

VIỆN ĐIỆN TỬ VIỄN THÔNG
BỘ MÔN ĐIỆN TỬ VÀ KTMT

MÔN HỌC: KỸ THUẬT VI XỬ LÝ
Học kỳ: 2 Năm học: 2017-2018

BÀI TẬP VỀ NHÀ SỐ 4

Sinh viên: Lương Văn Minh
MSSV: 20152445
Lớp: Điện tử 06 – K60

Bài 1 (15 điểm)

Viết lại hàm C sau đây bằng hợp ngữ cho 8086 (tham số của hàm truyền qua stack, giá trị trả về lưu trong thanh ghi chứa)

```
/* Hàm tính số bit 1 của số nguyên 2 byte */  
int Count(int a)  
{  
    int s = 0;  
    while (a != 0) {  
        if ((a & 1) != 0)  
            s++;  
        a >>= 1;  
    }  
    return s;  
}
```

Hàm viết lại bằng ASM:

```
count proc  
    pop bx ; Lay ra IP tu stack  
    pop ax ; Lay ra tham so truyen vao tu stack  
    push bx ; Cat lai IP vao stack  
    mov cx,16 ; 2 bytes = 16 bits  
    xor dx,dx ; dx = 0  
    ; dx chua so bit 1 cua so nguyen 2 bytes  
dich:  
    shr ax,1  
    jnc bit0  
    inc dx  
    bit0:  
    loop dich  
    ret ; tro lai chuong trinh chinh  
count endp
```

Bài 2 (20 điểm) (yêu cầu làm trên máy tính và in mã nguồn cùng kết quả chạy chương trình)

.Hãy viết chương trình assembly cho 8086 thực hiện:

1. Nhập số N (N nguyên dương, $N < 9$) từ bàn phím
2. Tính $S = 1^2 + 2^2 + 3^2 + \dots + N^2$
3. Kiểm tra tính chẵn lẻ của S và đưa kết quả ra màn hình

Mã nguồn:

