CS 271Computer Architecture and Assembly Language

Self-Check for Lecture#5

Solutions are posted

1. Define a MASM constant for your name as a z-byte terminated string.

MY_NAME   EQU   <"Name O. Student",0>

2. What is the size of the string in the following MASM data segment declaration?

```
.data
stones      BYTE        "You Can't Always Get What You Want.",10,13,0
```

38 Bytes (Don't forget to count spaces, punctuation, and extra 8-bit numbers.)

3. Solve each problem using the following data segment:

```
.data
k      DWORD ?
n      DWORD ?
yes    BYTE        "Yes", 0
no     BYTE        "No", 0
maybe BYTE         "Maybe" ,0
```

NOTE: You cannot **cmp** memory to memory. At least one of the operands must be a register or a constant.

Assume that variables have been initialized. Write MASM code to implement the following high-level pseudo-code repetition structures.

3.1.
```
while (k < n){
     print (yes);
     k += 2;
}
```

3.2.
```
do{
     print (maybe);
     k++;
while (k < n);
```

3.3.
```
for (k = 10; k > 0; k--)
     print (k);
```

3.4.
```
for (k = 10; k <= n; k++)
     print (no);
```

1.
```
        mov    eax, k
again:
        cmp    eax, n
        jge    quit
        mov    edx, OFFSET yes
        call   WriteString
        add    eax, 2
        jmp    again
quit:
```

2.
```
        mov    eax, k
again:
        mov    edx, OFFSET maybe
        call   WriteString
        inc    eax
        cmp    eax, n
        jl     again
```

3.
```
        mov    ecx, 10
again:
        mov    eax, ecx
        call   WriteDec
        call   CrLf
        loop   again
```

4.
```
        mov    ecx, n
        sub    ecx, 10
        add    ecx, 1
again:
        mov    edx, OFFSET no
        call   WriteString
        loop   again
```