

Лабораторное занятие №5

Изучение процесса разработки циклов с условием на языке ассемблера

1 Цель работы:

1.1 Приобрести навыки разработки разветвляющихся алгоритмов на языке ассемблера.

2 Литература:

2.1 Приложение п.9.

3 Оборудование:

3.1 Персональный компьютер;

4 **Подготовка к работе:** 4.1 Изучить приложение

5 Задание:

5.1 Выполнить задания из пункта 6.

5.2 Составить электронный отчет.

5.3 Ответить на контрольные вопросы.

6 Порядок выполнения работы:

Код для последующих заданий должен быть написан в ассемблерных вставках на `masm`! Для сравнения использовать команду `cmp`, а для ветвления команды перехода! **Обязательно ознакомьтесь с приложением.**

6.1 Написать программу, выводящую частное двух чисел. Пользователь вводит делитель, если делитель равен нулю, повторить ввод, иначе вывести результат. Использовать цикл с предусловием.

6.2 Написать программу с использованием цикла с постусловием, в которой пользователь пытается угадать число. Если введено число меньше или больше загаданного, дать возможность заново угадать (повторять до тех пор, пока не будет названо корректное число).

6.3 Написать программу с использованием цикла с предусловием, запрашивающую у пользователя сумму, на которую он хочет открыть вклад, и процент годовых. Вывести на экран, через сколько лет он станет миллионером и сумму на вкладе за каждый год. Ежегодно размер вклада увеличивается на указанный процент, на эти деньги в следующем году также будут начислены проценты

7 Содержание отчёта:

7.1 Титульный лист.

7.2 Цель работы.

7.3 Последовательность действий по выполнению задания.

7.4 Ответить на контрольные вопросы.

7.5 Вывод по проделанной работе.

8 Контрольные вопросы:

8.1 Как отследить значения регистров в Visual Studio?

8.2 Какие команды перехода вы знаете?

8.3 С помощью какой команды можно сравнить значения в регистрах?

9 Приложение

["X:\Абрамова\Системное программирование\Лекции\Ветвление на ассемблере.pdf"](#)

Пример цикла со счетчиком на ассемблере:

```
_asm {
    xor ecx, ecx;
    mov ecx, 5; // цикл выполнится 5 раз
    mov eax, 0;
    jz EXIT
SYCLE:
    add eax, 3;
    dec ecx // ecx--
    jnz SYCLE
    mov res, eax
EXIT:
}
```

Пример ввода и вывода числа на ассемблере

```
int scan(int a){
    cin >> a;
    return a;
}

void print(int a){
    cout << a;
}

int _tmain(int argc, _TCHAR* argv[])
{
    __asm{
        //пример ввода числа в ассемблерной вставке
        push eax //добавление eax в стек
        call scan //вызов функции для считывания числа
        pop eax //извлечение eax из стека (теперь в eax значение, введенное пользователем)

        //пример вывода числа в ассемблерной вставке
        push eax //добавление eax в стек
        call print //вызов функции для вывода числа
        pop eax //извлечение eax из стека (на консоль вывелось значение из eax)
    }

    return 0;
}
```