

# **Лабораторная работа №5**

**Архитектура вычислительных систем**

Дадилов Руслан

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>5</b>	<b>Самостоятельная работа</b>	<b>12</b>
<b>6</b>	<b>Выводы</b>	<b>14</b>
	<b>Список литературы</b>	<b>15</b>

# Список иллюстраций

4.1	мс . . . . .	7
4.2	мс . . . . .	8
4.3	мс . . . . .	9
4.4	мс . . . . .	9
4.5	мс . . . . .	10
4.6	мс . . . . .	10
4.7	мс . . . . .	11
4.8	мс . . . . .	11
5.1	мс . . . . .	12
5.2	мс . . . . .	13
5.3	мс . . . . .	13

# 1 Цель работы

Освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

## 2 Задание

1. Создайте копию файла lab6-1.asm. Внесите изменения в программу (без использования внешнего файла in\_out.asm), так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введённую строку на экран.
2. Получите исполняемый файл и проверьте его работу. На приглашение ввести строку введите свою фамилию.
3. Создайте копию файла lab6-2.asm. Исправьте текст программы с использование подпрограмм из внешнего файла in\_out.asm, так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введённую строку на экран.
4. Создайте исполняемый файл и проверьте его работу.

### 3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы. Например, в табл. 3.1 приведено краткое описание стандартных каталогов Unix.

Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Более подробно об Unix см. в [1–6].

## 4 Выполнение лабораторной работы

1. Открываем mc.

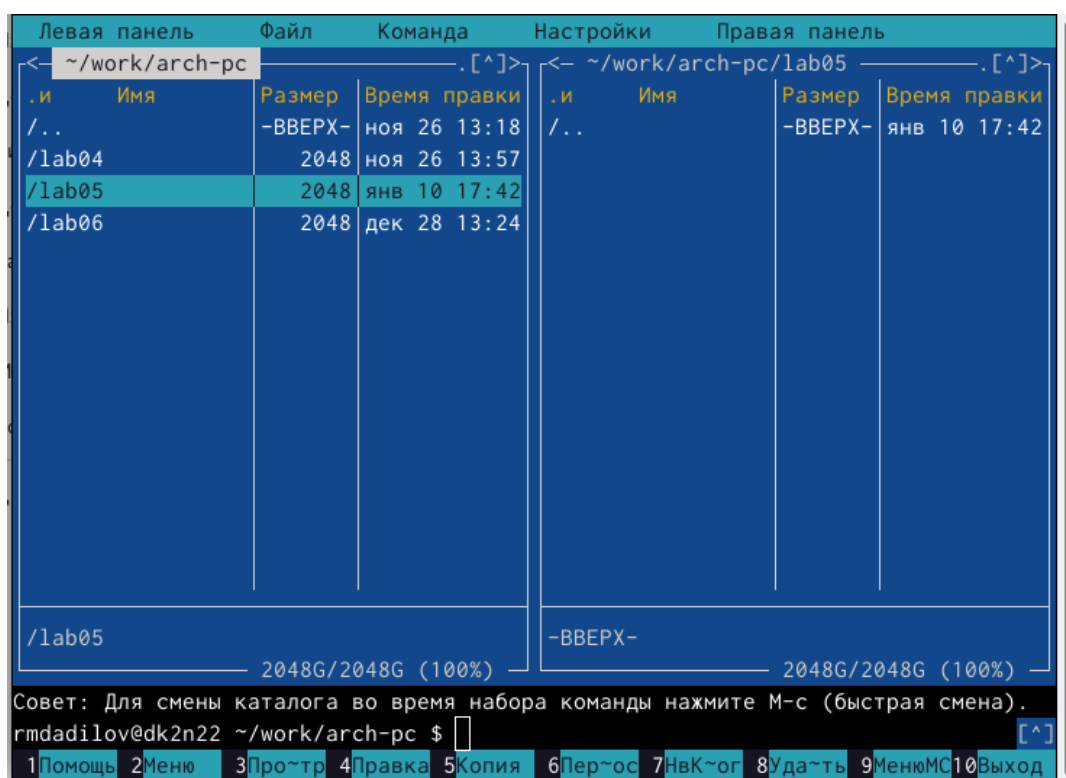


Рис. 4.1: mc

2. Создаем директорию по условию задания . Создаем файл lab5-1 с помощью F7.

