

Цель

Получение навыков проектирования программ методом «сверху вниз», а также опыта программной обработки строковых данных.

Задание

Исходная задача должна быть разбита на четыре основные подзадачи:

- 1) Ввод данных – `input ()` – предусматривает взаимодействие с пользователем и возвращает строку, содержащую входные данные (путь);
- 2) проверка корректности данных – `check ()` – которая обеспечивает проверку допустимости длины входной строки и используемых в ней символов. Значения, возвращаемые функцией `check()`, должны позволять определить тип ошибки и (если возможно) номер символа, в которой она обнаружена;
- 3) обработка – `process ()` – входных данных согласно заданию;
- 4) вывод данных – `output ()` – на экран, обеспечивает отображение полученных результатов или сообщений об ошибке.

Вариант №2

На вход программы поступает строка, соответствующая абсолютному пути. Для заданной строки выполнить следующие операции:

- а) определить тип ОС, использующей данный путь;
- б) найти количество символов в имени конечного файла (длина файла без учета каталогов) и распечатать его имя;
- в) определить, присутствуют ли во входном пути директории, имена которых состоят только из цифр, если да – заменить имя такого каталога на число, которое на единицу превышает числовое представление имени каталога.

Анализ задачи

1. Способ представления данных, над которыми производятся операции

- Файловые пути Windows:

Полное имя файла в Windows-системах состоит из буквы диска, после которого ставится двоеточие и обратная наклонная черта (обратный слеш), затем через обратные слеша перечисляются подкаталоги, в конце пишется имя файла. Например:

`C:\Windows\System32\calc.exe`

Полное имя файла (включая расширение) в ОС Windows может содержать до 260 символов, данное значение определено константой `MAX_PATH`.

- Файловые пути GNU/Linux:

В UNIX-подобных операционных системах (например, GNU/Linux) полный путь состоит из слеша ("/"), обозначающего корневой каталог, после которого идет перечисление каталогов и подкаталогов, разделенных также символом "/", в конце указывается имя файла. Например:

`/usr/local/bin/gcc`

Пути, начинающиеся не с символа "/", считаются относительными и отсчитываются относительно рабочего каталога. Пример:

Пусть текущий каталог – `/home/alex/video` тогда относительный путь `prog/c_manual.avi` соответствует абсолютному пути `/home/alex/video/prog/c_manual.avi`

Особое значение имеют пути, которые начинаются со знака тильда ("~"). Тильда обозначает домашнюю директорию текущего или указанного пользователя.

Если после знака тильда ("~") следует слэш ("/"), то речь идет о пути относительно домашнего каталога текущего пользователя:

Пусть текущий пользователь – **iv421s01** Домашний каталог пользователя **iv421s01** – `/home/STUDENTS/iv421s01/` тогда путь `~/video/prog/c_manual.avi` соответствует абсолютному пути: `/home/STUDENTS/iv421s01/video/prog/c_manual.avi`