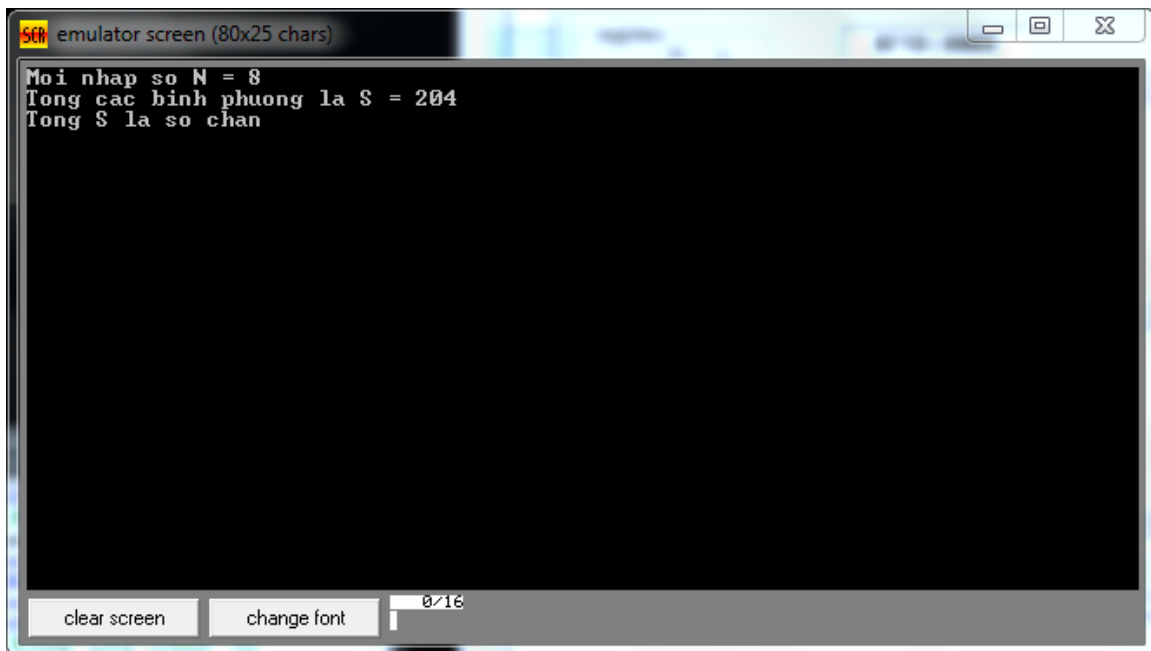
```
.model small
.stack 100
.data
    msg1 db 'Moi nhap so N = $'
    msg2 db 10,13,'Tong cac binh phuong la S = $'
    msg3 db 10,13,'Tong S la so chan$'
    msg4 db 10,13,'Tong S la so le$'
    flag db 0
.code
main proc
    ; Init DS
    mov ax,@data
    mov ds,ax
    ; Thong bao nhap N
    mov ah,9
    lea dx,msg1
    int 21h
    ; Nhap N
    mov ah,1
    int 21h
    ; Tinh tong
    sub al,30h
    mov cl,al
    xor ch,ch
    xor bx,bx
tinhtong:
    mov ax,cx
    mul cx
    add bx,ax
    loop tinhtong
    ; Hien thi tong
    mov ah,9
    lea dx,msg2
    int 21h
    ; Hien thi hang tram
    mov ax,bx
    mov bl,100
    div bl
    mov bl,ah
    cmp al,0
    jz hangchuc
    mov flag,1
    mov dl,al
    add dl,30h
```

```

mov ah,2
int 21h
; Hien thi hang chuc
hangchuc:
mov al,bl
xor ah,ah
mov bl,10
div bl
mov bl,ah
cmp flag,1
jz _chuc
cmp al,0
jz hangdonvi
_chuc:
mov dl,al
add dl,30h
mov ah,2
int 21h
; Hien thi hang don vi
hangdonvi:
mov dl,bl
add dl,30h
mov ah,2
int 21h
; Kiem tra chan le
mov ah,9
test dl,1
jz tong_chan
lea dx,msg4
jmp hienthi_chanle
tong_chan:
lea dx,msg3
; Hien thi chan le
hienthi_chanle:
int 21h
main endp
end code

```

Kết quả chạy chương trình:



Bài 3 (15 điểm)

Viết lại hàm C sau đây bằng hợp ngữ cho 8086 (tham số của hàm truyền qua stack, giá trị trả về lưu trong thanh ghi chứa)

/* Hàm tính tổng các số chia hết cho 4 của mảng N số nguyên 2 byte */

```
int Sum(int A[], int N)
{
    int s = 0;
    for (int i = 0; i < N; i++)
        if (A[i] % 4 == 0)
            s += A[i];
    return s;
}
```

Hàm viết lại bằng ASM:

```
sum proc
    pop si ; Lay ra IP tu stack
    pop cx ; Lay ra so phan tu N
    xor dx,dx ; dx = 0
    ; dx chua tong cac so chia het cho 4
    tong:
        pop bx ; Lay ra tung phan tu cua mang trong stack
        mov ax,bx
        xor ah,ah
        mov ch,4
        div ch
        cmp ah,0
        jnz chia_du
        add dx,bx
```

```

        xor ch,ch
        chia_du:
        loop tong
        push si ; Cat lai IP vao stack
sum endp

```

Bài 4 (15 điểm)

Viết lại hàm C sau đây bằng hợp ngữ cho 8086 (tham số của hàm truyền qua stack, giá trị trả về lưu trong thanh ghi chứa)

/* Hàm tính tổng các chữ số của một số cho bởi xâu ký tự */

```

int Sum(char A[])
{
    int s = 0;
    for (int i = 0; A[i] != 0; i++)
        s += A[i] - 48;
    return s;
}

