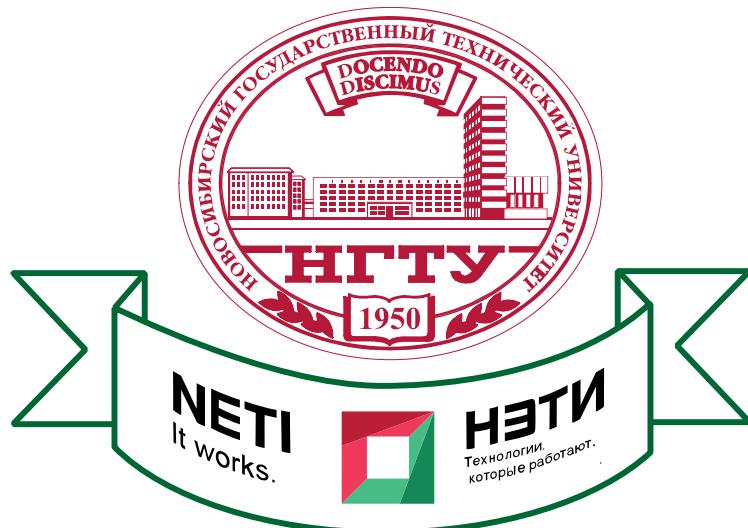


Министерство науки и высшего образования
Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

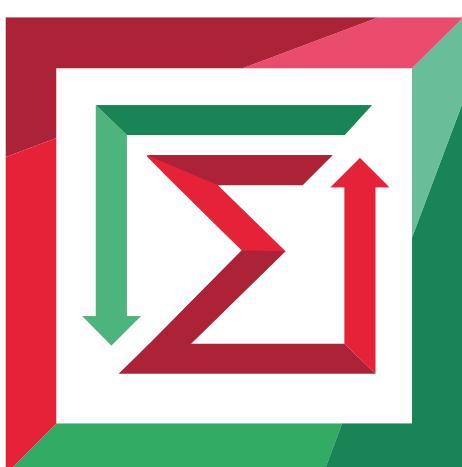


Теоретической и прикладной математики

Лабораторная работа № 2

по дисциплине «Операционные системы, среды и оболочки»

Текстовый редактор Vim



Факультет: ПМИ
Группа: ПМИ-02
Вариант: 2
Студент: Сидоров Даниил Игоревич
Преподаватель: Кобылянский Валерий Григорьевич,
Филиппова Елена Владимировна
Место для ввода текста.

Новосибирск

2026

1. Цель работы.

Освоение основных возможностей экранного редактора vim, который является расширением стандартного UNIX-редактора vi.

2. Ход работы

1.1. Создали каталог с именем practice.

```
[pmi-b0702@students ~]$ mkdir practice  
[pmi-b0702@students ~]$ █
```

1.2. Перешли в только что созданный каталог с именем practice.

```
[pmi-b0702@students ~]$ cd practice  
[pmi-b0702@students practice]$ pwd  
/home/NSTU/pmi-b0702/practice  
[pmi-b0702@students practice]$ █
```

1.3. Вызвали vi и создали файл с именем “memo”.

```
[pmi-b0702@students practice]$ vi memo  
█  
~  
~  
~  
~  
~  
~
```

1.4. Нажали клавишу i и ввели следующий текст:

```
@REM AUTOEXEC.BAT DTK 386/40  
ECHO OFF  
Path c:\dos;c:\stacker;c:\Util;c:\NC;c:\MOUSE  
SET PROMPT=$P$G  
SET TMP=C:\TEMP  
LH C:\UTIL\RKEGA  
:goto %config%  
:student1  
C:\DOS\SMARTDRV.EXE C+ 2048 1024  
goto nc  
:student2  
APPEND E:\tc\bgi  
:teacher  
PATH %path%;E:\windows;e:\tc;e:\tc\bin;e:\foxpro;  
goto win  
:onc  
PATH %path%;G:\pctcp;  
SET TZ=GMT  
goto nc  
:nc  
nc.exe  
goto end  
win.com  
:end
```

1.5. Нажали ":" для перехода в режим последней строки.

```
~  
~  
:
```

- 1.6. Нажали wq, а затем нажали клавишу ENTER для сохранения введённого текста и завершения работы.

```
~  
:wq  
:end  
~  
"memo" 24L, 370C
```

