

Отчет по выполнению лабораторной работы №8

Дисциплина: архитектура компьютеров

Новиков Никита Владимирович

Содержание

1 Цель работы

Приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки.

2 Задание

1. Реализация циклов в NASM
2. Обработка аргументов командной строки

3 Выполнение лабораторной работы

3.1 Реализация циклов в NASM

Создаю каталог lab08, перехожу в него и создаю файл lab8-1.asm (рис. [??]).

A screenshot of a terminal window with a dark background. The prompt is (nvnovikov1@nvnovikov1)~[~/.../arch-pc/labs/lab08/report]. The first command is \$ touch lab8-1.asm. The second command is \$ ls, followed by the output: Makefile bib image lab8-1.asm pandoc report.md.

```
(nvnovikov1@nvnovikov1)~[~/.../arch-pc/labs/lab08/report]  
$ touch lab8-1.asm  
  
(nvnovikov1@nvnovikov1)~[~/.../arch-pc/labs/lab08/report]  
$ ls  
Makefile  bib  image  lab8-1.asm  pandoc  report.md
```

Создание каталога и файла

Ввожу в файл текст программы из листинга 8.1(рис. [??]).

```

1 %include 'in_out.asm'
2 SECTION .data
3 msg1 db 'Введите N: ',0h
4 SECTION .bss
5 N: resb 10
6 SECTION .text
7 global _start
8 _start:
9 ; ---- Вывод сообщения 'Введите N: '
10 mov eax,msg1
11 call sprintf
12 ; ---- Ввод 'N'
13 mov ecx, N
14 mov edx, 10
15 call sread
16 ; ---- Преобразование 'N' из символа в число
17 mov eax,N
18 call atoi
19 mov [N],eax
20 ; ---- Организация цикла
21 mov ecx,[N] ; Счетчик цикла, 'ecx=N'
22 label:
23 mov [N],ecx
24 mov eax,[N]
25 call iprintf ; Вывод значения 'N'
26 loop label ; 'ecx-ecx-1' и если 'ecx' не '0'
27 ; переход на 'label'
28 call quit

```

Ввод программы из листинга 8.1

Создаю исполняемый файл и проверяю его работу.(рис. [??]).

```

(nvnovikov1@nvnovikov1)~[~/../arch-pc/labs/lab08/report ]
$ nasm -f elf lab8-1.asm

(nvnovikov1@nvnovikov1)~[~/../arch-pc/labs/lab08/report ]
$ ld -m elf_i386 -o lab8-1 lab8-2.o
ld: cannot find lab8-2.o: No such file or directory

(nvnovikov1@nvnovikov1)~[~/../arch-pc/labs/lab08/report ]
$ ld -m elf_i386 -o lab8-1 lab8-1.o

(nvnovikov1@nvnovikov1)~[~/../arch-pc/labs/lab08/report ]
$ ./lab8-1
Введите N: 8
8
7
6
5
5
4
3
2
1

```

Проверка работы файла

Далее изменяю текст программы, изменив значение регистра ecx в цикле.
(рис. [??]).

```
~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab08/report/lab8-1.asm - Mousepad
File Edit Search View Document Help

1 %include "in_out.asm"
2 SECTION .data
3 msg1 db "Введите N: ",0h
4 SECTION .bss
5 N resb 10
6 SECTION .text
7 global _start
8 _start:
9 ; ---- Вывод сообщения "Введите N: "
10 mov eax,msg1
11 call sprint
12 ; ---- Ввод 'N'
13 mov ecx, N
14 mov edx, 10
15 call sread
16 ; ---- Преобразование 'N' из символа в число
17 mov eax,N
18 call atoi
19 mov [N],eax
20 ; ---- Организация цикла
21 mov ecx,[N] ; Счетчик цикла, 'ecx-N'
22 label:
23 sub ecx,1
24 mov [N],ecx
25 mov eax,[N]
26 call iprintLF ; Вывод значения 'N'
27 loop label ; 'ecx-ecx-1' и если 'ecx' не '0'
28 ; переход на 'label'
29 call quit
```

Изменение файла

Создаю исполняемый файл и проверяю его работу.(рис. [??]).

```
4294965500
4294965498
4294965496
4294965494
4294965492
4294965490
4294965488
4294965486
4294965484
4294965482
4294965480
4294965478
4294965476
```

Проверка работы файла

В данном случае число проходов цикла не соответствует значению N.

