to succumb to her wounds, she told \n

Aethe 99 stories, started to go to sleep, and said that the last \n

one was for when she woke up, but she died in her sleep. \n

Upon her death, Aethe used these stories as the base for the \n

Lethani.”

Також додаватися буде речення “Test.”

|  |  |  |  |
| --- | --- | --- | --- |
| № | Назва | Вхідні Дані | Очікуваний результат |
| 1 | Додавання речення перед першим | Додати речення перед першим. | Test.\n  Aethe was the best archer, and no one could beat him.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n |
| 2 | Додавання речення після останнього | Додати речення після останнього. | Aethe was the best archer, and no one could beat him.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n  Test.\n |
| 3 | Додавання речення всередині | Додати речення після шостого.(або перед сьомим) | Aethe was the best archer, and no one could beat him.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  Test.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n |
| 4 | Комбінація перших трьох тестів | Додати речення перед першим.  Додати речення після сьомого(або перед восьмим).  Додати речення після десятого. | Test.\n  Aethe was the best archer, and no one could beat him.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  Test.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n  Test.\n |
| 5 | Видалення першого речення | Видалити перше речення. | He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n |
| 6 | Видалення останнього речення | Видалити восьме речення. | Aethe was the best archer, and no one could beat him.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n |
| 7 | Видалення речення з середини | Видалити шосте речення. | Aethe was the best archer, and no one could beat him.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n |
| 8 | Комбінація тестів 5-7 | Видалити перше речення.  Видалити п’яте речення.  Видалити шосте речення. | He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n |
| 9 | Комбінація тесту 4 та 8(спочатку додавання, потім видалення) | Додати речення перед першим.  Додати речення після сьомого(або перед восьмим).  Додати речення після десятого.  Видалити перше речення.  Видалити сьоме речення.  Видалити дев’яте речення. | Aethe was the best archer, and no one could beat him.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n |
| 10 | Тест 9 у зворотному порядку | Видалити перше речення.  Видалити п’яте речення.  Видалити шосте речення.  Додати речення перед першим.  Додати речення після п’ятого.(або перед шостим)  Додати речення після сьомого. | Test.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  But one day they argued and arranged a fight to determine by virtue of skill who deserved to lead the school.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Test.\n  In the three days it took her to succumb to her wounds, she told Aethe 99 stories, started to go to sleep, and said that the last one was for when she woke up, but she died in her sleep.\n  Test.\n |
| 11 | Випадкова комбінація додавань та видалень. | Додати речення після першого.  Додати речення перед дев’ятим.  Видалити п’яте речення.  Додати речення після восьмого.  Видалити сьоме речення. | Aethe was the best archer, and no one could beat him.\n  Test.\n  He had a student, called Rethe, who was almost as skilled as him.\n  When he founded the first school, and Rethe was his right hand.\n  Rethe came unarmed, so Aethe shot an arrow at her heart, mortally wounding her.\n  Rethe didn't die immediately.\n  Test.\n  Test.\n  Upon her death, Aethe used these stories as the base for the Lethani.\n |

# **Текст програми (всі файли проекту).**

“main.cpp”

#include <cstdio>

#include <iostream>

#include "txtProcessing.h"

#include <vector>

#include <algorithm>

int main() {

std::cout << "This application is the text redactor that opens \*.txt files." << std::endl;

std::cout << "Makes a copy of that file formatted so that one line contains one full sentence." << std::endl;

std::cout << "It can scroll one sentence at the time, add new sentences and delete them." << std::endl;

system("pause");

system("cls");

errno\_t err;

FILE\* txtOLD, \* txtNEW;

char fileInName[256];//D:\1uno1\diit\year1\year1\_semester2\WIP\OP\_Lr8\_almost\OP\_LR8\test.txt

bool exit;

do {

do {

std::fill\_n(fileInName, 256, 0);//0 = '\0'

std::cout << "input address of file(or only name if it is in the same folder as programm) starting with disk:\\ and ending with \*.txt: " << std::endl;

std::cin.getline(fileInName, 255);

err = fopen\_s(&txtOLD, fileInName, "r+");//old file

if (err)

std::cout << "Can't open that file!" << std::endl;

} while (err);

char fileOutName[256];

err = tmpnam\_s(fileOutName, 255);

if (!err) {

err = fopen\_s(&txtNEW, fileOutName, "w+");//resulting file

if (!err) {

copySentenceLines(txtOLD, txtNEW);

std::vector<fpos\_t> pointers{ 0 };

int index = 0;

int choice;

do {

system("cls");

coutSentence(txtNEW, pointers, index);

std::cout << std::endl;

std::cout << "1. Insert new sentence before current." << std::endl;

std::cout << "2. Insert new sentence after current." << std::endl;

if (index < pointers.size() - 1)//if not last sentence in file

std::cout << "3. Scroll to next sentence." << std::endl;

if (index > 0)//if not 1st sentence in file

std::cout << "4. Scroll to previous sentence." << std::endl;

std::cout << "5. Delete this sentence" << std::endl;

std::cout << "6. End work with file." << std::endl;

std::cin >> choice;

switch (choice) {

case 1: {

char sentence[256];

std::fill\_n(fileInName, 256, '\0');

std::cin.get();//get enter

std::cout << "Input new sentence: " << std::endl;

std::cin.getline(sentence, 255);

int i;

for (i = 0; i < 256 && sentence[i] != '\0'; i++) {}

insertSentence(txtNEW, sentence, pointers, index, i + 2);

break;

}

case 2: {

char sentence[256];

std::fill\_n(fileInName, 256, '\0');

std::cin.get();//get enter

std::cout << "Input new sentence: " << std::endl;

std::cin.getline(sentence, 255);

int i;

for (i = 0; i < 256 && sentence[i] != '\0'; i++) {}

if (index == pointers.size() - 2) {

index++;

coutSentence(txtNEW, pointers, index);

insertSentence(txtNEW, sentence, pointers, index, i + 2);

index--;

}

else

insertSentence(txtNEW, sentence, pointers, index + 1, i + 2);

break;

}

case 3: {

if (index < pointers.size() - 1)

index++;

else {

std::cout << "wrong input";

system("pause");

}

break;

