***Задание на Лекцию № 15. 03.12.2024.***

***НА СЛЕДУЮЩЕЙ НЕДЕЛЕ 10.12.2024 года НА ЛЕКЦИИ БУДЕТ КОНТРОЛЬНАЯ – РЕШЕНИЕ КАЖДЫМ СВОЕЙ ЗАДАЧИ ИЗ СПИСКА ЗАДАЧ, который передан Вам в прошлом письме!!!! НА КОНТРОЛЬНОЙ МОЖНО ПОЛЬЗОВАТЬСЯ ВСЕМ ЧЕМ УГОДНО!!!!***

***Чтобы хорошенько подготовиться к контрольной ЗУБАРИМ ТО, ЧТО ПРЕДСТАВЛЕНО ДАЛЕЕ:***

1. Досконально разбираемся в текстах представленных далее на языке записи алгоритмов ассемблер, которые были получены когда-то в рамках курса БИС(частично эти тексты попали в Ваши билеты):

1.1.

.MODEL SMALL

.CODE

.STARTUP

MOV CX,3999

MOV AL,'$'

PRINT:

INT 29H

LOOP PRINT

MOV AX,40H

MOV ES,AX

MOV BH,ES:[62H]

MOV AH,0AH

MOV AL,'$'

MOV CX,1

INT 10H

MOV AX,4C00H

INT 21H

END

1.2.

.MODEL SMALL

.CODE

JMP\_OLD\_HANDLER:

JMP\_COMMAND DB 0EAH

JMP\_OFFSET DW ?

JMP\_SEGMENT DW ?

NEW\_HANDLER:

CMP AH,30H

JNE JMP\_OLD\_HANDLER

MOV AX,050CH

IRET

START:

MOV AX,CS

MOV DS,AX

MOV AX,3521H

INT 21H

CMP BX,OFFSET NEW\_HANDLER

JE ALREADY\_LOADED

MOV CS:JMP\_OFFSET,BX

MOV CS:JMP\_SEGMENT,ES

MOV DX,OFFSET NEW\_HANDLER

MOV AX,2521H

INT 21H

MOV AH,62H

INT 21H

MOV DX,CS

SUB DX,BX

MOV AX,OFFSET START

ADD AX,15

MOV CL,4

SHR AX,CL

ADD DX,AX

MOV AX,3100H

INT 21H

ALREADY\_LOADED:

MOV DX,OFFSET ALREADY\_LOADED\_TEXT

MOV AH,09H

INT 21H

MOV AX,4C00H

INT 21H

ALREADY\_LOADED\_TEXT DB 'The handler is already loaded!$'

END START

1.3.

.MODEL SMALL

.CODE

START:

MOV AX,ES:[2CH]

MOV DS,AX

MOV SI,-1

SEARCH\_01:

ADD SI,1

MOV AL,[SI]

CMP AL,0

JNE SEARCH\_01

MOV AL,[SI+1]

CMP AL,1

JNE SEARCH\_01

ADD SI,2

SEARCH\_START:

ADD SI,1

MOV AL,[SI]

CMP AL,'\'

JNE SKIP\_SAVE

MOV DI,SI

SKIP\_SAVE:

CMP AL,0

JNE SEARCH\_START

PRINT\_NAME:

ADD DI,1

MOV AL,[DI]

CMP AL,0

JE FINISH

INT 29H

JMP PRINT\_NAME

FINISH:

MOV AX,4C00H

INT 21H

END START

1.4.

.MODEL SMALL

.DATA

FONT DB 16 DUP(?)

.CODE

.STARTUP

MOV BX,OFFSET FONT

MOV BYTE PTR [BX], 01000100B

MOV BYTE PTR [BX+1], 01000100B

MOV BYTE PTR [BX+2], 11101110B

MOV BYTE PTR [BX+3], 10101010B

MOV BYTE PTR [BX+4], 10101010B

MOV BYTE PTR [BX+5], 10001000B

MOV BYTE PTR [BX+6], 11101110B

MOV BYTE PTR [BX+7], 00100010B

MOV BYTE PTR [BX+8], 10101010B

MOV BYTE PTR [BX+9], 10101010B

MOV BYTE PTR [BX+10],11101110B

MOV BYTE PTR [BX+11],01000100B

MOV BYTE PTR [BX+12],01000100B

MOV BYTE PTR [BX+13],00000000B

MOV BYTE PTR [BX+14],00000000B

MOV BYTE PTR [BX+15],00000000B

MOV AX,DS

MOV ES,AX

MOV BP,OFFSET FONT

MOV CX,1

MOV DX,'$'