Вношу изменения в текст программы, добавив команды push и pop.(рис. [??]).

```
~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab08/report/lab8-1.asm - Mousepad
File Edit Search View Document Help

1 %include "in_out.asm"
2 SECTION .data
3 msg1 db "Введите N: ",0h
4 SECTION .bss
5 N resb 10
6 SECTION .text
7 global _start
8 _start:
9 ; ---- Вывод сообщения "Введите N: "
10 mov eax,msg1
11 call sprint
12 ; ---- Ввод 'N'
13 mov ecx, N
14 mov edx, 10
15 call sread
16 ; ---- Преобразование 'N' из символа в число
17 mov eax,N
18 call atoi
19 mov [N],eax
20 ; ---- Организация цикла
21 mov ecx,[N] ; Счетчик цикла, 'ecx-N'
22 label:
23 push ecx
24 sub ecx,1
25 mov [N],ecx
26 mov eax,[N]
27 call iprintLF ; Вывод значения 'N'
28 pop ecx
29 loop label ; 'ecx-ecx-1' и если 'ecx' не '0'; переход на 'label'
30 call quit
```

Внесение изменений в текст программы

Создаю исполняемый файл и запускаю его.(рис. [??]).

```
(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ nasm -f elf lab8-1.asm

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ld -m elf_i386 -o lab8-1 lab8-1.o

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ./lab8-1
Введите N: 10
9
x не 0; переход на `label`
8
7
6
5
4
3
2
1
0
```

Запуск программы

В данном случае число проходов цикла соответствует значению N.

3.2 Обработка аргументов командной строки

Создаю файл lab8-2.asm и проверяю его создание.(рис. [??]).

```
(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ touch lab8-2.asm

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ls
Makefile  bib  image  in_out.asm  lab8-1  lab8-1.asm  lab8-1.o  lab8-2.asm  pandoc  report.md
```

Создание файла lab8-2.asm

Ввожу в этот файл программу из листинга 8.2.(рис. [??]).

```
1;-----
2; Обработка аргументов командной строки
3;-----
4#include 'in_out.asm'
5SECTION .text
6global _start
7_start:
8pop ecx ; Извлекаем из стека в 'ecx' количество
9; аргументов (первое значение в стеке)
10pop edx ; Извлекаем из стека в 'edx' имя программы
11; (второе значение в стеке)
12sub ecx, 1 ; Уменьшаем 'ecx' на 1 (количество
13; аргументов без названия программы)
14next:
15cmp ecx, 0 ; проверяем, есть ли еще аргументы
16jz _end ; если аргументов нет выходим из цикла
17; (переход на метку '_end')
18pop eax ; иначе извлекаем аргумент из стека
19call sprintf ; вызываем функцию печати
20loop next ; переход к обработке следующего
21; аргумента (переход на метку 'next')
22_end:
23call quit
```

Ввод программы из листинга 8.2

Создаю исполняемый файл и запускаю его, указав аргументы.(рис. [??]).

```

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab08/report]
$ nasm -f elf lab8-2.asm

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab08/report]
$ ld -m elf_i386 -o lab8-2 lab8-2.o

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab08/report]
$ ./lab8-2

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab08/report]
$ ./lab8-2 a b c
a
b
c

```

Запуск программы

Программа обработала 3 аргумента.