}

case 4: {

if (index > 0)

index--;

else {

std::cout << "wrong input";

system("pause");

}

break;

}

case 5: {

fclose(txtNEW);

err = deleteSentence(fileOutName, pointers, index);

if (!err)

err = fopen\_s(&txtNEW, fileOutName, "r+");

}

case 6:

break;

default: {

std::cout << "wrong input";

system("pause");

break;

}

}

} while (choice != 6 && !err);

}

}

if (err)

std::cout << "Error occured." << std::endl;

else

std::cout << "outputFile: " << fileOutName << std::endl;

\_fcloseall();

char chExit;

std::cout << "Do you want to exit the programm?(Y/N): ";

std::cin >> chExit;

while ((chExit != 'Y' && chExit != 'y' && chExit != 'N' && chExit != 'n') || (std::cin.peek() != '\n')) {//until correct input

while (std::cin.get() != '\n');//clear istream

std::cout << "wrong input, try again." << std::endl;

std::cout << "Do you want to exit the programm?(Y/N): ";

std::cin >> chExit;

}

if (chExit == 'Y' || chExit == 'y')

exit = true;

else

exit = false;

std::cin.get();//enter

} while (!exit);

system("pause");

return 0;

}

“txtProcessing.h”

#ifndef \_\_TXTPROCESSING\_H\_\_

#define \_\_TXTPROCESSING\_H\_\_

#include <cstdio>

#include <iostream>

#include <vector>

void insertSentence(FILE\* txt, char\* sentence, std::vector<fpos\_t>& pointers, int index, int sentenceSize);

//appends sentence from txtIn to txtOut

void copySentenceLines(FILE\* txtIn, FILE\* txtOut);

errno\_t deleteSentence(const char\* fileName, std::vector<fpos\_t>& pointers, int& index);

void coutSentence(FILE\* txt, std::vector<fpos\_t>& pointers, int index);

errno\_t fresize(const char\* fileName, size\_t newSize);

#endif

“txtProcessing.cpp”

#define \_CRT\_SECURE\_NO\_WARNINGS

#include "txtProcessing.h"

#include <string.h>

void copySentenceLines(FILE\* txtIn, FILE\* txtOut) {

std::vector<fpos\_t> pointers{ 0 };

int index = 0;

do {//copy string-sentences to new file

fsetpos(txtIn, &(pointers[index]));

char current;

char sentence[256];

std::fill\_n(sentence, 256, '\0');//set all aray values to '/0'

int i = 0;//iterator

do {

current = fgetc(txtIn);

if (current != '\n') {

sentence[i] = current;

i++;

}

} while ((current != '.' && current != '!' && current != '?') || feof(txtIn));//check until the end of file or sentence

while ((current == '\n' || current == ' ' || current == '.' || current == '!' || current == '?') && !feof(txtIn))//skip dot-space sequence and check if current sentence is last

current = fgetc(txtIn);

fpos\_t temp;

if (!feof(txtIn)) {

ungetc(current, txtIn);//set from 2nd to 1st char of sentence

fgetpos(txtIn, &temp);//save pointer to start of next sentence

pointers.push\_back(temp);

}

fputs(sentence, txtOut);//adding new sentence

fputc('\n', txtOut);

index++;

} while (index < pointers.size());

}

void coutSentence(FILE\* txt, std::vector<fpos\_t>& pointers, int index) {

fsetpos(txt, &(pointers[index]));

char sentence[256];

std::fill\_n(sentence, 256, '\0');//set all aray values to '\0'

fgets(sentence, 255, txt);

std::cout << sentence[0];

for (int i = 1; i < 255 && sentence[i] != '\n' && sentence[i - 1] != '.'; i++)//added check for '.' because else it prints next message on the curent if inserted after last

std::cout << sentence[i];

if (index == pointers.size() - 1) {//to prevent adding old indexes to vector

fpos\_t temp;

char current = fgetc(txt);

if (!feof(txt)) {

ungetc(current, txt);//set from 2nd to 1st char of sentence

fgetpos(txt, &temp);//save pointer to start of next sentence

pointers.push\_back(temp);

}

}

}

void insertSentence(FILE\* txt, char\* sentence, std::vector<fpos\_t>& pointers, int index, int sentenceSize) {

//move anything after index to right N characters

fseek(txt, 0, SEEK\_END);

fpos\_t end = ftell(txt);

end--;//to avoid EOF

fpos\_t insertPos;

if (index <= pointers.size() - 1)//fix if after last

insertPos = pointers[index];

else

insertPos = end;

pointers.push\_back(insertPos);

for (int i = pointers.size() - 1; i > index; i--)

pointers[i] = pointers[i - 1] + sentenceSize;//updating pointers to sentences

char currentChar;

for (end; end >= insertPos; end--) {

fsetpos(txt, &end);

currentChar = fgetc(txt);

end += sentenceSize;

if (currentChar == '\n')//otherwise overrites previous symbol and writes '\n' twice

end--;

fsetpos(txt, &end);

fputc(currentChar, txt);

end -= sentenceSize;

}

fsetpos(txt, &insertPos);

fputs(sentence, txt);//??

if (index < pointers.size() - 1)//weirdo '\n'

fputc('\n', txt);

}

errno\_t deleteSentence(const char\* fileName, std::vector<fpos\_t>& pointers, int& index) {

FILE\* txt;

errno\_t err = fopen\_s(&txt, fileName, "r+");

if (!err) {

fseek(txt, 0, SEEK\_END);

fpos\_t end = ftell(txt);

int sentenceSize;

if (index < (pointers.size() - 1))//if last

sentenceSize = pointers[index + 1] - pointers[index];

else

sentenceSize = end - pointers[index];

//move anything after index to left N characters

fsetpos(txt, &pointers[index]);

fpos\_t start = ftell(txt);

char currentChar;

for (start; start < end - sentenceSize; start++) {

start += sentenceSize;

fsetpos(txt, &start);

currentChar = fgetc(txt);

start -= sentenceSize;

fsetpos(txt, &start);

fputc(currentChar, txt);

if (currentChar == '\n')//otherwise overrites previous symbol and writes '\n' twice

start++;

