**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное

учреждение высшего образования

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Инженерная школа информационных технологий и робототехники

Отделение автоматизации и робототехники

Направление мехатроника и робототехника

Отчет

по лабораторной работе №4

по дисциплине

**«**Основы программирования и алгоритмизации**»**

**Строки в C++**

Выполнил:

Студент группы 8E21 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.С. Моисеев

Проверил:

Ассистент ОАР ИШИТР \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Я.О. Кургинов

Томск 2023

Вариант 10

Цель работы:

Поработать со строками в C++.

# Задание 1

Найдите количество вхождения 'aba' в строку.

**Блок схема** (Рисунок 1)

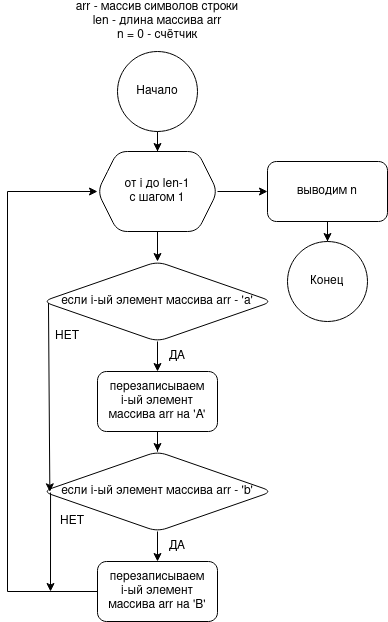


Рисунок 1 - Блок-схема к заданию 1.

Текст программы (Листинг 1, Приложение А)

**Результаты работы** (Рисунок 2)

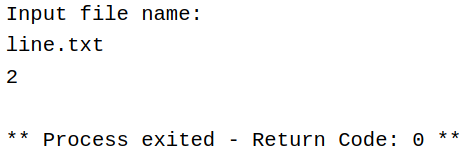
****

Рисунок 2 - Скриншот результата работы программы к заданию 1 на консоли.

Задание 2:

Дана строка. Заменить все символы 'a' и 'b' на 'A' и 'B' соответственно.

Блок схема (Рисунок 3)

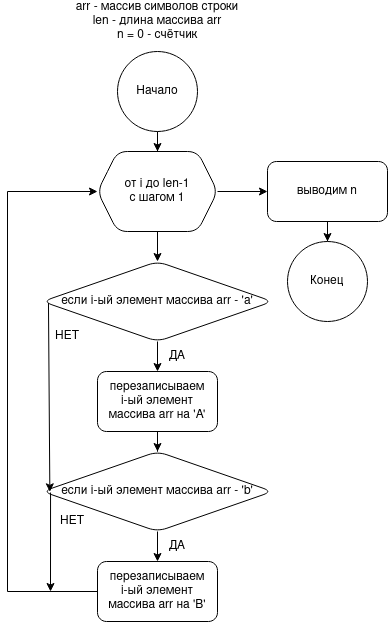


Рисунок 3 - Блок-схема к заданию 2.

Текст программы (Листинг 2, Приложение А)

**Результаты работы** (Рисунок 4)

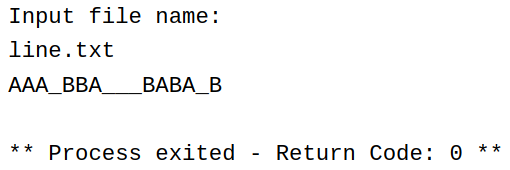


Рисунок 4 - Скриншот результата работы программы к заданию 2 на консоли.

Задание 3

В данной строке вставить после каждогосимвола 'a' символ 'b'.

