Отчёт по лабораторной работе №7

Дисциплина: архитектура компьютера

Алиев Эльхан

Содержание

# 1 Цель работы

Изучение команд условного и безусловного переходов. Приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга

# 2 Выполнение лабораторной работы

**1**

С помощью утилиты mkdir создаю директорию lab07, перехожу в нее и создаю файл для работы.

**2**

Копирую в текущий каталог файл in\_out.asm из загрузок, т.к. он будет использоваться в других программах .

**3**

Открываю созданный файл lab7-1.asm, вставляю в него программу реализации безусловных переходов.

**4**

Создаю исполняемый файл программы и запускаю его . Инструкции jmp \_label2 меняет порядок исполнения инструкций и позволяет выполнить инструкции начиная с метки \_label2.

**5**

Изменяю текст программы так, чтобы она выводила сначала ‘Сообщение № 2’, потом ‘Сообщение № 1’ и завершала работу

**6**

Создаю новый исполняемый файл программы и запускаю его . Убеждаюсь в том, программа раотает верно.

**7**

Изменяю текст программы, так чтобы вывод происходил в обратном порядке

**8**

Создаю исполняемый файл и проверяю работу программы . Программа отработало верно.

**9**

Создаю новый файл lab7-2.asm для программы с условным оператором.

**10**

Вставляю программу, которая определяет и выводит на экран наибольшее число.

**12**

Создаю файл листинга для программы в файле lab7-2.asm .

**13**

Открываю файл листинга с помощью редактора mcedit. Расмотрим 9-11 строки:

9 строка:

* Перые цифры [9] - это номер строки файла листинга.
* Cледующие цифры [00000006] адрес — это смещение машинного кода от начала текущего сегмента, состоит из 8 чисел.
* следующие числа [7403] - это машинный код, который представляет собой ассемблированную исходную строку в виде шестнадцатеричной последовательности, поэтоу и появляются буквы латынского алфавита.
* следющее [jz finished] - исходный текст программы, которая просто состоит из строкк исходной программы вместе с комментариями.

10 строка:

* Перые цифры [10] - это номер строки файла листинга.
* Cледующие цифры [00000008] адрес — это смещение машинного кода от начала текущего сегмента, состоит из 8 чисел.
* следующие числа [40] - это машинный код, который представляет собой ассемблированную исходную строку в виде шестнадцатеричной последовательности, поэтоу и появляются буквы латынского алфавита.
* следющее [inc eax] - исходный текст программы, которая просто состоит из строкк исходной программы вместе с комментариями

11 строка:

* Перые цифры [11] - это номер строки файла листинга.
* Cледующие цифры [00000009] адрес — это смещение машинного кода от начала текущего сегмента, состоит из 8 чисел.
* следующие числа [EBF8] - это машинный код, который представляет собой ассемблированную исходную строку в виде шестнадцатеричной последовательности, поэтоу и появляются буквы латынского алфавита.
* следющее [jmp nextchar] - исходный текст программы, которая просто состоит из строкк исходной программы вместе с комментариями

**14**

Открываю файл lab7-2.asm с помощью редактора и Удаляю один операнд в инструкции cmp.

**15**

Открываю файл листинга с помощью редактора mcedit и замечаю, что в файле листинга появляется ошибка.

Отсюда можно сделать вывод, что, если в коде появляется ошибка, то ее описание появится в файле листинга

# 3 Сомастоятельная работа

**1**

Создаю файл lab7.asm с помощью утилиты touch

**2**

Ввожу в созданный файл текст программы для вычисления наибольшего из 3 чисел. Числа беру, учитывая свой вариант из прошлой лабораторной работы. 2 вариант

**3**

Создаю исполняемый файл и запускаю его

**Текст программы**

%include ‘in\_out.asm’ section .data msg1 db ’ а = ‘,0h msg2 db’ b = ‘,0h msg3 db’ c = ‘,0h msg4 db “Наибольшее число:”,0h a dd ’95’ b dd ‘2’ c dd ‘61’

section .bss max resb 10

section .text global \_start \_start: ; ———- Вывод всех чисел: mov eax,msg1 call sprint mov eax,a call atoi call iprintLF

mov eax,msg2 call sprint mov eax,b call atoi call iprintLF

mov eax,msg3 call sprint mov eax,c call atoi call iprintLF

;————-сравнивание чисел mov eax,b call atoi ;перевод символа в число mov [b],eax ; запись преобразованного числа в b ;———— запись b в переменную мах mov ecx,[a] ; mov [max],ecx ; ;————сравнивание чисел a c cmp ecx,[c]; if a>c jg check\_b ; то перход на метку mov ecx,[c] ; mov [max],ecx ; ;——-метка check\_b

**4**

Создаю новый файл lab77.asm для написания программы второго задания.

**5**

Ввожу в него программу, в которую ввожу 2 значения x и a, и которая выводит значения функции. Функцию беру из таблицы в соответствии со своим вариантом

**6**

Создаю испольняемый файл и проверяю её выполнение при x=1, a=2 Программа отработала верно!

**7**

Повторный раз запускаю программу и проверяю ее выполнение при x=2 и a=1 Программа отработала верно!

**Текст программы** %include ‘in\_out.asm’ section .data msg1 db ‘Введите x:’,0h msg2 db ‘Введите a:’,0h msg3 db ‘f(x) =’,0h

section .bss x resb 10 a resb 10

section .text global \_start \_start: mov eax,msg1 call sprint mov ecx,x mov edx,10 call sread mov eax,x ;———- call atoi mov [x],eax ;———–

mov eax,msg2 call sprint mov ecx,a mov edx,10 call sread mov eax,a ; call atoi mov [a],eax ; ;———- mov ecx,[a] cmp ecx,[x] ;x<a jg check\_a ; mov eax,[a] mov ebx,-1 mul ebx mov ecx,[x] add ecx,eax jmp \_end check\_a: mov ecx,5; \_end: mov eax,msg3 ; call sprint ; mov eax,ecx ;

# 4 Выводы

При выполнении данной лабораторной работы я освоил инструкции условного и безусловного вывода и ознакомился с структурой файла листинга.ы