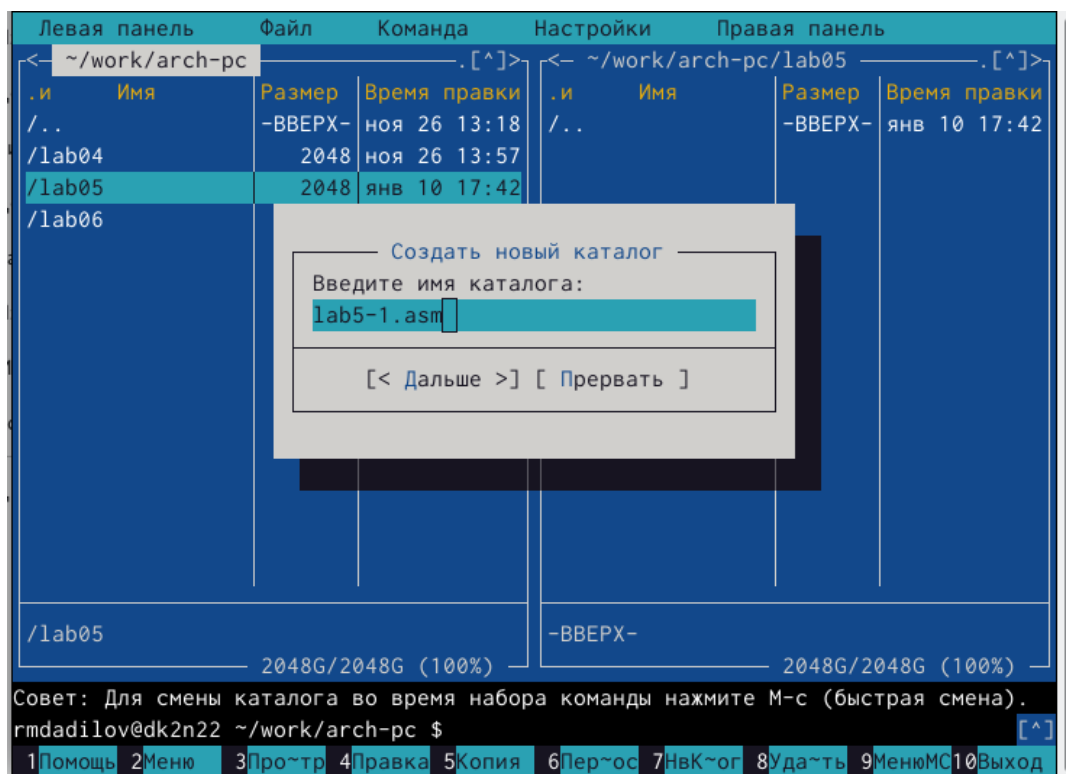


Рис. 4.2: mc

3. Вводим текст программы из задания в туис.



```

SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
;----- Системный вызов 'write'
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
mov eax, 3 ; Системный вызов для чтения (sys_read)
mov ebx, 0 ; Дескриптор файла 0 - стандартный ввод

```

^G Справка    ^O Записать    ^W Поиск    ^K Вырезать    ^T Выполнить    M-U Отмена  
 ^Y Выход    ^P Подсказка    ^Z Занести    ^H Вставить    ^C Прервать    M-F Повтор

Рис. 4.3: mc

#### 4. Вводим данные ФИ

```

rmdadilov@dk2n22 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1.asm
rmdadilov@dk2n22 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
rmdadilov@dk2n22 ~/work/arch-pc/lab05 $ ./lab5-1
Введите строку:
Дадилов Руслан