Блок схема (Рисунок 5)

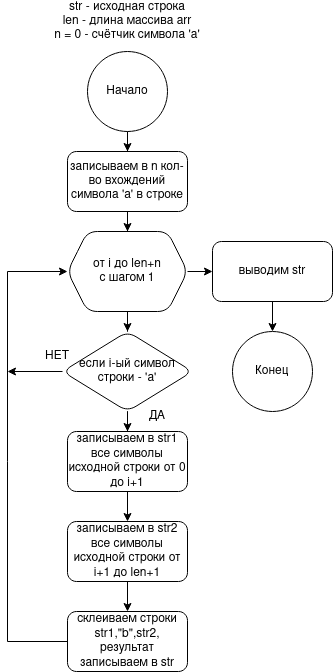


Рисунок 5 - Блок-схема к заданию 3.

Текст программы (Листинг 3, Приложение А)

**Результаты работы** (Рисунок 6)

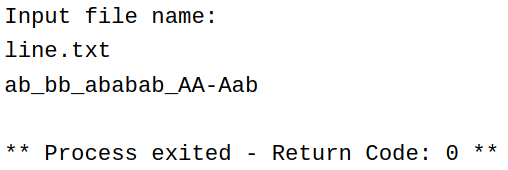


Рисунок 6 - Скриншот результата работы программы к заданию 3 на консоли.

**Задание 4**

Вывести в алфавитном порядке все слова, содержащие наибольшее  
количество гласных букв; найти все слова, в которые буква «а» входит  
не менее двух раз.

**Блок схема** (Рисунок 7)

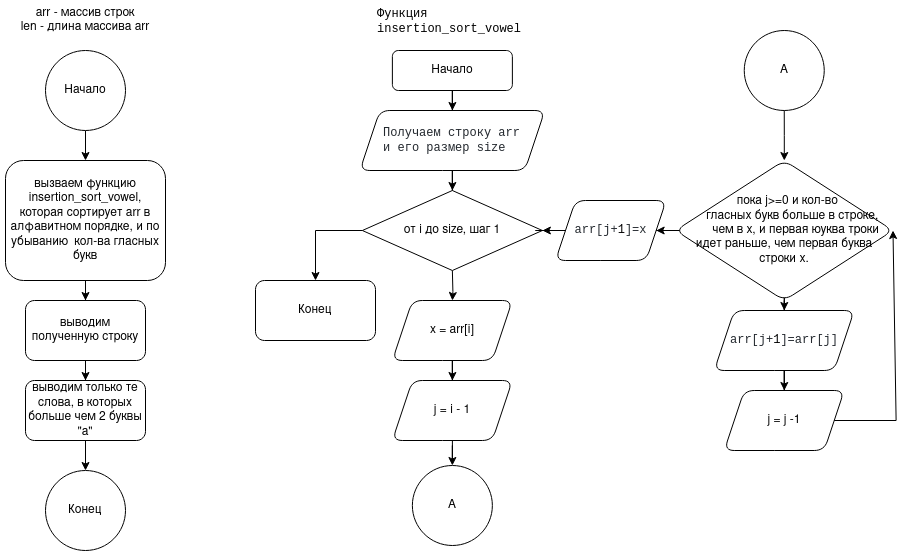


Рисунок 7 - Блок-схема к заданию 4.

Текст программы(Листинг 4, Приложение А)

**Результаты работы** (Рисунок 8)

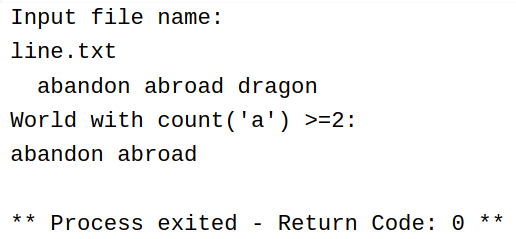


Рисунок 8 - Скриншот результата работы программы к заданию 4 на консоли.

**Задание 5**

Строка состоит из слов, разделенных одним или несколькими пробелами. Переставьте слова по убыванию их длин.

