

## Задание 2

### Моделирование работы интерпретатора SHELL

(программа My\_Shell)

**Входной язык:** подмножество командного языка SHELL (определяется вариантом).

**Поток команд:**

1. командный файл, т.е. каждая строка файла - это отдельная команда, которая должна быть выполнена интерпретатором (имя файла - аргумент в командной строке при вызове интерпретатора);
2. стандартный входной поток команд;
3. для исполнения каждой команды требуется запуск программы My\_Shell.

**ВНИМАНИЕ!!!** "побочный" эффект выполнения уже обработанных команд (например, перенаправление ввода-вывода) не должен влиять на выполнение последующих команд.

**Входной язык (варианты):**

*Общая часть (одинаковая для всех вариантов):*

- # конвейер  $pr_1 | pr_2 | \dots | pr_N$  для произвольного  $N \geq 2$ ; считать, что аргументов у  $pr_i$  ( $1 \leq i \leq N$ ) нет (но возможна реализация с произвольным числом аргументов у каждого процесса)
- # перенаправление ввода-вывода  $<$ ,  $>$ ,  $>>$  (в том числе для  $pr_1$  и  $pr_N$  в конвейере)  
Например,  $pr < data > res$   
 $pr_1 | pr_2 > res.txt$
- # запуск в фоновом режиме  $\&$  (в том числе и для конвейеров)  
Например,  $pr arg1 arg2 \&$   
 $pr_1 | pr_2 | pr_3 > res.all \&$

*Вариантная часть:*

В каждый вариант входит (как минимум) один из подпунктов каждого пункта, отмеченного римской цифрой. Звездочкой отмечены более сложные подпункты. Части подпунктов, содержащие слово «возможно», могут быть опущены при выборе варианта; их реализация усложняет вариант. Вариант определяет преподаватель.

I. 1. **mv old\_file new\_file**

2. cp file copy\_file

II. 1. wc filename

результат: filename строк слов символов (возможен список имен файлов; в этом случае подобная информация выдается о каждом файле)

2. **grep substring filename**

результат: строки файла filename, содержащие substring как подстроку (возможен флаг -v; в этом случае результат - это строки, которые не содержат substring как подстроку)

3. cmp filename1 filename2

результат: информация о первом различии в содержимом двух файлов

Например, filename1 differs from filename2: line 5 char 36

#### \*4. sort filename

сортировка строк файла в соответствии с кодировкой ASCII

возможны флаги:

- r обратный порядок
- f не различать большие и малые буквы
- n числовой порядок
- +n начать сортировку с (n+1)-ой строки

#### III. 1. cat filenames

возможен флаг:

- n с нумерацией строк (если файлов несколько, то нумерация сквозная)

#### 2. tail filename

вывод 10 последних строк файла

возможны флаги:

- n n последних строк
- +n с n-ой строки и до конца файла

#### 3. od filename

вывод содержимого файла по 10 символов в строке с указанием номера первого символа в каждой десятке

Например, 000001 a b c d \n e f g h i

000011 j k \t l m n

возможен флаг:

- b с указанием восьмеричных кодов символов

#### IV. 1. pr<sub>1</sub> ; pr<sub>2</sub> ; ... ; pr<sub>N</sub>

последовательное выполнение команд pr<sub>1</sub> - как если бы они были переданы интерпретатору по одной команде в строке

**ВНИМАНИЕ !!!** приоритет операции | выше, чем приоритет операции ; однако, возможно использование скобок: напрмер, ( pr<sub>1</sub>; pr<sub>2</sub> ) | pr<sub>3</sub>, что приведет к конкатенации результатов работы pr<sub>1</sub> и pr<sub>2</sub>, которые будут переданы процессу pr<sub>3</sub> как входные данные.

#### 2. pr<sub>1</sub> && pr<sub>2</sub>

выполнить pr<sub>1</sub>; в случае успеха выполнить pr<sub>2</sub>

#### 3. pr<sub>1</sub> || pr<sub>2</sub>

выполнить pr<sub>1</sub>; в случае неудачи выполнить pr<sub>2</sub>