```

Рис. 4.4: mc

#### 5. Создаем файл lab5-2.

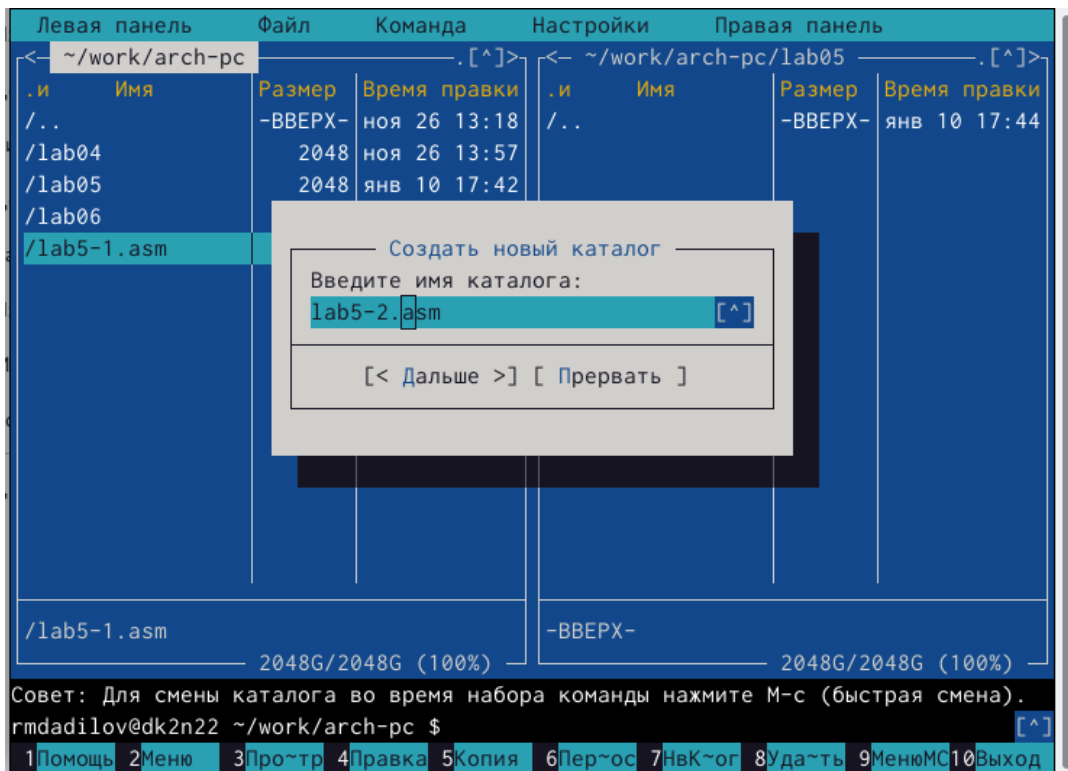


Рис. 4.5: mc

## 6. Изменяем код в lab5-2.

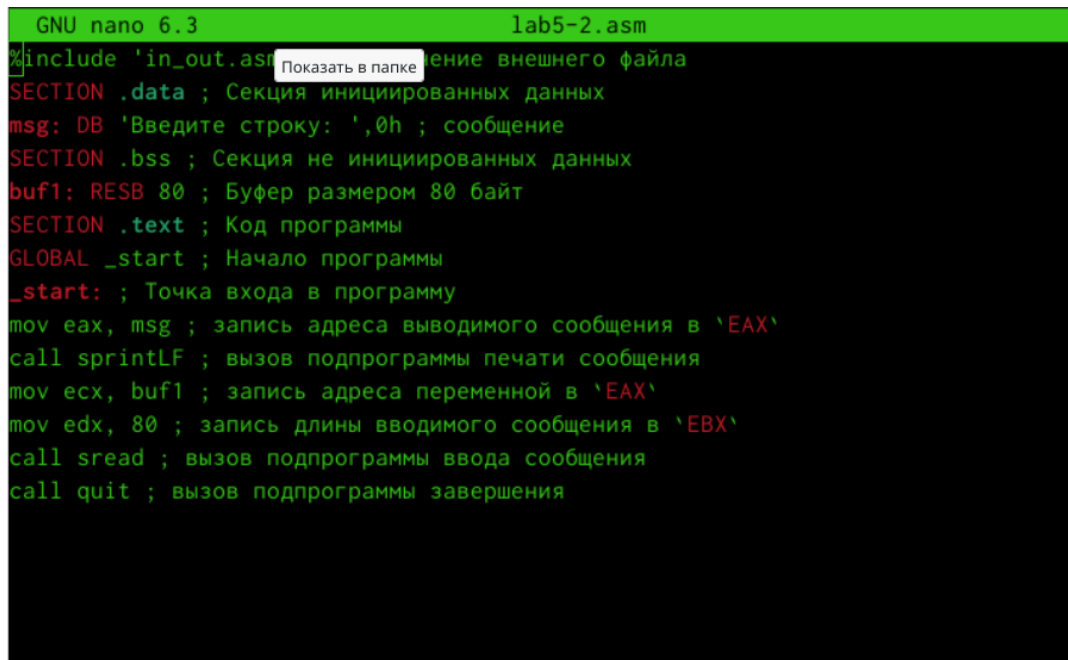


Рис. 4.6: mc

7. Похожие действия выполняем с lab5-3.

