

ОБОЛОЧКА СИСТЕМЫ (ИНТЕРПРЕТАТОР КОМАНД SHELL)

sh номер_терминала [имя_носителя]
shELL

Утилита -sh- предназначена для ведения диалога с системой. После запуска -sh- на терминале доступны ее команды path, cd, set, us, mem, quit, bye, hi, news (смотрите соответствующие пункты). Утилита как правило запускается в командном файле конфигурации, подготовленном администратором системы.

Для запуска программы необходимо указать полное имя файла, в котором хранится код программы, либо имя файла на директории, на которую установлен путь (см. path), например:

/\$v/моя_директ/мой_файл
- запуск программы, коды которой лежат в файле '\$v/моя_директ/мой_файл'.

mem	- показывает карту оперативной памяти.
quit	- переход в пультовой режим (FNO).
hi	- выдает историю последнего неудавшегося процесса.
bye	- заканчивает работу с утилитой sh.

ПРИМЕРЫ:

ПРИМЕР написания командного файла:

\$Командный файл -doc-.

\$Установить путь на директории -sym-, -bin- и текущую;
path . /\$v/sym /\$v/bin

\$Установить приглашение с именем текущей директории;
set pmt doc_\[/]

\$Перейти на директорию -doc-:
cd /\$v/doc

\$Перейти на директорию, указанную после -doc-:
cd \$1

КОНЕЦ ПРИМЕРА.

Замечание. Чтобы прервать выполнение командного файла, нажмите CTRL U.

ОПИСАНИЕ КОМАНД

ПУТИ ПОИСКА ФАЙЛОВ:

```
path [.] [..] {"/"<имя_директории>"} "
```

Задаёт последовательность полных имен директорий, на которых надо искать коды загружаемой программы, например:

```
path . .. /$v/моя_дир
```

- кодофайл ищется сначала на текущей директории, затем на предыдущей, затем на директории '/\$v/моя_дир'.

```
path ?
```

- выдается ранее установленный путь.

```
path *+ /ii/users/mydir
```

- к ранее установленному пути добавляется путь на директорию /ii/users/mydir.

```
path
```

- путь заменяется на пустой.

СМЕНА ТЕКУЩЕЙ ДИРЕКТОРИИ:

```
cd <имя_директории>  
CHANGE dIRECTORY
```

Осуществляет переход на другую директорию, например:

```
cd ..
```

- переход на предыдущую директорию;

```
cd /
```

- переход на корневую директорию;

```
cd ?
```

- показывает путь к текущей директории от корня файлового дерева.

УСТАНОВКА ПАРАМЕТРОВ:

```
set [<параметры>]
```

Устанавливает параметры -shell-.

Имеется возможность:

- сменить имя стартового носителя системы;
- установить приглашение в начале командной строки;
- установить разделитель в командной строке;

- включить/отключить запрос на выполнение строк командного файла;
- включить/отключить показ строк командного файла;
- установить стек для запуска задачи;

При вызове без параметров выдаются установленные ранее параметры.

СМЕНА НОСИТЕЛЯ:

```
set vol <новое имя>
set volume
```

Везде вместо \$v (см. -up-) будет пониматься новое имя стартового носителя системы.

ПРИГЛАШЕНИЕ:

```
set pmt <приглашение>
set pROmPtER
```

Устанавливает приглашение в начале командной строки.

Приглашение:: { литера | '\' управление }
 Управление:: '/' | '?' | 't' | 'c' | '%' | '\'

Вместо символа управления в приглашение будет вставлено:

```
'/' - имя текущей директории;
'?' - полный путь до текущей директории;
't' - текущее время;
'c' - число исполненных команд;
'%' - пробел
'\ ' - символ '\'
```

Остальные символы будут вставлены в приглашение без изменения.

Пример:

```
set pmt \t \]\%
соответствует приглашение
17:00.00 currentdir]
```

РАЗДЕЛИТЕЛЬ В КОМАНДНОЙ СТРОКЕ:

```
set ics <разделитель>
iNTER cOMMAND sEPARATOR
```

Устанавливает разделитель в командной строке, например:

```
set ics |
- теперь команды в одной строке можно разделять символом
```

'|'. По умолчанию в системе установлен разделитель ";".

ЗАПРОС НА ПОДТВЕРЖДЕНИЕ:

```
set qcf [OFF|ON]
set qQUERY IN cOMMAND fILE
```

Запрашивать подтверждение перед исполнением каждой строки командного файла. Ответы:

```
'Spase', 'n', 'н'      - не исполнять;
'CR', 'LF', 'y', 'д'   - исполнять.
```

ПОКАЗ СТРОК:

```
set scf [OFF|ON]
set sHOW cOMMAND fILE
```

on/off - включить/отключить показ строки командного файла перед исполнением.

РАЗМЕР СТЕКА:

```
set stk <размер стека>
set stEk
```

Устанавливает размер стека для запуска задач (в Кб). Теперь коды каждой запускаемой задачи (кроме уже загруженных в память) и ресурсы для всех задач (и тех, чьи коды загружены) размещаются на этом стеке. Исключение составляют буферы ввода-вывода модуля StdIO и задачи, запущенные навсегда (например, с помощью утилит load, run, ins) - они размещаются вне стека.

Установленный размер стека не гарантирует наличие оперативной памяти такого размера, и -sh- пытается запустить задачу в имеющейся памяти. Если это не удастся, то выдается соответствующее сообщение.

Задача может затребовать память вне стека с помощью процедур модуля Storage.

ЗАПУСК КОМАНДНОГО ФАЙЛА ПОЛЬЗОВАТЕЛЯ:

```
us <имя файла>
usER
```

Запускает командный файл, находящийся на директории /\$v/usr.

НОВОСТИ:

```
news <имя файла>
```

Запускает командный файл `/$v/usr/NEWS`.