Создаю файл lab8-3.asm и ввожу в него программу из листинга 8.3.(рис. [??]).

```

File Edit Search View Document Help
~\work\study\2023-2024\Архитектура компьютера\arch-pc\labs\lab08\report\lab8-3.asm - Mousepad

1 %include 'in_out.asm'
2 SECTION .data
3 msg db "Результат: ",0
4 SECTION .text
5 global _start
6 _start:
7 pop ecx ; Извлекаем из стека в 'ecx' количество
8 ; аргументов (первое значение в стеке)
9 pop edx ; Извлекаем из стека в 'edx' имя программы
10 ; (второе значение в стеке)
11 sub ecx,1 ; Уменьшаем 'ecx' на 1 (количество
12 ; аргументов без названия программы)
13 mov esi,0 ; Используем 'esi' для хранения
14 ; промежуточных сумм
15 next:
16 cmp ecx,0h ; проверяем, есть ли еще аргументы
17 jz _end ; если аргументов нет выходим из цикла
18 ; (переход на метку '_end')
19 pop eax ; иначе извлекаем следующий аргумент из стека
20 call atoi ; преобразуем символ в число
21 add esi,eax ; добавляем к промежуточной сумме
22 ; след. аргумент 'esi=esi+eax'
23 loop next ; переход к обработке следующего аргумента
24 _end:
25 mov eax,msg ; вывод сообщения "Результат: "
26 call _printf
27 mov eax,esi ; записываем сумму в регистр 'eax'
28 call _printf ; печать результата
29 call _quit ; завершение программы

```

Ввод программы из листинга 8.3

Созаю исполняемый файл и запускаю его, указав аргументы.(рис. [??]).

```

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab08/report]
$ nasm -f elf lab8-3.asm

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab08/report]
$ ld -m elf_i386 -o lab8-3 lab8-3.o

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab08/report]
$ ./lab8-3 5 5 5 4
Результат: 19

```

Проверка работы файла

Программа работает.

Теперь изменяю текст программы из листинга 8.3 так, чтобы она вычисляла произведение аргументов каждой строки.(рис. [??]).

```
~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab08/report/lab8-3.asm - Mousepad
File Edit Search View Document Help

1 %include "in_out.asm"
2 SECTION .data
3 msg db "Результат: ",0
4 SECTION .text
5 global _start
6 _start:
7 pop ecx
8 pop edx
9 sub ecx,1
10 mov esi,1
11 next:
12 cmp ecx,0h
13 jz end
14 pop eax
15 call atoi
16 mul esi
17 mov esi,eax
18 loop next
19 end
20 mov eax,msg
21 call sprint
22 mov eax,esi
23 call iprintLF
24 call quit
```

Изменение текста листинга 8.3

Создаю исполняемый файл и запускаю его, указав аргументы.(рис. [??]).

```
(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ nasm -f elf lab8-3.asm

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ld -m elf_i386 -o lab8-3 lab8-3.o

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ./lab8-3 2 2 3
Результат: 12
```

Проверка работы программы

#Задания для самостоятельной работы

Создаю файл test.asm.(рис. [??]).

```
(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ touch test.asm

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ls
Makefile bin image in_out.asm lab8-1 lab8-1.asm lab8-1.o lab8-2 lab8-2.asm lab8-2.o lab8-3 lab8-3.asm lab8-3.o pandoc report.md test.asm
```

Написание программы для самостоятельной работы

Начинаю написание программы, которая будет вычислять сумму значений $f(x)=3(10+x)$.(вариант 20).(рис. [??]).

```
~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab08/report/test.asm - Mousepad
File Edit Search View Document Help

1 %include "in_out.asm"
2 SECTION .data
3 msg db "Результат: ",0
4 SECTION .text
5 global _start
6 _start:
7 pop ecx
8 pop edx
9 sub ecx,1
10 mov esi,0
11 mov edi,3
12 next:
13 cmp ecx,0h
14 jz end
15 pop eax
16 call atoi
17 add eax,10
18 mul edi
19 add esi,eax
20 loop next
21 end
22 mov eax,msg
23 call sprint
24 mov eax,esi
25 call iprintLF
26 call quit
```

Написание программы для самостоятельной работы

Создаю исполняемый файл и запускаю его, указав аргументы.(рис. [??]).

```
(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ nasm -f elf test.asm

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ld -m elf_i386 -o test test.o

(nvnovikov1@nvnovikov1)-[~/.../arch-pc/labs/lab08/report]
$ ./test 5 2 1 7
Результат: 165
```

Запуск программы

Программы работает.

4 Выводы

После выполнения данной лабораторной работы я приобрел навыки написание программ с использованием циклов и обработкой аргументов командной строки