}

for (int i = index; i < pointers.size() - 1; i++)//mb start from index + 1

pointers[i] = pointers[i + 1] - sentenceSize;//updating pointers to sentences

pointers.pop\_back();

fclose(txt);

err = fresize(fileName, end - sentenceSize);

if (index >= pointers.size())//if deleting last

index--;

}

return err;

}

errno\_t fresize(const char\* fileName, size\_t newSize) {

FILE\* oldFile, \* newFile;

errno\_t err = fopen\_s(&oldFile, fileName, "r+");

if (!err && oldFile != NULL) {

fseek(oldFile, 0, SEEK\_END);

size\_t oldSize = ftell(oldFile);

fseek(oldFile, 0, SEEK\_SET);

if (oldSize > newSize) {

char tmpFileName[256];

err = tmpnam\_s(tmpFileName, 255);

if (!err) {

err = fopen\_s(&newFile, tmpFileName, "w+");

if (!err && newFile != NULL) {

char currentCh;

for (newSize; newSize > 0; newSize--) {

currentCh = fgetc(oldFile);

fputc(currentCh, newFile);

if (currentCh == '\n')//weirdo '\n'

newSize--;

}

fclose(oldFile);

fclose(newFile);

remove(fileName);

err = rename(tmpFileName, fileName);

}

if (newFile == NULL)

err = EINVAL;

}

}

else if (oldSize < newSize) {

newSize -= oldSize;

for (newSize; newSize > 0; newSize--)

fputc('\0', oldFile);

}

}

if (oldFile == NULL)

err = EINVAL;

return err;

}

# **Результати тестування програми та їх аналіз.**

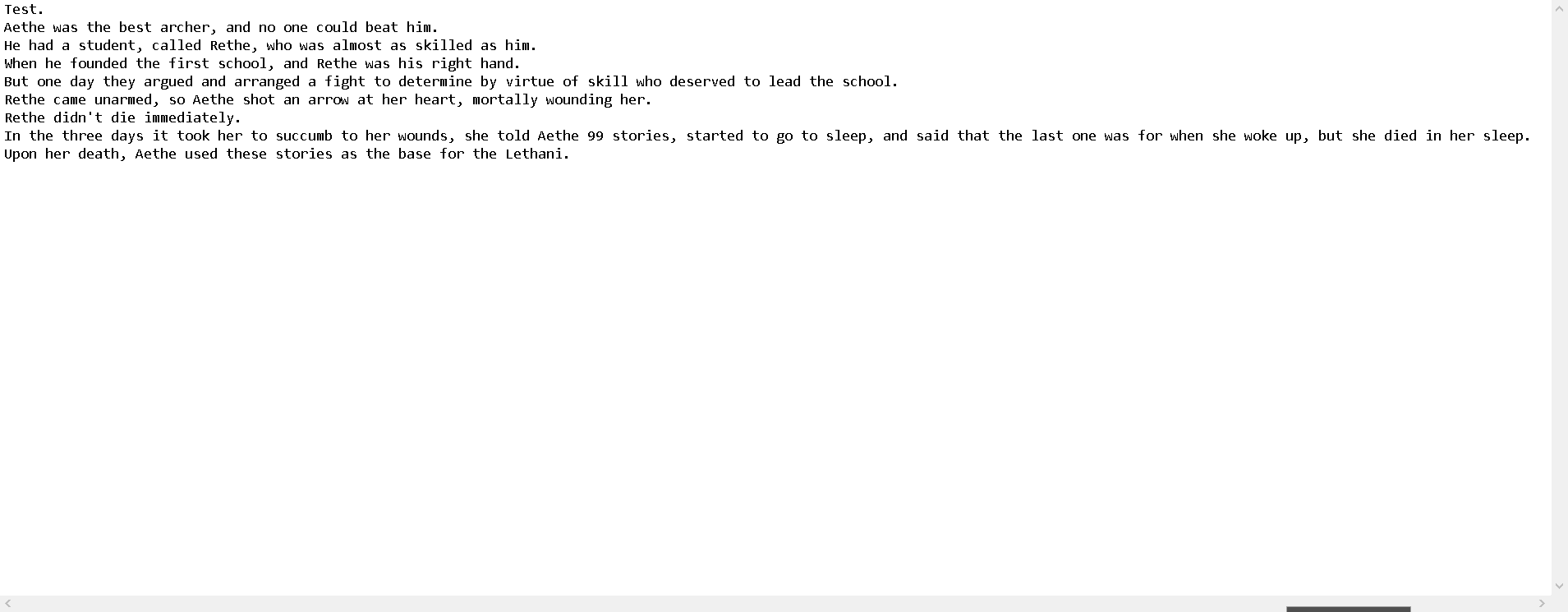


Рисунок 4(Тест 1)

