

**Федеральное государственное автономное образовательное учреждение  
высшего образования  
"Национальный исследовательский университет "Высшая школа экономики"**

Московский институт электроники и математики им. А.Н. Тихонова НИУ ВШЭ  
Департамент компьютерной инженерии

**Курс: Алгоритмизация и программирование**

Раздел	Макс. оценка	Итоговая оценка
Работа программы	1	
Тесты	1	
Правильность алгоритма	3	
Ответы на вопросы	2	
Дополнительное задание	3	
Итого	10	

**Отчет по лабораторной работе № 9**

**Студент: Чапайкин Арсений Георгиевич**

**Группа: БИВ242**

**Вариант: № 171 (4, 1, 2)**

**Руководитель: Елисеенко А.М.**

**Оценка:**

**Дата сдачи:**

# Содержание

Задание	2
Листинг программы	3
Распечатка тестов к программе и результатов	5

## Задание

1. Даны символьные строки (количество строк задает пользователь). Требуется выделить из каждой строки и напечатать подстроки ограниченные с обеих сторон одной или несколькими заглавными латинскими буквами.
2. Среди выделенных подстрок найти самую короткую подстроку.
3. Преобразовать исходную строку, которой принадлежит найденная подстрока, следующим образом: вставить три звездочки после первой цифры.

## Листинг программы

```
1 #include <iostream>
2 #include <stdexcept>
3 #include <string>
4 #include <vector>
5
6 bool is_capital_latin(char c) {
7     return 'A' <= c && c <= 'Z';
8 }
9
10 bool is_digit(char c) {
11     return '0' <= c && c <= '9';
12 }
13
14 std::vector<std::pair<size_t, std::string>> task1(const std::vector<std::string>&
    strings) {
15     std::vector<std::pair<size_t, std::string>> substrings;
16
17     for (size_t index = 0; index < strings.size(); index++) {
18         for (size_t i = 0; i < strings[index].size(); i++) {
19             std::string temp;
20             for (size_t j = i + 1; j < strings[index].size(); j++) {
21                 if (is_capital_latin(strings[index][i])
22                     && is_capital_latin(strings[index][j]) && i + 1 != j) {
23                     substrings.emplace_back(index, temp);
24                 }
25                 temp += strings[index][j];
26             }
27             temp += strings[index][i];
28         }
29     }
30
31     if (substrings.empty()) {
32         throw std::runtime_error("No viable substrings");
33     } else {
34         return substrings;
35     }
36 }
37
38 size_t task2(const std::vector<std::pair<size_t, std::string>>& substrings)
    noexcept {
39     size_t min = 0;
40     for (size_t i = 0; i < substrings.size(); i++) {
41         if (substrings[i].second.size() < substrings[min].second.size()) {
42             min = i;
43         }
44     }
45     std::cout << substrings[min].second << '\n';
46     return substrings[min].first;
47 }
48
49 void task3(size_t index, std::vector<std::string>& strings) {
50     std::string new_string = "";
51     bool digit_found = false;
52
53     for (size_t i = 0; i < strings[index].size(); i++) {
54         new_string += strings[index][i];
55         if (is_digit(strings[index][i]) && !digit_found) {
```

```

56         new_string += "***";
57         digit_found = true;
58     }
59 }
60
61 if (!digit_found) {
62     throw std::runtime_error("No digits found");
63 } else {
64     strings[index] = std::move(new_string);
65 }
66 }
67
68 int main() {
69     size_t n;
70     std::cin >> n;
71
72     std::vector<std::string> strings;
73     strings.assign(n, "");
74
75     for (size_t i = 0; i < n; i++) {
76         std::cin >> strings[i];
77     }
78
79     std::cout << "Executing Task1...\n";
80     std::vector<std::pair<size_t, std::string>> substrings;
81     try {
82         substrings = task1(strings);
83         std::cout << "Results:\n";
84         for (size_t i = 0; i < substrings.size(); i++) {
85             std::cout << substrings[i].second << '\n';
86         }
87     } catch (const std::runtime_error& error) {
88         std::cout << error.what() << '\n';
89         return 1;
90     }
91
92     std::cout << "Executing Task2...\n";
93     std::cout << "Results:\n";
94     size_t min = task2(substrings);
95
96     std::cout << "Executing Task3...\n";
97     try {
98         task3(min, strings);
99         std::cout << "Results:\n";
100        std::cout << strings[min] << '\n';
101    } catch (const std::runtime_error& error) {
102        std::cout << error.what() << '\n';
103        return 3;
104    }
105
106    return 0;
107 }

```

---

# Распечатка тестов к программе и результатов

## Задание 1

Номер	Исходные данные	Результат
1	1 aaaa	Executing Task1... No viable substrings
2	2 ABCD EFGH	Executing Task1... Results: B BC C F FG G Executing Task2... Results: B Executing Task3... No digits found
3	1 ABCD1234	Executing Task1... Results: B BC C Executing Task2... Results: B Executing Task3... ABCD1***234

Номер	Исходные данные	Результат
4	1 ABcD1234	Executing Task1... Results: Bc c Executing Task2... Results: c Executing Task3... ABcD1***234