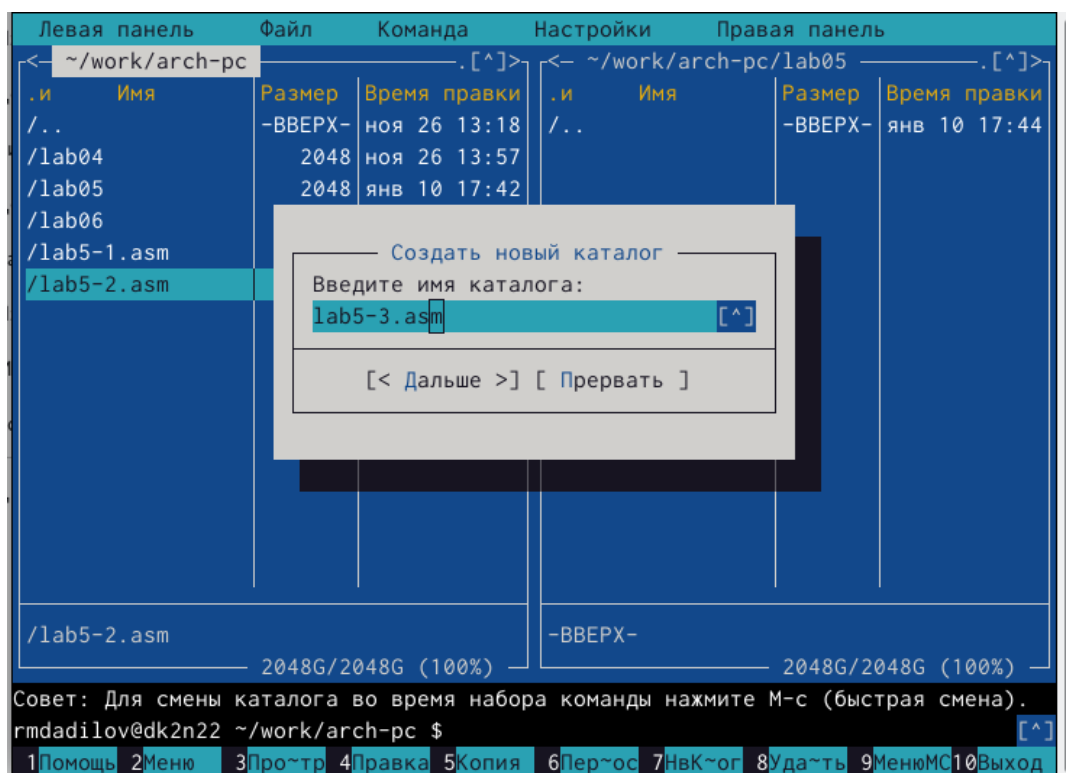


Рис. 4.7: mc

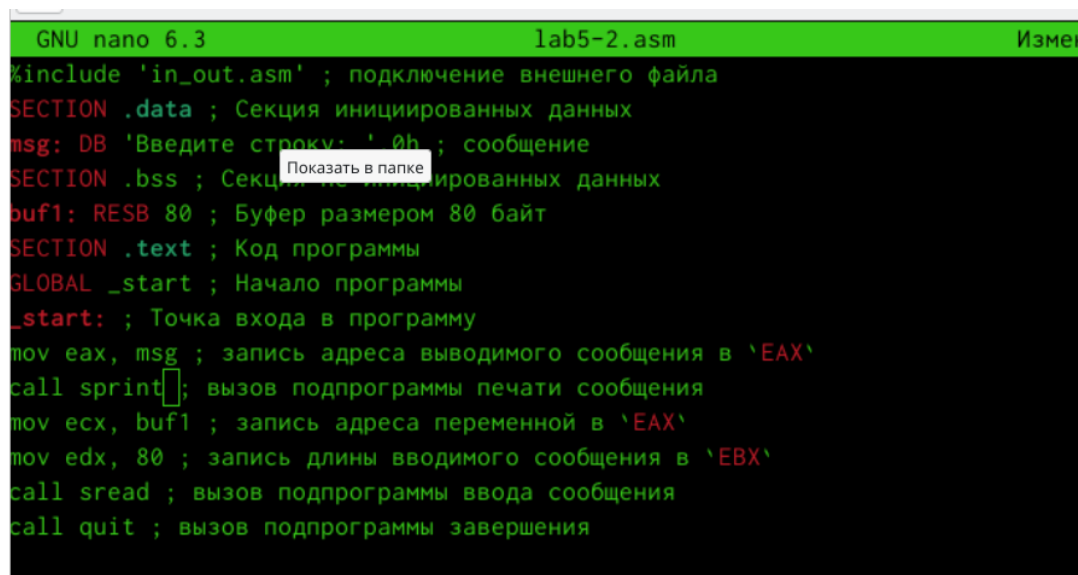


Рис. 4.8: mc

## 5 Самостоятельная работа

1. Создаем копию файла lab5-1.asm. Внесем изменения в программу), так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран.