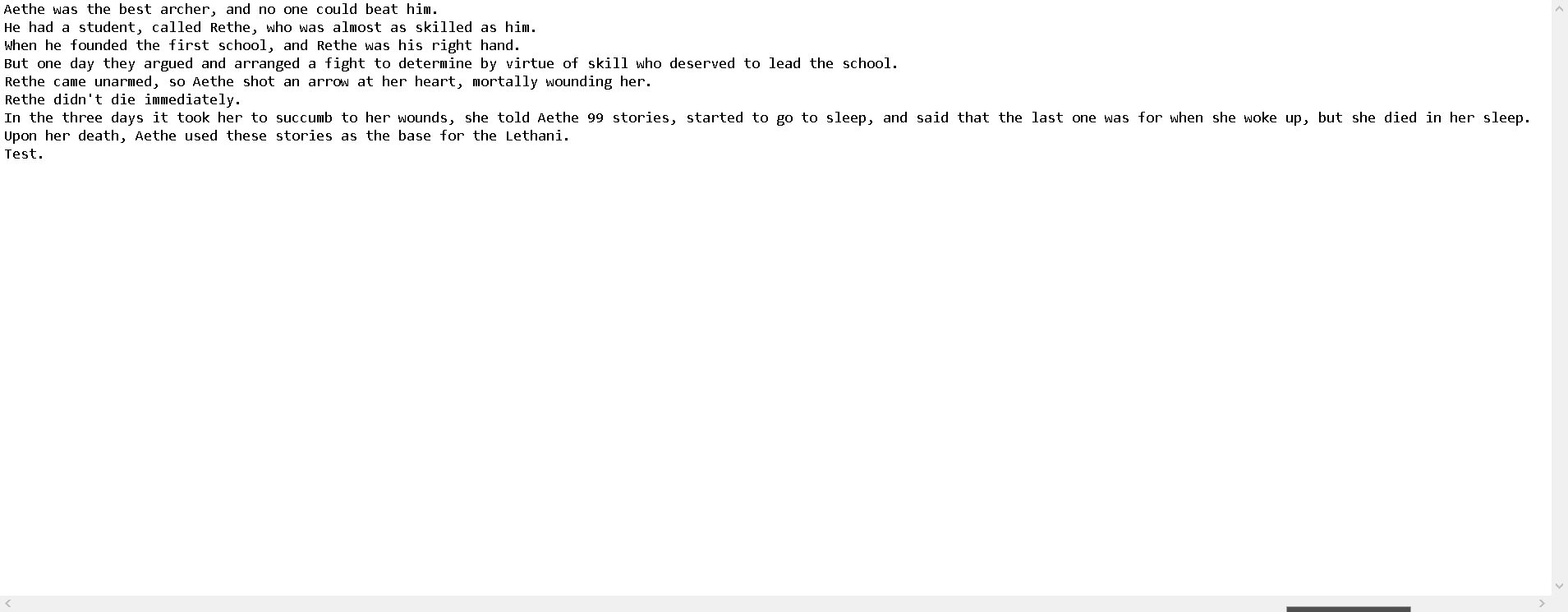


Рисунок 5(Тест 2)

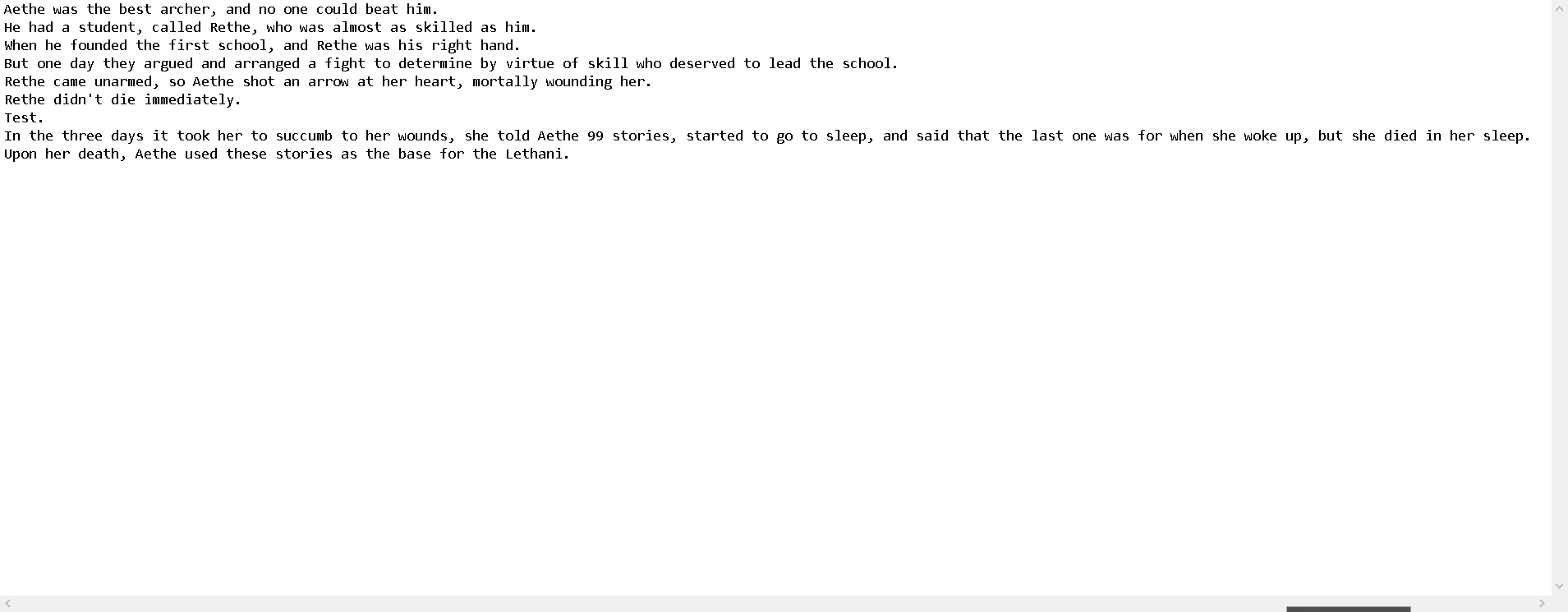


Рисунок 6(Тест 3)

