

# Цель работы

---

Познакомиться с операционной системой Linux. Получить практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.

## Задание

---

Выполнить упражнения, используя команды vi

## Ход работы

---

### Задание 1. Создание нового файла с использованием vi

1. Создадим каталог с именем `~/work/2020-2021/OperatingSystems/laboratory/lab09` и переходим в него. В этом каталоге вызываем **vi** и создаем файл **hello.sh** (Рис. -@fig:001)

```
[dnbabkov@dnbabkov ~]$ cd work/2020-2021/OperatingSystems/laboratory/lab09
[dnbabkov@dnbabkov lab09]$ vi hello.sh
```

{ #fig:001 }

2. Нажав клавишу **i**, введем текст на рисунке (Рис. -@fig:002). Нажав клавишу **Esc** перейдем в командный режим.

```
#!/bin/hash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

{ #fig:002 }

3. Нажимаем **Shift + ;** для перехода в режим последней строки. Нажимаем клавиши **w** (запись) и **q** (выход), а после - **Enter**. Этим мы сохраним текст и завершим работу. (Рис. -@fig:003)

~

:wq

{ #fig:003 }

4. Делаем файл исполняемым (Рис. -@fig:004)

```
chmod u+x hello.sh
```

{ #fig:004 }

### Задание 2. Редактирование существующего файла

1. Вызываем vi на редактирование файла командой **vi ~/work/2020-2021/OperatingSystems/laboratory/lab09/hello.sh**
2. Устанавливаем курсор после слова **HELL** во второй строке, переходим в режим вставки и заменяем это слово на **HELLO**, после чего возвращаемся в командный режим (Рис. -@fig:005)

```
#!/bin/nasn
HELLO=Hello
{ #fig:005 }
```

3. Ставим курсор на четвертую строку и стираем слово **LOCAL** комбинацией клавиш **d w**.

Переходим в режим вставки и пишем **local** на месте стертого слова. (Рис. -@fig:006)

```
1 hello {
  local HELLO=World
  echo $HELLO
} { #fig:006 }
```

4. Устанавливаем курсор на последней строке файла. После неё вставляем строку, содержащую следующий текст: `echo $HELLO` (Рис. -@fig:007).

```
#!/bin/hash
HELLO=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO
{ #fig:007 }
```

5. Удаляем последнюю строку с помощью клавиш **d d** (Рис. -@fig:008)

```
echo $HELLO
}
echo $HELLO
hello
~
{ #fig:008 }
```

6. Отменяем последнее действие клавишей **u** в командном режиме (Рис. -@fig:009).