MOV BX,1600H

MOV AX,1100H

INT 10H

MOV CX,1999

MOV AL,'$'

PRINT:

INT 29H

LOOP PRINT

MOV AX,40H

MOV ES,AX

MOV BH,ES:[62H]

MOV AH,0AH

MOV AL,'$'

MOV CX,1

INT 10H

ZU:

JMP ZU

MOV AX,4C00H

INT 21H

END

1.5.

.MODEL SMALL

.STACK 100H

.DATA

PROGRAM\_NAME DB 256 DUP(?)

.CODE

PRINT\_NUMBER PROC

MOV CX,0

PRINT\_SYMBOL:

MOV DX,0

MOV BX,10000

DIV BX

CMP AX,0

JNE CONTINUE\_PRINTING

CMP CX,0

JE SKIP\_PRINTING

CONTINUE\_PRINTING:

ADD AL,30H

INT 29H

MOV CX,1

SKIP\_PRINTING:

MOV AX,DX

MOV BX,10

MUL BX

CMP AX,0

JNE PRINT\_SYMBOL

CMP CX,0

JNE SKIP\_ZERO

ADD AL,30H

INT 29H

SKIP\_ZERO:

RET

PRINT\_NUMBER ENDP

START:

MOV AX,@DATA

MOV DS,AX

MOV AX,ES:[2CH]

MOV ES,AX

MOV SI,-1

SEARCH\_01:

ADD SI,1

MOV AL,ES:[SI]

CMP AL,0

JNE SEARCH\_01

MOV AL,ES:[SI+1]

CMP AL,1

JNE SEARCH\_01

ADD SI,2

MOV BX,OFFSET PROGRAM\_NAME

COPY\_NAME:

ADD SI,1

MOV AL,ES:[SI]

MOV [BX],AL

INC BX

CMP AL,0

JNE COPY\_NAME

MOV DX,OFFSET PROGRAM\_NAME

MOV AX,3D00H

INT 21H

MOV BX,AX

MOV AH,3EH

INT 21H

MOV AX,BX

CALL PRINT\_NUMBER

MOV AX,4C00H

INT 21H

END START

1. Используя ДУРКование, DOSBOX, оттранслировать, слинковать все указанные тексты на языке записи алгоритмов, осуществить исправление всех синтаксических и семантических ошибок, получить исполнимые модули И ПОЛУЧИТЬ РЕЗУЛЬТАТ – Вы должны увидеть на экране НЕЧТО. При этом процесс записи, трансляции, линковки, получения исполнимого модуля и ИСПОЛНЕНИЯ ПРОГРАММЫ ДОЛЖЕН БЫТЬ ЗАДОКУМЕНТИРОВАН В ВИДЕ ПОСЛЕДОВАТЕЛЬНОСТИ СКРИН-ШОТОВ. По каждой из ПЯТИ ПРОГРАММ. К этому отчёту, как обычно, приклеить 5 исходных текстов и 5 исполнимых модулей и прислать их мне на почту в виде заархивированного с паролем файла (ПАРОЛЬ ZukaISS2024).
2. После получения РЕЗУЛЬТАТОВ НЕОБХОДИМО РАЗОБРАТЬСЯ В ФУНКЦИОНАЛЕ КАЖДОЙ ИЗ ПРОГРАММ. А именно: в отдельном Word-файле записать УСЛОВИЕ ЗАДАЧИ, которая РЕШАЕТСЯ в каждом из пяти случаев, НА РУССКОМ ЯЗЫКЕ – как будто бы Вы задали эти задачи студенту и именно так, как бы Вы хотели, чтобы Вам формулировалось то или задание в рамках курсов по «программированию». Полученные пять условий прислать мне на почту не позднее 10.12.2024.

P.S. В других прикреплённых файлах другие примеры программ, которые могут пригодиться как на контрольной, так и на экзамене. Свинья 2 к экзамену говорит, что делается в Свинье 3, а Свинья 1 должна быть изучена и разобрана до мелочей, чтобы было понятно, что происходит при её запуске в качестве программы.