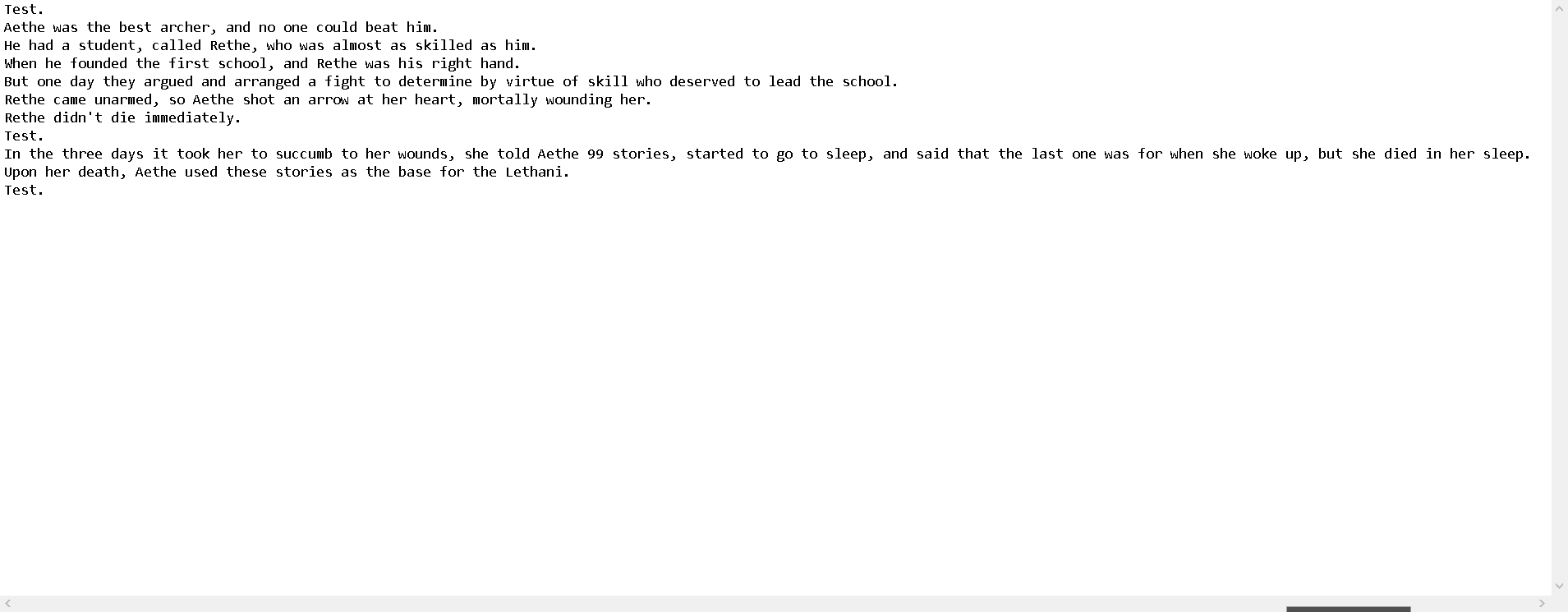


Рисунок 7(Тест 4)

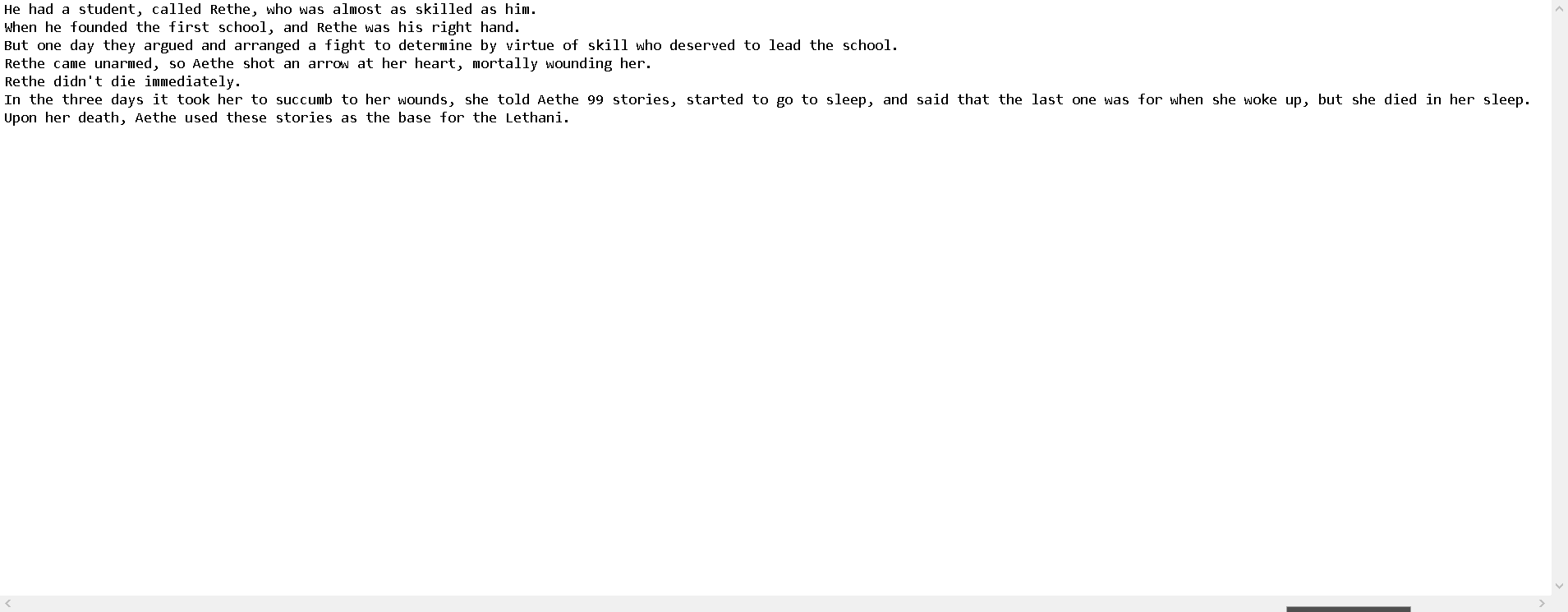


Рисунок 8(Тест 5)