Копируем файл.

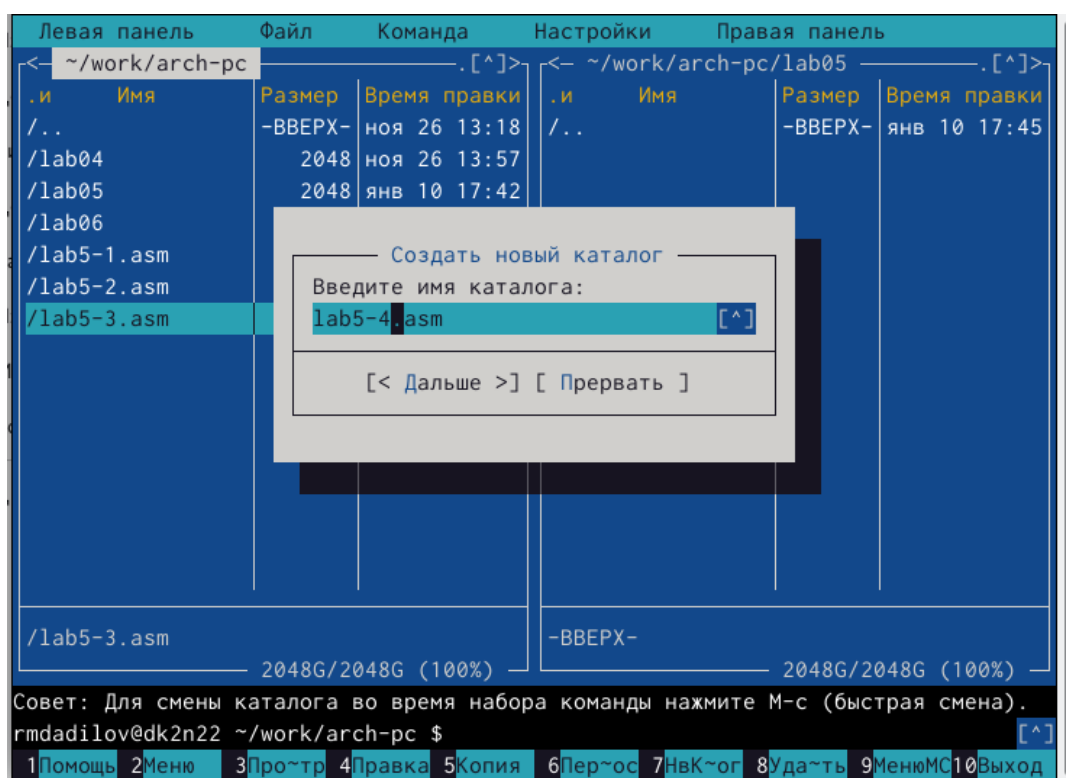


Рис. 5.1: mc

```

%include 'in_out.asm' ; Показывать в папке ; чтение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в 'EAX'
call sprint ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в 'EAX'
mov edx, 80 ; запись длины вводимого сообщения в 'EBX'
call sread ; вызов подпрограммы ввода сообщения

mov eax, buf1
call sprintLF ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в 'EAX'
mov edx, 80 ; запись длины вводимого сообщения в 'EBX'

call quit ; вызов подпрограммы завершения

```

Рис. 5.2: mc

2. Дописываем 4 строки после call sread вызывающие функцию sprintLF. Создаем исполняемый файл и проверяем его работу.

```

rmdadilov@dk2n22 ~/work/arch-pc/lab05 $ nasm -f elf lab5-4.asm
rmdadilov@dk2n22 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-4 lab5-4.o
rmdadilov@dk2n22 ~/work/arch-pc/lab05 $ ./lab5-4
Введите строку: Дадилов Руслан
Дадилов Руслан

```

Рис. 5.3: mc

## 6 Выводы

В ходе выполнения данной лабораторной работы были приобретены практические навыки работы в Midnight Commander. Были освоены инструкции языка ассемблера `mov` и `int`.

# Список литературы

::: {#refs}:

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016.  
URL: <https://www.gnu.org/software/bash/manual/>.
2. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
5. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.
6. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.