%macro scall 4

mov rax,%1

mov rdi,%2

mov rsi,%3

mov rdx,%4

%endmacro

section .data

nline db 10,10

nlinle\_len equ $-nline

m1 db 10,"Enter first 2 hex digit numbers:"

l1 equ $-m1

m2 db 10,"Enter next 2 hex digit numbers:"

l2 equ $-m2

m3 db 10,"Result of successive addition:"

l3 equ $-m3

section .bss

buf resb 3

buf\_len equ $-buf

no1 resw 1

no2 resw 1

ansl resw 1

ansh resw 1

ans resd 1

char\_ans resb 4

section .text

global \_start

\_start:

scall 1,1,m1,l1

call accept\_16

mov [no1],bx

scall 1,1,m2,l2

call accept\_16

mov [no2],bx

call SA

mov rax,60

mov rdi,0

syscall

SA:

mov rbx,[no1]

mov rcx,[no2]

xor rax,rax

xor rdx,rdx

back:

add rax,rbx

jnc next

inc rdx

next:

dec rcx

jnz back

mov [ansl],rax

mov [ansh],rdx

scall 1,1,m3,l3

mov ax,[ansh]

call display\_16

ret

accept\_16:

scall 0,0,buf,buf\_len

xor bx,bx

mov rcx,2

mov rsi,buf

next\_digit:

shl bx,04

mov al,[rsi]

cmp al,39h

jbe L1

sub al,07h

L1:

sub al,30h

add bx,ax

inc rsi

loop next\_digit

ret

display\_16:

mov rsi,char\_ans

mov rcx,4

cnt:

rol ax,04

mov bl,al

and bl,0fh

cmp bl,09h

jbe L2

add bl,07h

L2:

add bl,30h

mov [rsi],bl

inc rsi

dec rcx

jnz cnt

scall 1,1,char\_ans,4

ret