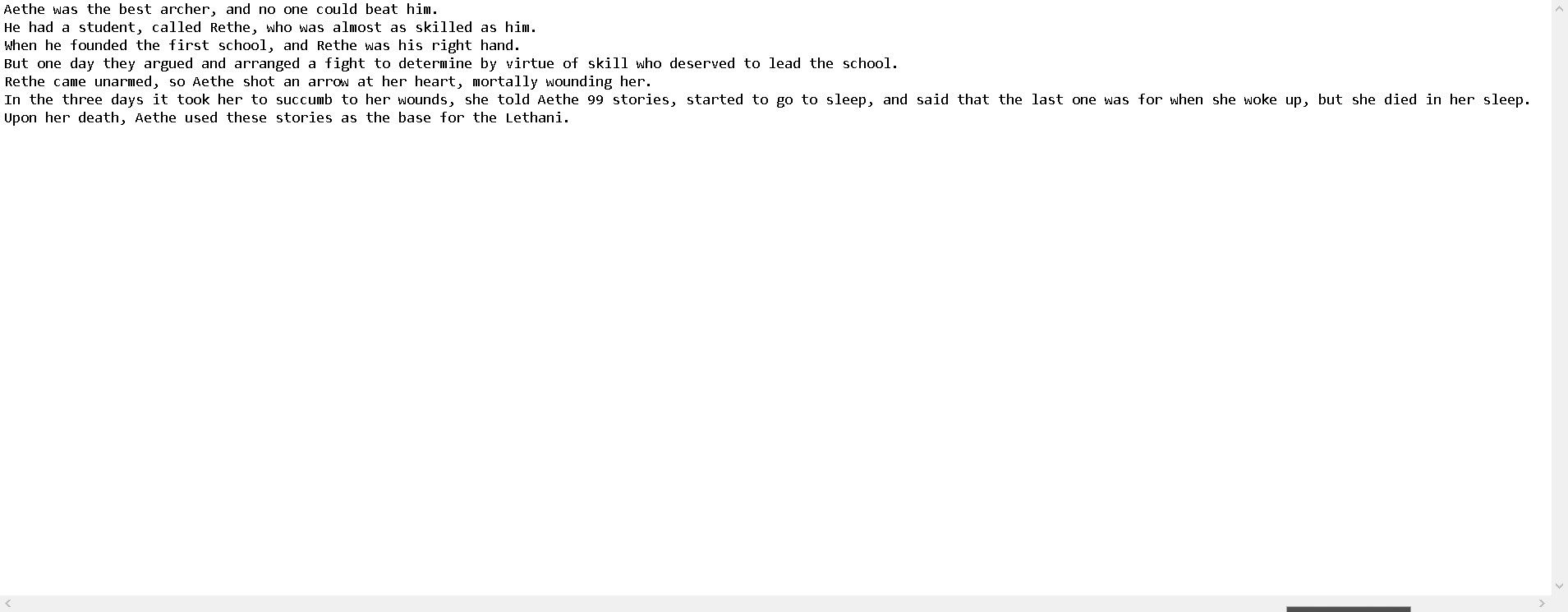


Рисунок 9(Тест 6)

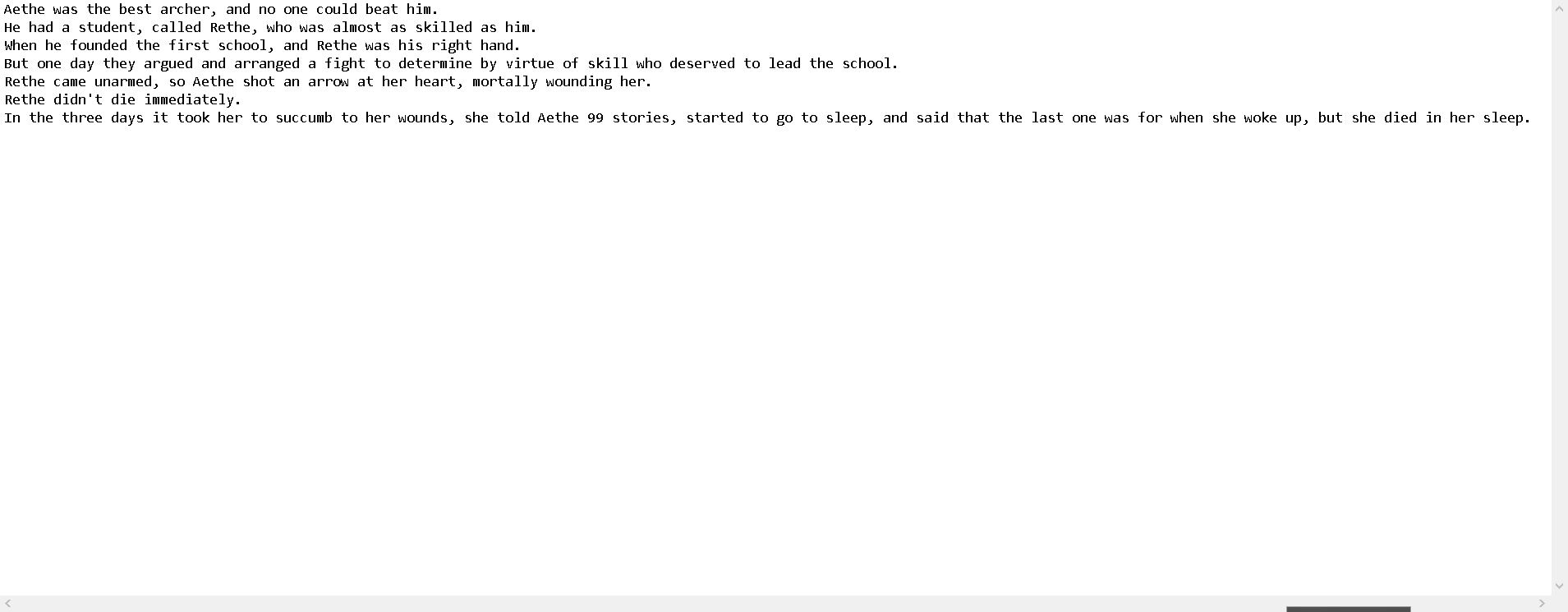


Рисунок 10(Тест 7)