```

Hàm viết lại bằng ASM:

```

sum proc
    pop si ; Lay ra IP trong stack
    xor dx,dx ; dx = 0
    ; dx chua tong cac chu so
    tinhcong:
    pop ax ; Lay ra tung ky tu
    cmp ax,0 ; Kiem tra neu la NULL (0)
    jz kt_tinhcong
    sub ax,30h
    add dx,ax
    jmp tinhcong
    kt_tinhcong:
    push si ; Cat lai IP vao stack
sum endp

```

Bài 5 (20 điểm)

Viết đoạn chương trình assembly thực hiện:

1. Nhập hai số nguyên a và n từ bàn phím, kiểm tra điều kiện $0 < a, n < 10$. Nếu không thỏa mãn điều kiện thì thoát khỏi chương trình.
2. Tính a^n
3. Đưa kết quả ra màn hình sử dụng hàm Xuất cho dưới đây

Hàm Xuất in một giá trị có độ dài thập phân bất kỳ ra màn hình	Đầu vào: AX chứa giá trị của số cần hiển thị Sử dụng: CALL Xuất
--	--

Mã nguồn chương trình:

```
.model small
.stack 100
.data
    msg1 db 'Nhap so a (0 < a < 10): $'
    msg2 db 10,13,'Nhap so n (0 < n < 10): $'
    err  db 10,13,'So vua nhap khong hop le! Chuong trinh ket
thuc!$'
    res  db 10,13,'Ket qua: $'
    a db 0
    n db 0
.code
main proc
    ; Init DS
    mov ax,@data
    mov ds,ax
    ;
    mov ah,9
    lea dx,msg1
    int 21h
    ;
    mov ah,1
    int 21h
    cmp al,31h
    jb baoloi
    cmp al,39h
    ja baoloi
    ; Luu a
    sub al,30h
    mov a,al
    ;
    mov ah,9
    lea dx,msg2
    int 21h
    ;
    mov ah,1
    int 21h
    cmp al,31h
    jb baoloi
    cmp al,39h
    ja baoloi
    ; Luu n
    sub al,30h
    mov n,al
    ;
    jmp tinh_luythua
baoloi:
    mov ah,9
    lea dx,err
    int 21h
    jmp kt_chuongtrinh
```

```

    tinh_luythua:
    mov bl,a
    xor bh,bh
    mov cl,n
    xor ch,ch
    mov ax,1
    ;
    luythua:
    mul bx
    loop luythua
    push ax
    mov ah,9
    lea dx,res
    int 21h
    call xuat
    kt_chuongtrinh:
    call endmain
main endp

xuat proc
    pop bx ; Lay ra IP tu stack
    xor si,si ; dung lam flag
    ; Hien thi hang chuc nghin
    hang_chucnghin:
    pop ax ; Lay ra tham so tu stack
    xor dx,dx
    mov cx,10000
    div cx
    mov ah,2
    add al,30h
    push dx
    cmp al,30h
    jz hang_nghin
    mov si,1
    mov dl,al
    int 21h
    ; Hien thi hang nghin
    hang_nghin:
    pop ax
    xor dx,dx
    mov cx,1000
    div cx
    mov ah,2
    add al,30h
    push dx
    cmp si,1
    jz _nghin
    cmp al,30h
    jz hang_tram
    _nghin:
    mov si,1