```
#!/bin/hash
HELLO=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO
{ #fig:009 }
```

7. Нажимаем **Shift + ;**, и пишем **wq** для выхода с сохранением.

## Контрольные вопросы

---

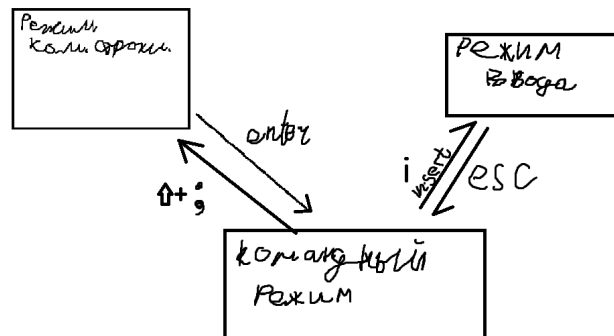
1. Редактор **vi** имеет три режима работы:
  - командный режим – предназначен для ввода команд редактирования и навигации по редактируемому файлу;
  - режим вставки – предназначен для ввода содержания редактируемого файла;
  - режим последней (или командной) строки – используется для записи изменений в файл и выхода из редактора.
2. Чтобы выйти из редактора, не сохраняя произведённые изменения, нужно в режиме командной строки нажать клавиши **: q !**
3. Команды позиционирования:
  - **0** (ноль) – переход в начало строки;
  - **\$** – переход в конец строки;

- **G** – переход в конец файла;
  - **nG** – переход на строку с номером n.
4. При использовании прописных W и B под разделителями понимаются только пробел, табуляция и возврат каретки. При использовании строчных w и b под разделителями понимаются также любые знаки пунктуации.
5. Чтобы из любого места редактируемого файла перейти в начало (конец) файла, нужно в режиме командной строки нажать клавиши **1 G (G)**.
6. Команды редактирования:
- Вставка текста
    - **a** – вставить текст после курсора;
    - **A** – вставить текст в конец строки;
    - **i** – вставить текст перед курсором;
    - **n i** – вставить текст n раз;
    - **I** – вставить текст в начало строки.
  - Вставка строки
    - **o** – вставить строку под курсором;
    - **O** – вставить строку над курсором.
  - Удаление текста
    - **x** – удалить один символ в буфер;
    - **dw** – удалить одно слово в буфер;
    - **d\$** – удалить в буфер текст от курсора до конца строки;
    - **d0** – удалить в буфер текст от начала строки до позиции курсора;
    - **dd** – удалить в буфер одну строку;
    - **n dd** – удалить в буфер n строк.
  - Отмена и повтор произведённых изменений
    - **u** – отменить последнее изменение;
    - **.** – повторить последнее изменение.
  - Копирование текста в буфер
    - **Y** – скопировать строку в буфер;
    - **n Y** – скопировать n строк в буфер;
    - **yw** – скопировать слово в буфер.
  - Вставка текста из буфера
    - **p** – вставить текст из буфера после курсора;
    - **P** – вставить текст из буфера перед курсором.
  - Замена текста
    - **cw** – заменить слово;
    - **n cw** – заменить n слов;

- **c \$** – заменить текст от курсора до конца строки;
  - **r** – заменить слово;
  - **R** – заменить текст.
- Поиск текста
    - **/** текст – произвести поиск вперёд по тексту указанной строки символов текст;
    - **?** текст – произвести поиск назад по тексту указанной строки символов текст.
  - Копирование и перемещение текста
    - **: n,m d** – удалить строки с n по m;
    - **: i,j m k** – переместить строки с i по j, начиная со строки k;
    - **: i,j t k** – копировать строки с i по j в строку k;
    - **: i,j w** имя-файла – записать строки с i по j в файл с именем имя-файла.
7. Чтобы заполнить строку символами \$, необходимо для начала перейти на эту строку, нажав клавиши **n G**, где n – номер строки, далее нажать **O** для перехода в начало строки. Теперь необходимо нажать **c \$**, чтобы заменить текст от курсора до конца строки, и ввести символы \$.
8. Чтобы отменить по одному предыдущему действию последовательно, необходимо нажать **u**. Чтобы отменить все изменения, произведённые со времени последней записи, нужно нажать **: e !**.
9. Команды редактирования в режиме командной строки
- Копирование и перемещение текста
    - **: n,m d** – удалить строки с n по m;
    - **: i,j m k** – переместить строки с i по j, начиная со строки k;
    - **: i,j t k** – копировать строки с i по j в строку k;
    - **: i,j w** имя-файла – записать строки с i по j в файл с именем имя-файла.
  - Запись в файл и выход из редактора
    - **: w** – записать изменённый текст в файл, не выходя из vi;
    - **: w** имя-файла – записать изменённый текст в новый файл с именем имя-файла;
    - **: w !** имя-файла – записать изменённый текст в файл с именем имя-файла;
    - **: w q** – записать изменения в файл и выйти из vi;
    - **: q** – выйти из редактора vi;
    - **: q !** – выйти из редактора без записи;
    - **: e !** – вернуться в командный режим, отменив все изменения, произведённые со времени последней записи.
  - Опции Опции редактора vi позволяют настроить рабочую среду. Для задания опций используется команда **set** (в режиме последней строки):
    - **: set all** – вывести полный список опций;
    - **: set nu** – вывести номера строк;
    - **: set list** – вывести невидимые символы;
    - **: set ic** – не учитывать при поиске, является ли символ прописным или строчным. Если вы хотите отказаться от использования опции, то в команде **set** перед именем опции надо поставить **no**.
10. Чтобы определить, не перемещая курсора, позицию, в которой заканчивается строка, нужно в командном режиме находясь на нужной строке нажать **\$** и посмотреть на число после запятой в

правом нижнем углу экрана.

11. Опции редактора vi позволяют настроить рабочую среду. Для задания опций используется команда set (в режиме командной строки). Если вы хотите отказаться от использования опции, то в команде set перед именем опции надо поставить no.  
Чтобы просмотреть опции редактора vi, необходимо нажать : set all. Нажав : help "название\_опции", можно узнать назначение конкретной опции.
12. В режиме командной строки внизу редактора присутствует :, в режиме ввода -- **ВСТАВКА** --, в командном режиме внизу ничего нет.
13. Граф взаимосвязи режимов работы редактора vi (Рис. -@fig:010)



{ #fig:010 }

## Вывод

Я ознакомился с операционной системой Linux и Получил практические навыки работы с редактором vi.