

Отчет о выполнении индивидуального домашнего задания №1

Выполнил: Иванов Георгий Ярославович

Группа: БПИ223

Вариант: 24

Условие: разработать программу, которая ищет в ASCII-строке заданную подстроку и возвращает индекс первого символа первого вхождения подстроки в строке. Подстрока вводится как дополнительный аргумент.

Выполнены критерии на оценку 8:

Структура файлов:

папка grade_four - файлы с программой на оценку 4

папка grade_five - файлы с программой на оценку 5

папка grade_six - файлы с программой на оценку 6

папка grade_seven - файлы с программой на оценку 7

папка grade_eight - файлы с программой на оценку 8

папка grade_nine - файлы с программой на оценку 9

tests\input - входные данные для тестов

tests\output - результаты тестов

Параметры программ

Программы на 4-7 баллов принимают на вход 3 аргумента командной строки: имя входного файла, имя выходного файла, подстроку, которую необходимо найти во входном файле.

Программы на 8-9 баллов:

1 программа на вход принимает 2 аргумента командной строки - имя входного файла и имя выходного файла.

2 программа на вход принимает 1 аргумент командной строки - подстроку, которую необходимо найти во входном файле.

Запуск программ

Программы на 4 - 7 баллов.

В терминале перейдите в каталог с программой и введите сначала `gcc main.c -o main`, затем введите `./main`

Программы на 8-9 баллов.

Чтобы запустить эти программы откройте два окна терминала сразу и введите в одном: `gcc main.c -o main`, а затем `./main`

А в другом окне введите `gcc finder.c -o finder`, а затем `./finder`

Схема выполнения задачи на 4 баллов:

- 1) Первый процесс считывает данные из файла и отправляет данные второму процессу через первый неименованный канал.
- 2) Второй процесс считывает данные от первого процесса через первый неименованный канал, находит первое вхождение подстроки в строку и отправляет индекс первого вхождения, если такого нет отправляет -1 через второй неименованный канал третьему процессу.
- 3) Третий процесс считывает данные из второго неименованного канала и записывает результаты в файл.

Процессы создаются с помощью `fork()`.

Схема выполнения задачи на 5 баллов:

- 1) Первый процесс считывает данные из файла и отправляет данные второму процессу через первый именованный канал, названный `read.fifo`
- 2) Второй процесс считывает данные из канала `read.fifo`, находит первое вхождение подстроки в строку и отправляет индекс первого вхождения, если такого нет отправляет -1 через второй именованный канал `write.fifo` третьему процессу.

3) Третий процесс считывает данные из второго имеенованного канала `write.fifo` и записывает результаты в файл.

Процессы создаются с помощью `fork()`.

Схема выполнения задачи на 6 баллов:

1) Первый процесс считывает данные из файла и отправляет данные второму процессу через первый неименованный канал.

2) Второй процесс считывает данные от первого процесса через первый неименованный канал, находит первое вхождение подстроки в строку и отправляет индекс первого вхождения, если такого нет отправляет -1 через второй неименованный канал первому процессу.

3) Первый процесс считывает данные из второго неименованного канала и записывает результаты в файл.

Процессы создаются с помощью `fork()`.

Схема выполнения задачи на 7 баллов:

1) Первый процесс считывает данные из файла и отправляет данные второму процессу через первый именованный канал, названный `read.fifo`

2) Второй процесс считывает данные из канала `read.fifo`, находит первое вхождение подстроки в строку и отправляет индекс первого вхождения, если такого нет отправляет -1 через второй именованный канал `write.fifo` первому процессу.

3) Первый процесс считывает данные из второго имеенованного канала `write.fifo` и записывает результаты в файл.

Процессы создаются с помощью `fork()`.

Схема выполнения задачи на 8 баллов:

- 1) Первый процесс считывает данные из файла и отправляет данные второму процессу через первый именованный канал, названный `read.fifo`
- 2) Второй процесс считывает данные из канала `read.fifo`, находит первое вхождение подстроки в строку и отправляет индекс первого вхождения, если такого нет отправляет -1 через второй именованный канал `write.fifo` первому процессу.
- 3) Первый процесс считывает данные из второго именованного канала `write.fifo` и записывает результаты в файл.

Схема выполнения задачи на 9 баллов:

- 1) Первый процесс считывает часть данных из файла и отправляет данные второму процессу через первый именованный канал, названный `read.fifo`
- 2) Второй процесс считывает данные из канала `read.fifo`, находит первое вхождение подстроки в строку и отправляет индекс первого вхождения через второй именованный канал `write.fifo`, завершая исполнение при этом.
- 3) Если такого вхождения в части данных нет, то он ждет следующей части данных, через первый именованный канал `read.fifo`, это продолжается пока будет либо не найдена нужная подстрока, либо не закончится файл. Если подстрока так и не будет найдена, во второй канал `write.fifo` будет передано -1.
- 4) Первый процесс считывает данные из второго именованного канала `write.fifo` и записывает результаты в файл.

Тестирование

В каталоге `tests\input` приведены входные данные тестов, сгенерированных случайным образом.

В каталоге `tests\output` приведены выходные данные (индексы найденных подстрок в файлах).

Генерация производилась с помощью программы на Python, на всякий случай приложил её к программам (файл `generating_string.py`). Также в этом файле закомментирован код, помогавший проверять правильность работы программ данной работы.