**Блок схема** (Рисунок 9)

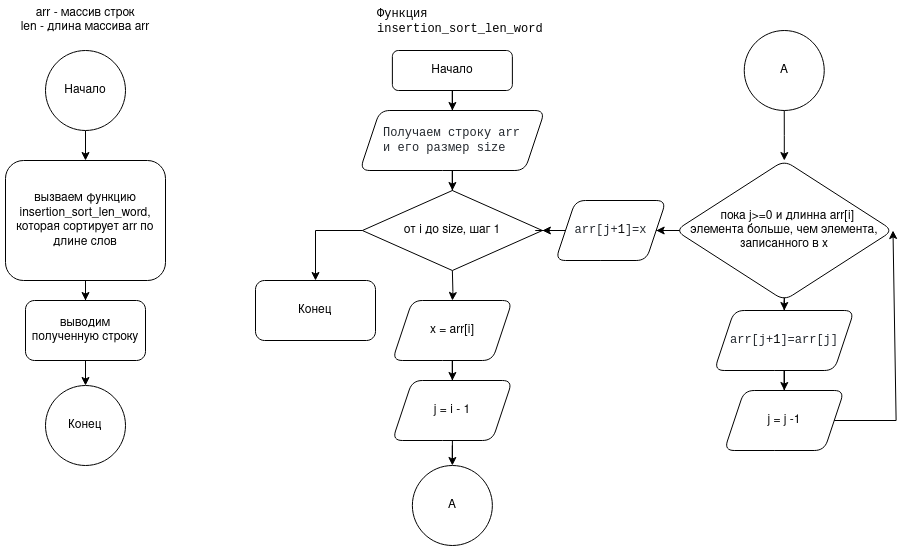
****

Рисунок 9 - Блок-схема к заданию 5.

Текст программы(Листинг 5, Приложение А)

**Результаты работы** (Рисунок 10)

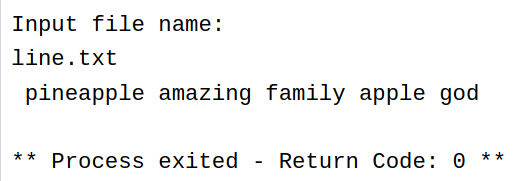
****

Рисунок 10 - Скриншот результата работы программы к заданию 5 на консоли.

**Задание 6**

Дан текст. Некоторые его фрагменты выделены группами символов ##.  
Заменить выделение группами символов '<' и '>)'. Пример: 'Это  
##тестовый пример## для задачи ##на## строки' преобразуется в 'Это  
<тестовый> пример для задачи <на> строки'.

**Блок схема** (Рисунок 11)

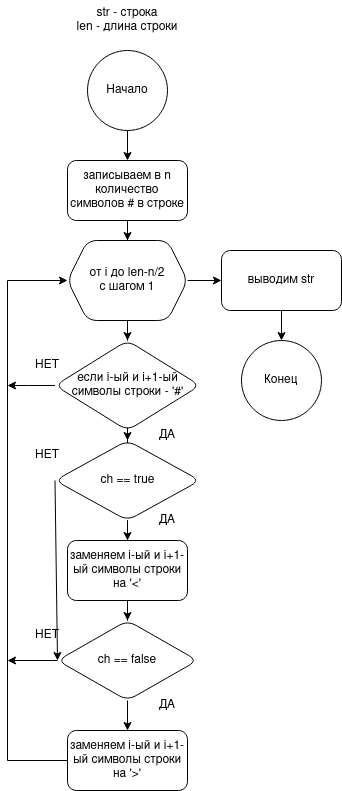
****

Рисунок 11 - Блок-схема к заданию 6.

Текст программы (Листинг 6, Приложение А)

**Результаты работы** (Рисунок 12)

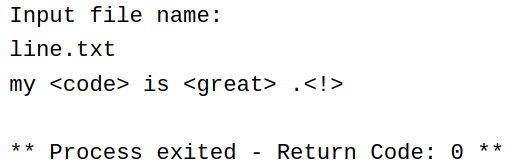
****

Рисунок 12 - Скриншот результата работы программы к заданию 6 на консоли.

**Задание 7**

Даны две строки s и w, вывести строку х максимальной длины,  
состоящую из букв, таких, что существует перестановка х, являющаяся  
подстрокой перестановки s и одновременно являющаяся подстрокой  
перестановки w.