- 2.1. Вызвали vi для редактирования файла memo.

```
@REM AUTOEXEC.BAT DTK 386/40  
ECHO OFF  
Path c:\dos;c:\stacker;c:\Util;c:\NC;c:\MOUSE  
SET PROMPT=$P$G  
SET TMP=C:\TEMP  
LH C:\UTIL\RKEGA  
goto %config%  
:student1  
C:\DOS\SMARTDRV.EXE C+ 2048 1024  
goto nc  
:student2  
APPEND E:\tc\bgi  
:teacher  
PATH %path%E:\windows;e:\tc;e:\tc\bin;e:\foxpro;  
goto win  
:onc  
PATH %path%G:\pctcp;  
SET TZ=GMT  
goto nc  
:nc  
nc.exe  
goto end  
win.com  
:end  
~
```

- 2.3. Перешли в режим вставки и набрали “1-203”.

```
@REM AUTOEXEC.BAT 1-203 DTK 386/40  
ECHO OFF  
Path c:\dos;c:\stacker;c:\Util;c:\NC;c:\MOUSE  
SET PROMPT=$P$G
```

- 2.4. Установили курсор на третью строку и стёрли слово C:\MOUSE с помощью “d\$” (удаление от курсора до конца строки).

```
ECHO OFF  
Path c:\dos;c:\stacker;c:\Util;c:\NC  
SET PROMPT=$P$G
```

- 2.5. Перешли в режим вставки и набрали следующий текст: C:\GMOUSE

```
Path c:\dos;c:\stacker;c:\Util;c:\NC;c:\GMOUSE  
SET PROMPT=$P$G  
SET TMP=C:\TEMP
```

- 2.6. Вставили строку, содержащую следующий текст: extention 287.

```
WIN.COM  
:end  
extention 287  
-- INSERT --
```

2.7. Заменили extention на x при помощи “:s/extension/x”.

```
:end  
x 287  
:s/extension/x
```

2.8. Удалили последнюю строку с помощью “dd”.

```
:end  
~
```

2.9. Ввели команду отмены изменений “u” и отменили удаление последней строки.

```
:end  
x 287  
1 more line; before #10 36 seconds ago
```

2.10. Установили курсор на 5 строку, вставили пустую строку при помощи “o” и ввели текст, а также оставили пустые строки между новым параграфом и следующим за ним.

```
SET PROMPT=$P$G  
  
REM this is a comment  
  
SET TMP=C:\TEMP
```

2.11. Перешли в режим последней строки, сохранили изменения и вышли из редактора.

```
[pmi-b0702@students practice]$ vi memo  
[pmi-b0702@students practice]$
```

3.1. Вернулись в домашний каталог.

```
[pmi-b0702@students practice]$ cd  
[pmi-b0702@students ~]$
```

3.2. Скопировали в каталог ~/practice файл testcase.c, предварительно осуществив его поиск во внешней памяти сервера.

```
[pmi-b0702@students ~]$ find / -name "testcase.c" -exec cp {} ~/practice \; 2>/dev/null  
[pmi-b0702@students ~]$ cd ~/practice  
[pmi-b0702@students practice]$ ls  
memo testcase.c  
[pmi-b0702@students practice]$
```

3.3. Загрузили vi для редактирования файла testcase.c.

```
#include <stdio.h>  
#include <ctype.h>  
#include <string.h>  
/* Manifests for state machine to parse input line. */  
#define WORD 0  
#define IGNORE 1  
/* Globals, used by both subroutines. */  
char *Words[BUFSIZ/2]; /* Worst case, single letters. */  
int WordCount;  
/* Walk through the array of words, find those with the  
* matching karakter, printing them on stdout. Note that  
* the NULL karakter will match all words. */  
  
void PrintWords(wc, match)  
  
    int wc; /* Number of words in Words[] */  
    char match; /* Attempt to match this karakter. */  
    { register int ix; /* Index in Words[]. */  
        register char *cp; /* Pointer for searching. */  
        for (ix=0; ix < wc; ix++) {  
            cp = Words[ix];  
            /* Try to match the given character.  
             * Scan the word, attempting to match,  
             * or until the end of the word is found. */  
            while ((*cp) && (*cp++ != match));
```

“testcase.c” [dos] 63L, 3243C

1,5

Top

3.4. Включили отображение номеров строк при помощи “:set nu”. В файле 63 строки.

```
59             while (gets (buf) != NULL)
60                 WordCount = GetWordCount (buf);
61                 PrintWords (WordCount);
62             return (0); /* Return success */
63 }
```

3.5. Вернулись в начало файла.

```
1 #include <stdio.h>
2 #include <ctype.h>
3 #include <string.h>
4 /* Manifests for state machine to
5 #define WORD      0
6 #define IGNORE    1
7 /* Globals, used by both subroutines */
```

