

Архитектура ЭВМ и язык ассемблера

Семинар #32:

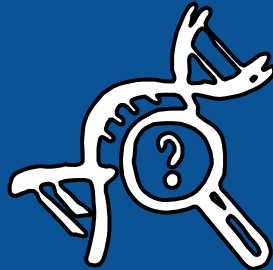
1. Организация цикла на условных переходах.
2. Компактное кодирование цикла, инструкция LOOP.
3. Задачи:
 - Вычисление максимального числа в массиве.
 - Конвертация числа в HEX.

22.04.2025

Организация цикла на условных переходах



Компактное кодирование циклов, инструкция LOOP



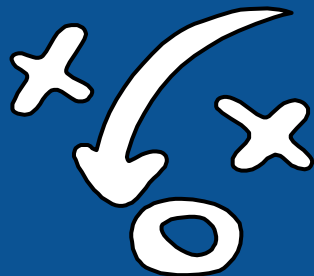
Инструкция LOOP

Команда	Описание
JCXZ/JECXZ	Переход выполняется, если значение регистра CX/ECX равно нулю.
LOOP	Переход выполняется, если значение регистра ECX не равно нулю.
LOOPZ/LOOPE	Переход выполняется, если значение регистра ECX не равно нулю и флаг ZF установлен.
LOOPNZ/LOOPNE	Переход выполняется, если значение регистра ECX не равно нулю и флаг ZF сброшен.

LOOP/LOOPcc—Loop According to ECX Counter

Opcode	Instruction	Op/ En	64-Bit Mode	Compat/ Leg Mode	Description
E2 cb	LOOP rel8	D	Valid	Valid	Decrement count; jump short if count $\neq 0$.
E1 cb	LOOPE rel8	D	Valid	Valid	Decrement count; jump short if count $\neq 0$ and ZF = 1.
E0 cb	LOOPNE rel8	D	Valid	Valid	Decrement count; jump short if count $\neq 0$ and ZF = 0.

Задачи

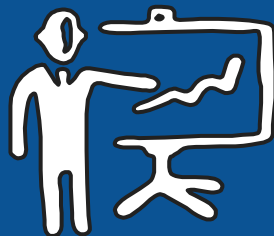


Сохранение регистров перед вызовами

Table 2.3: Register Usage

Register	Usage	Preserved across function calls
%eax	scratch register; also used to return integer and pointer values from functions; also stores the address of a returned struct or union	No
%ebx	callee-saved register; also used to hold the GOT pointer when making function calls via the PLT	Yes
%ecx	scratch register	No
%edx	scratch register; also used to return the upper 32bits of some 64bit return types	No
%esp	stack pointer	Yes
%ebp	callee-saved register; optionally used as frame pointer	Yes
%esi	callee-saved register	yes
%edi	callee-saved register	yes

Вопросы?



Красивые иконки взяты с сайта handdrawngoods.com