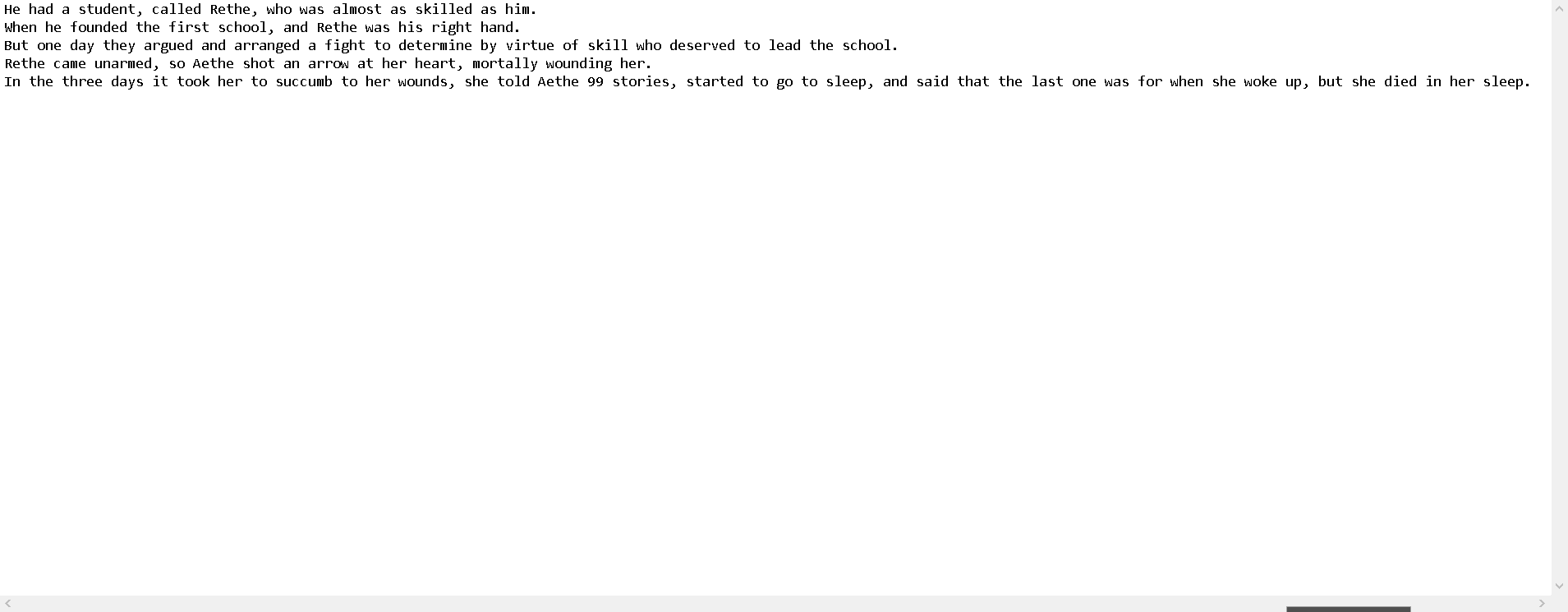


Рисунок 11(Тест 8)

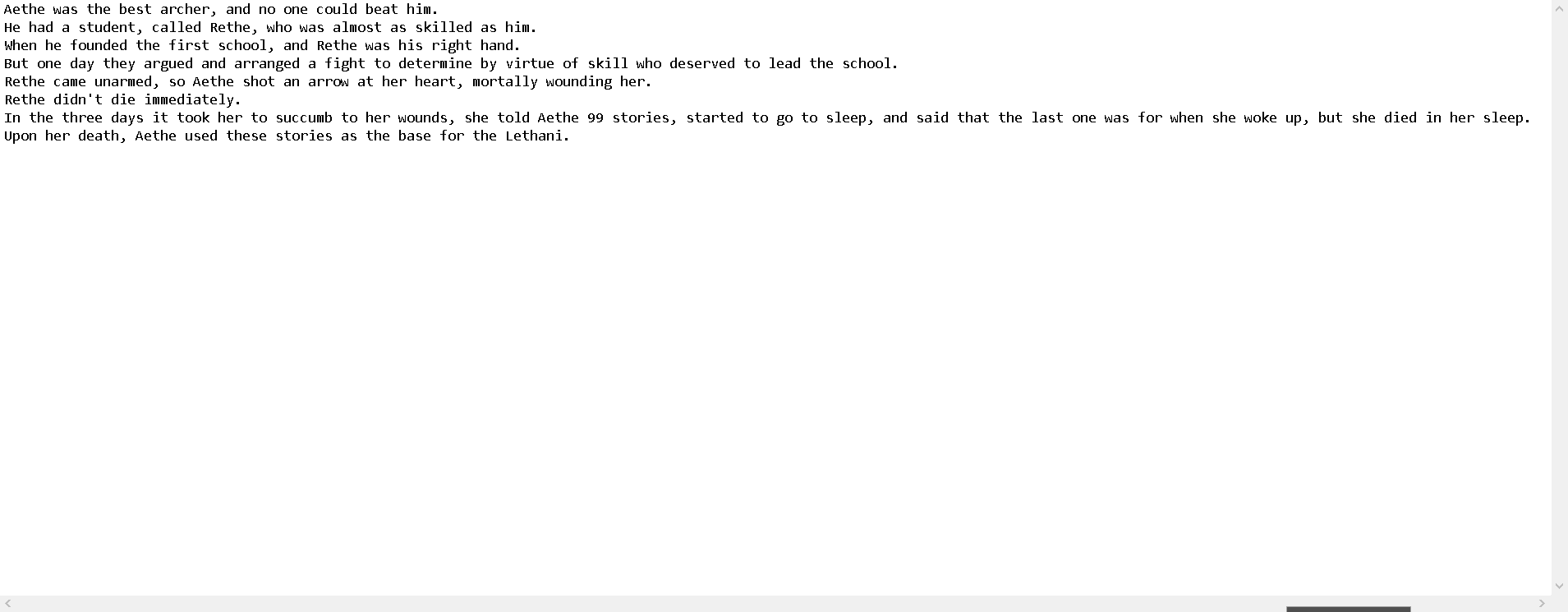


Рисунок 12(Тест 9)

