

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4

дисциплина: Архитектура компьютера

Студент: Маслова Анастасия Сергеевна

Группа: НКНбд-01-21

МОСКВА

2021 г.

Цель работы: получение навыков программирования небольших командных файлов, арифметических инструкций языка ассемблера NASM на практике.

Ход работы:

1. Создайте в каталоге *Architecture_PC* (созданном при выполнении Лабораторной работы №1) новый подкаталог с именем *lab04* и в нем файл *lab4-1.asm* (для задания №2) и *lab4-2.asm* (для задания №3).

В каталоге *Architecture_PC* я создала подкаталог *lab04* (с помощью команды *mkdir*) и файлы *lab4-1.asm* и *lab4-2.asm* (с помощью команды *touch*) (рис.1).

```
asmaslova@ubuntu2104:~$ cd Architecture_PC/
asmaslova@ubuntu2104:~/Architecture_PC$ mkdir lab04
asmaslova@ubuntu2104:~/Architecture_PC$ ls
lab01 lab02 lab03 lab04
asmaslova@ubuntu2104:~/Architecture_PC$ cd lab04
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ touch lab4-1.asm lab4-2.asm
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ls
lab4-1.asm lab4-2.asm
asmaslova@ubuntu2104:~/Architecture_PC/lab04$
```

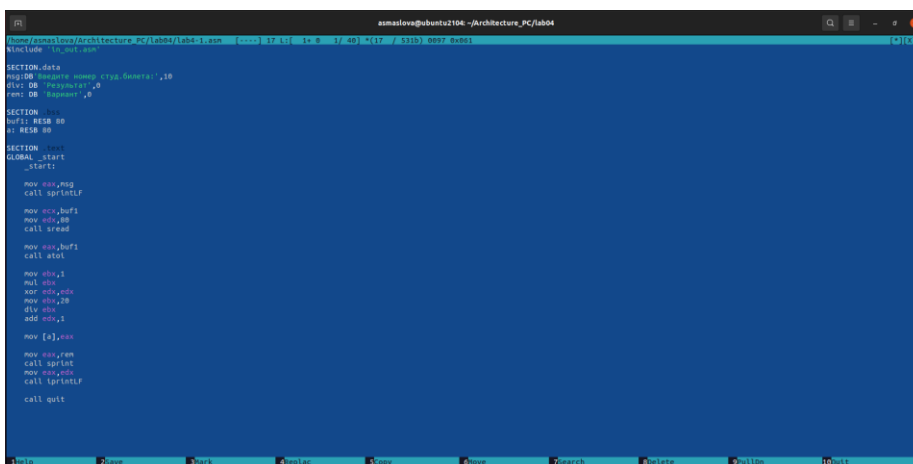
рис. 1. Создание подкаталога и файлов в нем

2. Напишите программу вычисления варианта задания, работающую по следующему алгоритму:

- вывести запрос на введение № студенческого билета
- вычислить номер варианта по формуле: $(Sn \bmod 20) + 1$, где Sn – номер студенческого билета (В данном случае $a \bmod b$ – это остаток от деления a на b).
- вывести на экран номер варианта.

Создайте исполняемый файл и проверьте его работу.

С помощью *msedit* я написала программу для вычисления номера варианта по установленной формуле.



```
asmaslova@ubuntu2104 -j Architecture_PC/lab04
/home/asmaslova/Architecture_PC/lab04/lab4-1.asm [....] 37 3: [ 1x 0 37 40] * (37 / 5105) 0007 0x001
WinCode 00_007.asm

SECTION .data
msg08 equiprntf ctp3.Su0era: 10
str: DB "Programat",0
str: DB "Su0era",0

SECTION .bss
buf1: RESB 80
str: RESB 80

SECTION .code
GLOBAL _start
_start:

mov eax,msg
call sprintf
mov ecx,buf1
mov edx,80
call read
mov ecx,buf1
call atoi
mov ebx,1
mul ebx
xor ecx,ecx
mov ebx,20
div ebx
add ebx,1
mov [a],eax
mov ecx,em
call sprintf
mov ecx,em
call sprintf
call quit
```