3.6. С помощью “s/WORD/IGNORE”, из начала текста переместились в строку, где впервые встречается слово WORD и заменили его на IGNORE.

```
4 /* Manifests for state machine to
5 #define IGNORE      0
6 #define WORD        1
7 /* Globals, used by both subroutines */
8 char      *Words[BUFSIZ/2];
9 int       WordCount;
10 /* Walk through the array
11   * matching charakter, print
12   * the NULL charakter will
13
14 void PrintWords (wc, match)
15
16 int wc;
17 char match;
18 { register int ix;
19   register char *cp;
20   for (ix=0; ix < wc; ix++)
21     cp = Words[ix];
22     /* Try to match
23     * Scan the word
24     * or until
25     while ((*cp)
26           if (*cp == r
```

s/WORD/IGNORE

3.7. Нашли слово Reset и заменили его на set при помощи “:%s/Reset/set/g”.

```

        case IGNORE:
            if (!isspace(*cp)) {
                Words[wc++] = cp; /* Just start
                                     state = WORD;      /* set the state */
            }
        case WORD:
            if (isspace(*cp)) {
                *cp = '\0';           /* Just complete
                                         state = IGNORE;    /* set the state */
            }
            return wc; /* Return the word count. */
        }

int main(argc, argv) int argc; char *argv[]; { char
/* Check command line arguments. */
if (argc < 2) match = ' ';
/* No command line argument, match all words. */
else match = *++argv[1]; /* match the char after the argument */
/* Until no more input on stdin. */
while(gets(buf) != (char *)NULL) {
    WordCount = GetWords(buf); /* Paste the output buffer. */
    PrintWords(WordCount, match); /* Print the matching words. */
return(0); /* Return success to the shell.

```

:%s/Reset/set/g

3.8. Нашли слово input и заменили его на output при помощи “:%s/Reset/set/g”.

```

/* Until no more output on stdin. */
while(gets(buf) != (char *)NULL) {
    WordCount = GetWords(buf); /* Paste the output buffer. */
    PrintWords(WordCount, match); /* Print the matching words. */
return(0); /* Return success to the shell. */

```

4 substitutions on 4 lines

60,19

3.9. Вставили строку, заполненную вопросительными знаками “?” под строкой: state = WORD.

Нашли фразу при помощи “/state = WORD”

```

        Words[wc++] = cp;
        state = WORD;
    case WORD:
        if (isspace(*cp)) {
            *cp = '\0';
            state = IGNORE;
        }
        return wc; /* Return the word count. */

int main(argc, argv) int argc; char
/* Check command line arguments. */
if (argc < 2) match = ' ';
/* No command line argument, match all words. */
else match = *++argv[1]; /* match the character after the argument */
/* Until no more output on stdin. */
while(gets(buf) != (char *)NULL) {
    WordCount = GetWords(buf);
    PrintWords(WordCount, match);
return(0); /* Return success to the shell.

```

:/state = WORD

Вставили пустую строку при помощи “o”:

```

        state = WORD; /* set the state. */ } break;

```

```

case WORD:

```

Вставили строку вопросительных знаков при помощи “60i ?”

```
state = WORD;      /* set the state. */ ]  
?????????????????????????????????????????????????????????????????????  
case WORD:
```

3.10. Скопировали строки с 16 по 29 в файл printwords.c при помощи “:16,29w printwords.c”.

```
:readin(0);      /* Readin success to  
"printwords.c" [New] [dos] 14L, 800C written
```

3.11. Перешли в конец файла и удалили две последние строки при помощи “2dd”.

```
61  
62          PrintWords(WordCount, match); /* Print the matching words. */ }  
63          return(0); /* Return success to the shell. *  
64  
62          PrintWords(WordCount, match); /* Print the matching words. */ }  
~  
~
```

3.12. Перенесли фрагмент текста в конец файла.

Нашли строку при помощи “/* Manifests”

```
3      #include <string.h>  
4      /* Manifests for state machine to parse output line. */  
5      #define IGNORE
```

Переместили строку в конец файла при помощи “:4,4m 62”

```
61          PrintWords(WordCount, match); /* Print the matching words. */ }  
62          /* Manifests for state machine to parse output line. */  
:4,4m 62
```

3.13. Записали произведенные изменения на диск в файл testvi.c при помощи “testvi.c” и вышли из редактора.

```
62          /* Manifests for state machine to parse output line. */  
:w! testvi.c  
"testvi.c" [New] [dos] 62L, 3242C written
```

3. Вывод:

Контрольные вопросы проработаны.