|  |  |
| --- | --- |
| Контролер скобок | |
| Внешняя спецификация | |
| Студент | Бебахани А. |
| Преподаватель | Матюшечкин Д. С.  Сычев О.А. |
| Сдано |  |

1. Назначение

Программа предназначена для проверки корректности расстановки скобок. Она анализирует входной файл с программной на языке C++ и определяет, все ли открывающие скобки имеют соответствующие закрывающие.

1. Функциональные требования
   1. Действия над объектами

Программа должна принимать текстовую строку, содержащую символы скобок, и определять, что каждая открывающая скобка имеет соответствующую закрывающую скобку.

* 1. Ограничения

Обрабатываемая строка должна содержать не более 1024 символов.

Анализ производится только для символов, являющихся скобками: (, ), [, ], {, }. Все остальные символы игнорируются при проверке баланса скобок.

Программа не должна учитывать символы, не влияющие на баланс скобок (например, буквы, цифры, пробелы).

Программа должна игнорировать скобки, упомянутые в комментариях различного вида (однострочные, многострочные) и в строках.

1. Входные и выходные данные

Программа должна получать один параметр командной строки: имя входного файла с записанной программной на языке C++.

Пример команды запуска программы: BracketChecker2.exe test.cpp.

Входные данные представляются в виде одного текстового файла с расширением “.cpp”. Данные представляются как корректно-записанная программа на языке C++.

Выходными данными является файл, содержащие в себе сведения о корректности расставленных скобок.

Примеры файлов с входными и выходными данных представлены в приложении А.

1. Требования к надежности

Программа не должна приводить к аварийному завершению или зависанию при обработке как корректных, так и некорректных входных данных. В процессе работы программы не должно происходить ее аварийного завершения или зависания.

1. Требования программной совместимости

Программа будет разработана на языке С++ с использованием библиотеки STL. Дополнительного программного обеспечения не требуется.

Входные файлы могут быть подготовлены в редакторе Блокнот.

Приложение А

Примеры входных и выходных данных

|  |  |
| --- | --- |
| input.txt | result.txt |
| #include <stdio.h>  int main() {  printf("Hello, world!\n");  return 0;  } | All brackets are correctly closed. |
| // В программе содержаться комментарии со скобками #include <stdio.h>  int main() {  // Этот комментарий содержит { [ ( ) ] }  printf("Hello, world!\n");  return 0;  } | All brackets are correctly closed. |
| // В программе пропущена скобка у функции main  #include <stdio.h>  int main() {  printf("Hello, world!\n");  return 0; | Unmatched brackets found:  Bracket '{' at Line 2, Position 12 is unmatched. |
| // Незакрытая скобка в строке, которая не должна учитываться  #include <stdio.h>  void foo(int a[5]) {  for (int i = 0; i < 5; i++) {  printf("Value: (%d\n", a[i]);  }  } | All brackets are correctly closed. |