```

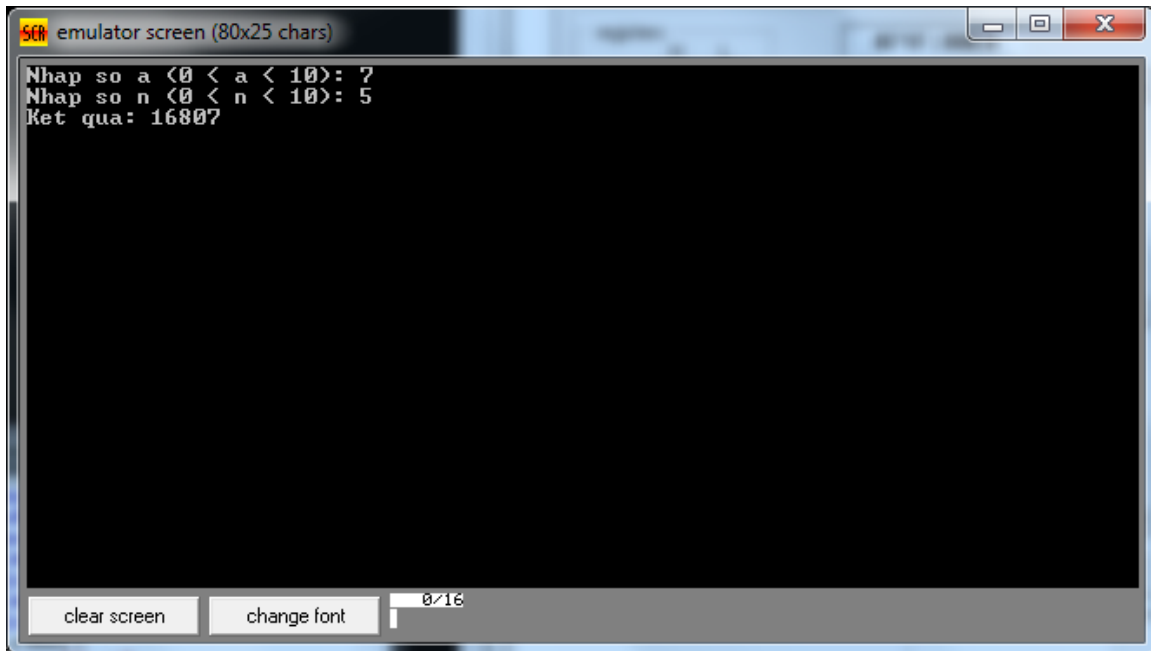
```

    mov dl,al
    int 21h
    ; Hien thi hang tram
hang_tram:
    pop ax
    xor dx,dx
    mov cx,100
    div cx
    mov ah,2
    add al,30h
    push dx
    cmp si,1
    jz _tram
    cmp al,30h
    jz hang_chuc
    _tram:
    mov si,1
    mov dl,al
    int 21h
    ; Hien thi hang chuc
hang_chuc:
    pop ax
    xor dx,dx
    mov cx,10
    div cx
    mov ah,2
    add al,30h
    push dx
    cmp si,1
    jz _chuc
    cmp al,30h
    jz hang_donvi
    _chuc:
    mov dl,al
    int 21h
    ; Hien thi hang don vi
hang_donvi:
    mov ah,2
    pop dx
    add dl,30h
    int 21h
    ;
    push bx
xuat endp

endmain proc
endmain endp
end code

```

Kết quả chương trình:



4. Giả sử $a=4$ thì n tối đa là bao nhiêu để hàm Xuất còn hiển thị đúng giá trị?

Trả lời: Ta thấy kết quả có thể được biểu diễn bởi tối đa 16 bits ứng với giá trị tối đa là $2^{16} - 1 = 65535$. Ta có $4^7 = 16384$ (thỏa mãn) mà $4^8 = 65536$ (vượt quá giới hạn biểu diễn 16 bits). Do đó, với $a = 4$ thì giá trị của n tối đa bằng 7.

Bài 6 (10 điểm)

Đặt 5 câu hỏi trắc nghiệm cho các nội dung đã học từ đầu kỳ đến tuần này. Mỗi câu hỏi có 4 phương án trả lời trong đó có 1 đáp án đúng. Nhập các câu hỏi vào form tại địa chỉ:

<https://docs.google.com/spreadsheet/viewform?formkey=dFh3MkZacWVTa0tNS0JWcm5QbF94bmc6MA>