Assignment no.8

Name:- Gaurav Patil

Roll No. - 21259

Batch – G2

Title - Write X86/64 ALP to perform non-overlapped/ Overlapped block transfer without string specific instructions. Block containing data can be defined in the data segment.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* C O D E \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

%macro print 2

mov rax,01

mov rdi,01

mov rsi,%1

mov rdx,%2

syscall

%endmacro

section .data

array db 10h,20h,30h,40h,50h

msg1: db 'Before overlapped :',0xa

len1: equ $-msg1

msg2: db 'After overlapped :',0xa

len2: equ $-msg2

msg3: db ' ',0xa

len3: equ $-msg3

msg4: db ' : '

len4: equ $-msg4

count db 0

count1 db 0

count2 db 0

count3 db 0

count4 db 0

count5 db 0

section .bss

addr resb 16

num1 resb 2

section .text

global \_start

\_start:

print msg1, len1

xor rsi,rsi

mov rsi,array

mov byte[count],05

up:

mov rbx,rsi

push rsi

mov rdi,addr

call HtoA1

pop rsi

mov dl,[rsi]

push rsi

mov rdi,num1

call HtoA2

pop rsi

inc rsi

dec byte[count]

jnz up

mov rsi,array

mov rdi,array+0Ah

mov byte[count3],05h

loop10:

mov dl,00h

movsb

dec byte[count3]

jnz loop10

xor rsi,rsi

mov rsi,array+3h

mov rdi,array+0Ah

mov byte[count5],05h

loop11:

mov dl,byte[rdi]

mov byte[rsi],dl

inc rsi

inc rdi

dec byte[count5]

jnz loop11

print msg2, len2

xor rsi,rsi

mov rsi,array

mov byte[count4],08h

up10:

mov rbx,rsi

push rsi

mov rdi,addr

call HtoA1

pop rsi

mov dl,[rsi]

push rsi

mov rdi,num1

call HtoA2

pop rsi

inc rsi

dec byte[count4]

jnz up10

Exit:

mov rax,60

mov rdx,00

syscall

; ================ SUBROUTINES ==========================

HtoA1:

mov byte[count1],16

dup1:

rol rbx,4

mov al,bl

and al,0fh

cmp al,09

jg p3

add al,30h

jmp p4

p3: add al,37h

p4:mov [rdi],al

inc rdi

dec byte[count1]

jnz dup1

print addr,16

print msg4, len4

ret

HtoA2:

mov byte[count2],02

dup2:

rol dl,04

mov al,dl

and al,0fh

cmp al,09h

jg p31

add al,30h

jmp p41

p31: add al,37h

p41:mov [rdi],al

inc rdi

dec byte[count2]

jnz dup2

print num1, 02

print msg3, len3

ret

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* O U T P U T \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

