Создание класса BadCharTbl:

Класс реализован в заголовочном файле как inline и содержит массив целых чисел размером 256 (по числу символов ASCII).

В методе Init происходит инициализация массива значением длины строки и заполнение таблицы плохих символов.

Реализация функции BMH:

Функция принимает две строки: шаблон t и строку для поиска T.

Используется таблица плохих символов для определения смещений при поиске.

Результатом функции является список позиций вхождения шаблона в строку.

Генерация случайных строк:

Реализована функция generateRandomString для генерации случайных строк из ограниченного алфавита ('a' и 'b').

Тестирование алгоритма:

В функции main генерируются случайные строки и выполняется поиск вхождений шаблона в строку.

Для проверки корректности поиска используется как алгоритм BMH, так и метод find класса string.

Сравнение производительности:

Включён заголовочный файл <chrono> для измерения времени выполнения поиска.

Время измеряется для обоих методов поиска на разных длинах строк и алфавитов.

Результаты исследований:

Корректность алгоритма:

Оба метода поиска (BMH и find) дают одинаковые результаты для всех тестовых строк, что подтверждает корректность реализации.

Производительность:

Измерение времени выполнения показало, что метод BMH может быть более эффективным для длинных строк и коротких шаблонов.

Время выполнения поиска зависит от длины строки и размера алфавита. При увеличении длины строки метод BMH показывает более стабильные результаты по сравнению с методом find.

Оптимизация:

Удалось снизить количество вызовов strlen за счёт сохранения длины строк в переменных.

Программа показала отсутствие утечек памяти при использовании инструментов анализа.

Заключение:

Алгоритм Бойера-Мура-Хорспула успешно реализован и протестирован. В ходе работы удалось убедиться в его эффективности по сравнению с методом find класса string. Оба метода поиска показали корректные результаты, однако метод BMH продемонстрировал лучшую производительность при определённых условиях. Проведённые измерения времени выполнения подтверждают теоретические преимущества алгоритма BMH.