рис. 2. Написание программы в mscedit

После этого я проверила создала исполняемый файл и проверила его работу (рис.3). В результате вышло, что мой вариант – 16.

```
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ mcedit lab4-1.asm
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ nasm -f elf lab4-1.asm
lab4-1.asm:3: warning: label alone on a line without a colon might be in error [-w+label-orphan]
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ld -m elf_i386 -o lab4-1 lab4-1.o
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ls
in_out.asm lab4-1 lab4-1.asm lab4-1.o lab4-2 lab4-2.asm lab4-2.lst lab4-2.o makefile
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ./lab4-1
Введите номер студ.билета:
Вариант
1032216455
Вариант16
```

рис. 3. Создание исполняемого файла и проверка его работы

3. Написать программу вычисления выражения $y = f(x)$. Программа должна выводить выражение для вычисления, выводить запрос на ввод значения x , вычислять заданное выражение в зависимости от введенного x , выводить результат вычислений. Вид функции $f(x)$ выбрать из таблицы 4.3 вариантов заданий в соответствии с номером полученным в задании №2. Создайте исполняемый файл и проверьте его работу для значений x_1 и x_2 из 4.3.

Так как у меня 16 вариант, мне было нужно программу для вычисления выражения $y = (10x-5)^2$. Написав программу в mcedit (рис. 4), я создала исполняемый файл и проверила его на $x=3$ и $x=1$ (рис.5).

```
/home/asmaslova/Architecture_PC/lab04/lab4-2.asm [----] 13 L: [ 1+39 40/ 42] *(474 / 476b) 0010 0x00A
#include----->'in_out.asm'

SECTION .data
msg: DB 'Введите x:',0
div: DB 'Результат:',0

SECTION .bss
x: RESB 80
rez: RESB 80

SECTION .text
GLOBAL _start
_start:

    mov eax,msg
    call sprintf

    mov ecx,x
    mov edx,80
    call sread

    mov eax,x
    call atoi

    mov ebx,10
    mul ebx ;eax=10*eax

    sub eax,5

    mov eax,eax
    mul eax

    mov [rez],eax

    mov eax,div
    call sprintf
    mov eax,[rez]
    call iprintf

    call quit
```

рис. 4. Создание программы в mcedit для вычисления функции

```

asmaslova@ubuntu2104:~/Architecture_PC/lab04$ mcedit lab4-2.asm

asmaslova@ubuntu2104:~/Architecture_PC/lab04$ nasm -f elf lab4-2.asm
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ld -m elf_i386 -o lab4-2 lab4-2.o
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ./lab4-2
Введите x:
3
Результат:625
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ./lab4-2
Введите x:
1
Результат:25

```

рис. 5. Создание исполняемого файла вычисления функции и проверка его работы

4. Создайте *make* файл с явными правилами получения исполняемого файла для написанной программы вычисления выражения. Проверьте работу *make*.

Вновь используя команду *touch* и *mcedit* (рис. 6), я создала файл *makefile* для получения исполняемого файла *lab4-2*, после чего проверила его работу (рис.7).

```

/home/asmaslova/Architecture_PC/lab04/makefile [----] 13 L: [ 1+ 0 1/ 6] *(13 / 133b) 0046 0x02E
build: lab4-2.asm
<----->nasm -f elf -g -l lab4-2.lst lab4-2.asm
<----->ld -m elf_i386 lab4-2.o -o lab4-2

clean:
<----->rm -f main lab4-2.o lab4-2.lst

```

рис. 6. Создание *makefile* в *mcedit*

```

asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ls
in_out.asm lab4-1 lab4-1.asm lab4-1.o lab4-2 lab4-2.asm lab4-2.lst lab4-2.o makefile
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ rm lab4-2
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ls
in_out.asm lab4-1 lab4-1.asm lab4-1.o lab4-2.asm lab4-2.lst lab4-2.o makefile
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ make
nasm -f elf -g -l lab4-2.lst lab4-2.asm
ld -m elf_i386 lab4-2.o -o lab4-2
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ls
in_out.asm lab4-1 lab4-1.asm lab4-1.o lab4-2 lab4-2.asm lab4-2.lst lab4-2.o makefile
asmaslova@ubuntu2104:~/Architecture_PC/lab04$ ./lab4-2
Введите x:
1
Результат:25

```

Вывод: в ходе работы я приобрела навыки написания простейших программ для вычисления на ассемблере NASM, а также научилась создавать файлы типа *makefile* для создания исполняемых файлов.