**Блок схема** (Рисунок 13)

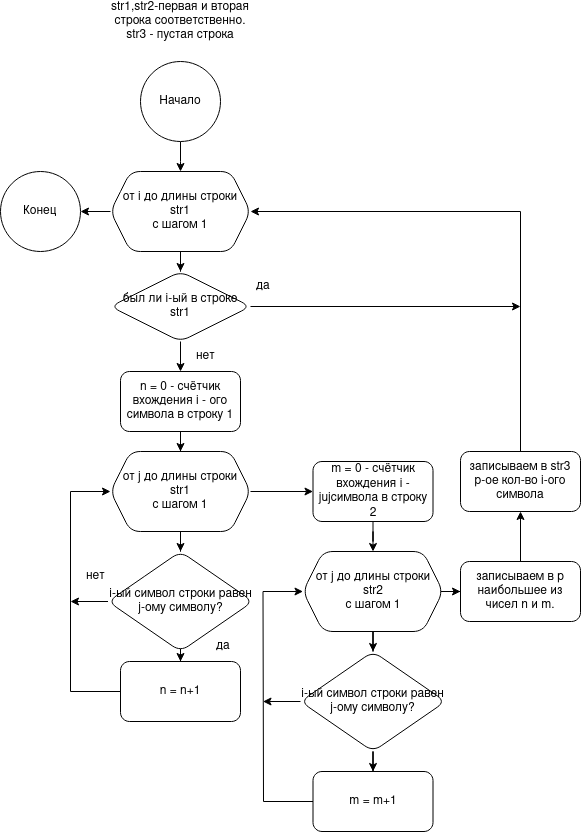
****

Рисунок 13 - Блок-схема к заданию 7.

Текст программы (Листинг 7, Приложение А)

**Результаты работы** (Рисунок 14)

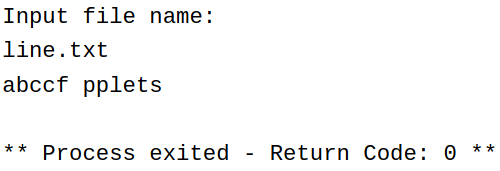
****

Рисунок 14 - Скриншот результата работы программы к заданию 7 на консоли.

**Приложение А**

Листинг 1 – Программа к заданию 1.

|  |
| --- |
| 1. #include <iostream> 2. #include <fstream> 3. int main() 4. { 5. std::string file\_name; std::cout << "Input file name: "; std::cin >> file\_name; 6. std::ifstream file; 7. file.open(file\_name); 8. if(file.is\_open()){ 9. std::string str; 10. getline(file,str); 11. const int len=str.length(); 12. char arr[len]; 13. for(int i=0;i<len;i++){ 14. arr[i]=str[i]; 15. } 16. int n = 0; 17. for (int i = 0; i <= len-2; i++) { 18. if (arr[i] == 'a' && arr[i + 1] == 'b' && arr[i + 2] == 'a') n++; 19. } 20. std::cout << n; 21. }else{ 22. std::cout << "ERROR" << std::endl; 23. } 24. return 0; 25. } |

Листинг 2 – Программа к заданию 2.

|  |
| --- |
| 1. #include <iostream> 2. #include <fstream> 3. int main() 4. { 5. std::string file\_name; std::cout << "Input file name: "; std::cin >> file\_name; 6. std::ifstream file; 7. file.open(file\_name); 8. if(file.is\_open()){ 9. std::string str; 10. getline(file,str); 11. const int len=str.length(); 12. char arr[len]; 13. for(int i=0;i<len;i++){ 14. arr[i]=str[i]; 15. } 16. int n = 0; 17. for (int i = 0; i <= len; i++) { 18. if (arr[i] == 'a') arr[i] = 'A'; 19. if (arr[i] == 'b') arr[i] = 'B'; 20. } 21. std::cout << arr; 22. }else{ 23. std::cout << "ERROR"<<std::endl; 24. } 25. return 0; 26. } |