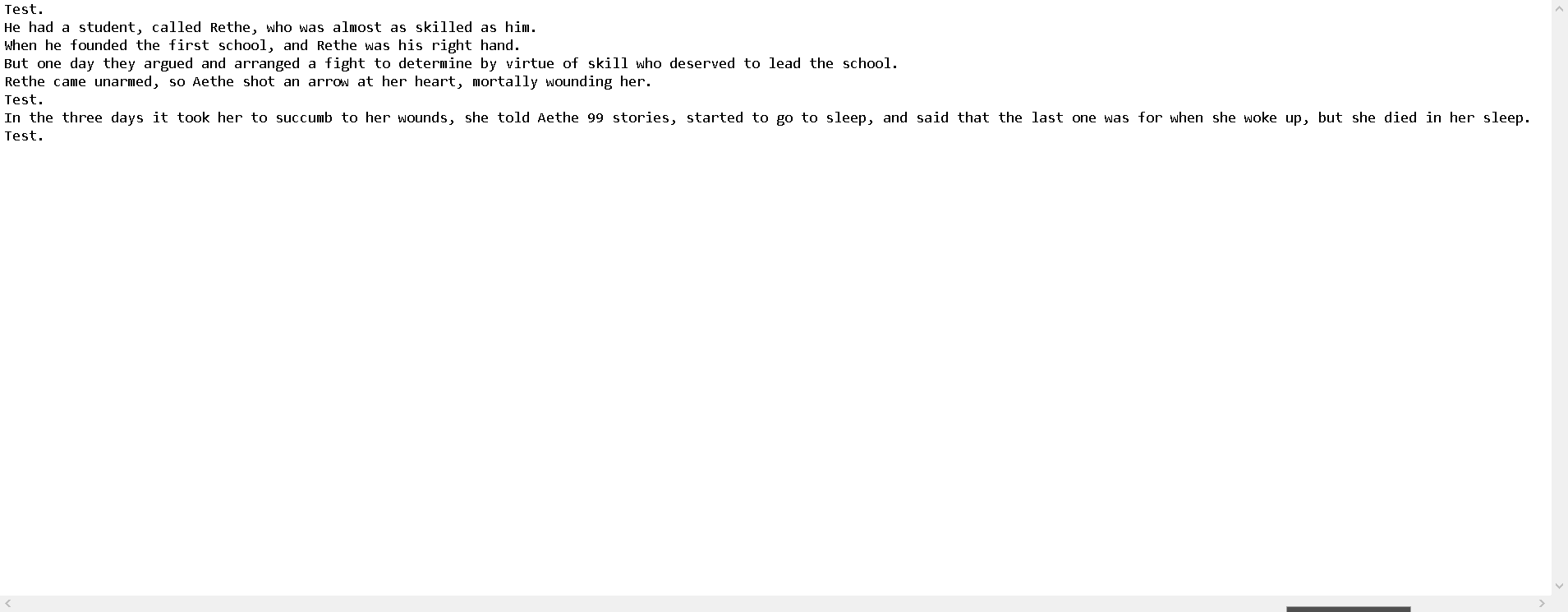


Рисунок 13(Тест 10)

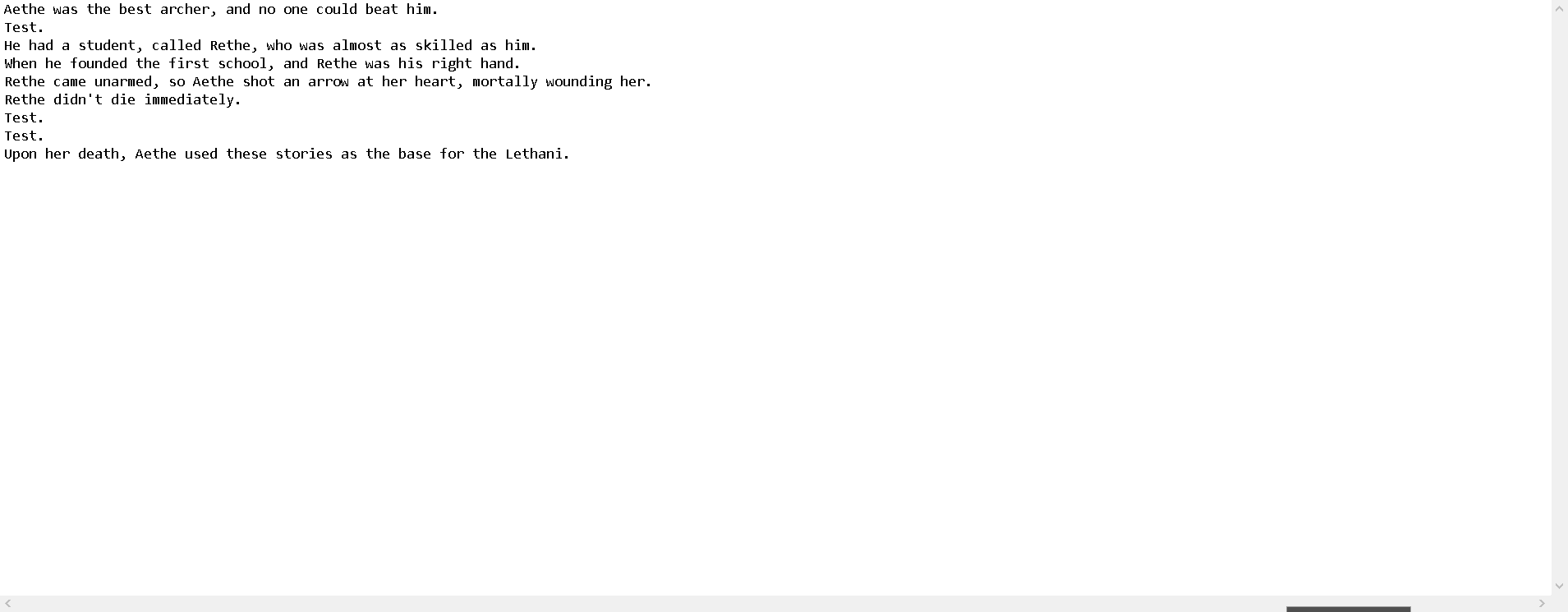


Рисунок 14(Тест 11)

Програма працює коректно, результати відповідають очікуванням.

# **Висновки щодо роботи з текстовими файлами і обробки символьних рядків.**

Роботу з текстовим файлом можна порівняти із роботою з масивом символів. Але є деякі відмінності, по перше файл зберігається не в оперативній пам’яті, а на зовнішньому накопичувачі. По друге, щоб звернутися до н-ного байту потрібно встановити внутрішній вказівник на потрібний байт и тоді зчитати/помістити значення. Також є різні режими відкриття файлів:

R – відкрити існуючий файл для зчитування.

W – відкрити файл для додавання інформації, якщо файл існує його зміст видаляється і розмір обнуляється.

A – відкрити файл для додавання інформації, уся інформація додається на кінці файлу, подовжуючи його. Функції перестановки вказівника не працюють.

R+ – відкрити існуючий файл для зчитування та додавання.

W+ – відкрити файл для зчитування та додавання, якщо файл існує його зміст видаляється і розмір обнуляється.

A+ – відкрити файл для додавання інформації, уся інформація додається на кінці файлу, подовжуючи його. Функції перестановки вказівника працюють, але після операцій додавання вказівник встановлюється на кінець файла.

Також існують додаткові модифікатори:

B – відкрити файл, як бінарний.

X – для модифікаторів заснованих на W, функція повертає помилку, якщо файл існує.

Також є деяка особливість: послідовність \r\n читається як \n, і навпаки \n записується як \r\n.