



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ «Информатика и системы управления»

КАФЕДРА _____ «Теоретическая информатика и компьютерные технологии»

Лабораторная работа № 12
по курсу «Языки и методы программирования»
«Обработка текстовых файлов»

Студент группы ИУ9-22Б Тараканов В. Д.

Преподаватель Посевин Д. П.

Москва 2024

1 Задание

Найти все файлы с расширением «java» в указанном каталоге, определить для каждого файла номер строки, с которой начинается объявление расположенного в этом файле публичного класса, и сохранить полученную информацию в файле classes.txt в текущем каталоге. Каждая строка файла classes.txt должна содержать имя публичного класса и номер строки. Имена классов должны быть отсортированы лексикографически. Работоспособность программы нужно проверить на наборе предварительно подготовленных файлов с исходниками на языке Java.

2 Результаты

Листинг 1 — Начало кода

```
1  #include <iostream>
2  #include <fstream>
3  #include <filesystem>
4  #include <regex>
5  #include <string>
6  #include <vector>
7
8
9  namespace fs = std::filesystem;
10
11  struct ClassInfo {
12      std::string className;
13      int lineNumber;
14  };
15
16  bool findPublicClass(const std::string& filePath, ClassInfo&
classInfo) {
17      std::ifstream inputFile(filePath);
18      if (!inputFile.is_open()) {
19          std::cerr << "Failed to open file: " << filePath << std::endl;
20          return false;
21      }
22
23      std::regex classPattern(R"(\bpublic\s+class\s+(\w+)\s+)");
24      std::string line;
25      int lineNumber = 0;
26      while (std::getline(inputFile, line)) {
27          lineNumber++;
28          std::smatch match;
29          if (std::regex_search(line, match, classPattern)) {
30              classInfo.className = match[1];
31              classInfo.lineNumber = lineNumber;
32              inputFile.close();
33              return true;
34          }
35      }
36
37      inputFile.close();
38      return false;
39  }
40
41
```

Листинг 2 — продолжение кода из Листинга 1

```
1
2     void findAndProcessJavaFiles(const fs::path& inputDir, std::vector
<ClassInfo>& classes) {
3         for (const auto& entry : fs::directory_iterator(inputDir)) {
4             if (entry.is_regular_file() && entry.path().extension() == ".
java") {
5                 ClassInfo classInfo;
6                 if (findPublicClass(entry.path().string(), classInfo)) {
7                     classes.push_back(classInfo);
8                 }
9             }
10        }
11    }
12
13    void writeClassesToFile(const std::string& outputPath, const
std::vector<ClassInfo>& classes) {
14        std::ofstream outputFile(outputFilePath);
15        if (!outputFile.is_open()) {
16            std::cerr << "Failed to open output file: " << outputPath
<< std::endl;
17            return;
18        }
19
20        for (const auto& classInfo : classes) {
21            outputFile << classInfo.className << " " << classInfo.
lineNumber << std::endl;
22        }
23
24        outputFile.close();
25    }
26
27    int main() {
28        std::string inputDirectory;
29        std::cout << "Enter the input directory: ";
30        std::cin >> inputDirectory;
31
32        std::vector<ClassInfo> classes;
33        findAndProcessJavaFiles(inputDirectory, classes);
34
35        std::sort(classes.begin(), classes.end(), [](const ClassInfo& a,
const ClassInfo& b) {
36            return a.className < b.className;
37        });
38
39        fs::path currentPath = fs::current_path();
40        fs::path outputPath = currentPath / "classes.txt";
41        writeClassesToFile(outputPath.string(), classes);
42        return 0;
43    }
44
45
```