Листинг 3 – Программа к заданию 3.

|  |
| --- |
| 1. #include <iostream> 2. #include <fstream> 3. int main() 4. { 5. std::string file\_name; std::cout << "Input file name: "; std::cin >> file\_name; 6. std::ifstream file; 7. file.open(file\_name); 8. if(file.is\_open()){ 9. std::string str; 10. getline(file,str); 11. const int len = str.length(); 12. int n = 0; 13. for (int i = 0; i < len; i++) { 14. if (str.at(i) == 'a') n++; 15. } 16. for(int i = 0; i < len+n; i++) { 17. if (str.at(i) == 'a') { 18. std::string str1(str, 0, i+1); 19. std::string str2(str, i+1, len+1); 20. str.clear(); 21. str.append(str1).append("b").append(str2); 22. } 23. } 24. std::cout << str; 25. } 26. file.close(); 27. return 0; 28. } |

Листинг 4 – Программа к заданию 4.

|  |
| --- |
| 1. #include <iostream> 2. #include <fstream> 3. #include <string> 4. int word\_count(std::string file\_name){ 5. std::ifstream file; 6. file.open(file\_name); 8. if(file.is\_open()){ 9. int count = 1; 10. std::string str; 11. getline(file,str); 13. for (int i = 0; i < str.size(); i++) 14. if (str[i] == ' ') count++; 16. file.close(); 17. return count; 18. } 19. else{ 20. file.close(); 21. } 22. return 0; 23. } 24. int vowel\_count(std::string str){ 25. int n=0; 26. for(int i=0; i<=str.length();i++){ 27. if(int(str[i])==97||int(str[i])==101||int(str[i])==105||int(str[i])==111||int(str[i])==117||int(str[i])==121) n++; 28. } 29. return n; 30. } 31. int a\_count(std::string str){ 32. int n=0; 33. for(int i=0; i<=str.length();i++){ 34. if(int(str[i])==97) n++; 35. } 36. return n; 37. } 38. void insertion\_sort\_vowel(std::string \*arr, int size){ 39. for(int i=0;i<size;i++){ 40. std::string x=arr[i]; 41. int j=i-1; 42. while(j>=0 && vowel\_count(arr[j])>=vowel\_count(x) && int(arr[j][0])<=int(x[0])){ 43. arr[j+1]=arr[j]; 44. j--; 45. } 46. arr[j+1]=x; 47. } 48. } 49. int main() 50. { 51. std::string file\_name; std::cout << "Input file name: "; std::cin >> file\_name; 52. int words = word\_count(file\_name); 54. std::string arr[words]; 56. std::ifstream file; 57. file.open(file\_name); 59. if(file.is\_open()){ 61. for(int i=0; i<=words;i++){ 62. file >> arr[i]; 63. } 65. insertion\_sort\_vowel(arr, words); 66. for(int i=words; i>=0;i--){ 67. std::cout << arr[i] << " "; 68. } 69. std::cout << std::endl; 70. std::cout << "World with count('a') >=2: " << std::endl; 71. for(int i=words; i>=0;i--){ 72. if(a\_count(arr[i])>=2) std::cout<<arr[i]<<" "; 73. } 74. }else{ 75. std::cout << "ERROR"; 76. } 78. return 0; 79. } |

Листинг 5 – Программа к заданию 5.

|  |
| --- |
| 1. #include <iostream> 2. #include <fstream> 3. #include <string> 4. int word\_count(std::string file\_name){ 5. std::ifstream file; 6. file.open(file\_name); 8. if(file.is\_open()){ 9. int count = 1; 10. std::string str; 11. getline(file,str); 13. for (int i = 0; i < str.size(); i++) 14. if (str[i] == ' ') count++; 16. file.close(); 17. return count; 18. } 19. else{ 20. file.close(); 21. } 22. return 0; 23. } 24. void insertion\_sort\_len\_word(std::string \*arr, int size){ 25. for(int i=0;i<size;i++){ 26. std::string x=arr[i]; 27. int j=i-1; 28. while(j>=0 && arr[j].length()>=x.length()){ 29. arr[j+1]=arr[j]; 30. j--; 31. } 32. arr[j+1]=x; 33. } 34. } 35. int main() 36. { 37. std::string file\_name; std::cout << "Input file name: "; std::cin >> file\_name; 38. int words = word\_count(file\_name); 40. std::string arr[words]; 42. std::ifstream file; 43. file.open(file\_name); 45. if(file.is\_open()){ 47. for(int i=0; i<words;i++){ 48. file >> arr[i]; 49. } 51. insertion\_sort\_len\_word(arr, words); 52. for(int i=words; i>=0;i--){ 53. std::cout << arr[i] << " "; 54. } 56. return 0; 57. }else{ 58. file.close(); 59. } 60. return 0; 61. } |

Листинг 6 – Программа к заданию 6.

|  |
| --- |
| 1. #include <iostream> 2. #include <fstream> 3. #include <string> 4. int main() 5. { 6. std::string file\_name; std::cout << "Input file name: "; std::cin >> file\_name; 8. std::ifstream file; 9. file.open(file\_name); 11. if(file.is\_open()){ 13. std::string str; 14. getline(file,str); 15. const int length = str.length(); 16. int n = 0; 17. for (int i = 0; i < length; i++) { 18. if (str.at(i) == '#') n++; 19. } 20. bool ch = false; 21. for(int i = 0; i < length-n/2; i++) { 22. if (str.at(i) == '#' && str.at(i+1) == '#') { 23. if(!ch){ 24. std::string str1(str, 0, i); 25. std::string str2(str, i+2, length+1); 26. str.clear(); 27. str.append(str1).append("<").append(str2); 28. ch = true; 29. continue; 30. } 31. if(ch){ 32. std::string str1(str, 0, i); 33. std::string str2(str, i+2, length+1); 34. str.clear(); 35. str.append(str1).append(">").append(str2); 36. ch = false; 37. continue; 38. } 39. } 40. } 42. std::cout << str; 43. }else{ 44. std::cout<<"ERROR"<<std::endl; 45. } 46. return 0; 47. } |

Листинг 7 – Программа к заданию 7.

|  |
| --- |
| 1. #include <iostream> 2. #include <fstream> 3. #include <string> 4. std::string str4 = ""; // символы которые уже были. 5. bool check\_symbol(char symbol, std::string str); 6. int main() 7. { 9. std::string file\_name; std::cout << "Input file name: "; std::cin >> file\_name; 10. std::ifstream file; 11. file.open(file\_name); 13. if(file.is\_open()){ 15. std::string str1; 16. std::string str2; 17. getline(file,str1); 18. getline(file,str2); 20. const int len1 = str1.length(); 21. const int len2 = str2.length(); 22. std::string str3 = ""; // конечная строка. 24. for(int i=0;i<len1;i++){ 25. std::string s(str1, i, 1); 26. if(check\_symbol(str1.at(i), str1)){//проверяем не был ли уже символ 27. str4.append(s); 28. int n = 0; // счётчик вхождения символа в строку 1. 29. for(int j=0;j<str1.length();j++){ 30. if(str1.at(i) == str1.at(j)) n++; 31. } 32. int m = 0; // счётчик вхождения символа в строку 2. 33. for(int j=0;j<str2.length();j++){ 34. if(str2.at(j) == str1.at(i)) m++; 35. } 36. int p = (n>m)? m:n; 37. for(int j=1;j<=p;j++) str3.append(s); 38. } 39. } 40. std::cout << str3; 41. }else{ 42. std::cout << "ERROR" << std::endl; 43. } 44. return 0; 45. } 46. bool check\_symbol(char symbol, std::string str){ 47. for(int i=0;i<str4.length();i++){ 48. if(str4.at(i) == symbol) return false; 49. } 50. return true; 51. } |