Lab Manual 03 – Nested Loops + Conditional Jumps

Activity 1[Nested Loop]: Write a program that computes double factorial of a number without using multiplication instruction.

Sample Run:

Num: 7
Example: 7!! = 7 × 5 × 3 × 1 = 105

Activity 2: Write a program that remove all the occurrences of a given number and compresses an ordered array (having multiple occurrences of one integer) . You are not allowed to use any extra array, just modify the input array.

Sample Run:

Num: 4
Arr: 2,2,2,3,4,4,5,5,5,6
After Compression, **Arr:** 2,3,5,6,0,0,0,0,0,0

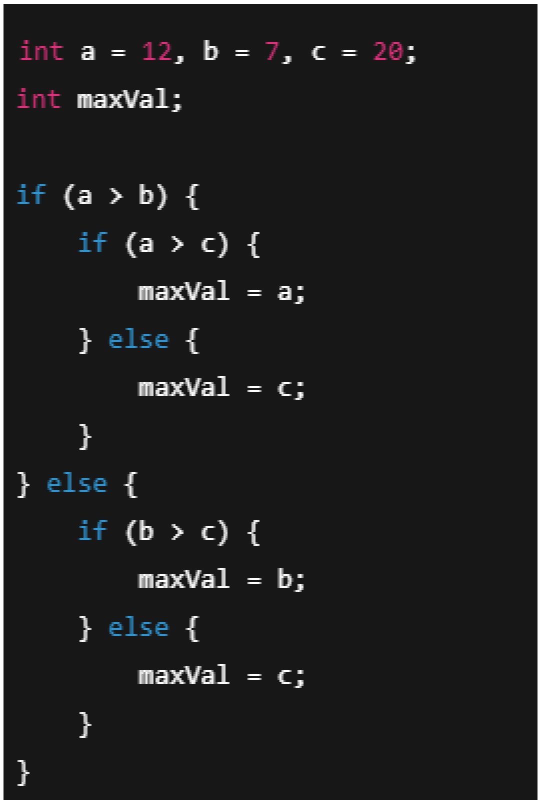
Activity 3: You did following Sorting Example in class.

i. Run this code (without any modification) on signed data and verify output.
ii. Modify this code to sort **signed data** and test it.

```
; sorting a list of ten numbers using bubble sort
; a program to add ten numbers without a separate counter
[org 0x0100]
    jmp start ; unconditionally jump over data
num1: dw 5,4,3,2,1

start:
mov si, 0

outerloop:
mov di, si
add di, 2

innerloop:
mov ax, [num1+si]
cmp ax, [num1+di]
jb noswap
mov dx, [num1+di]
mov [num1+si], dx
mov [num1+di], ax

noswap:
add di, 2
cmp di, 10
jb innerloop

add si,2
cmp si, 8
jb outerloop

mov ax, 0x4c00 ; terminate program
int 0x21
```

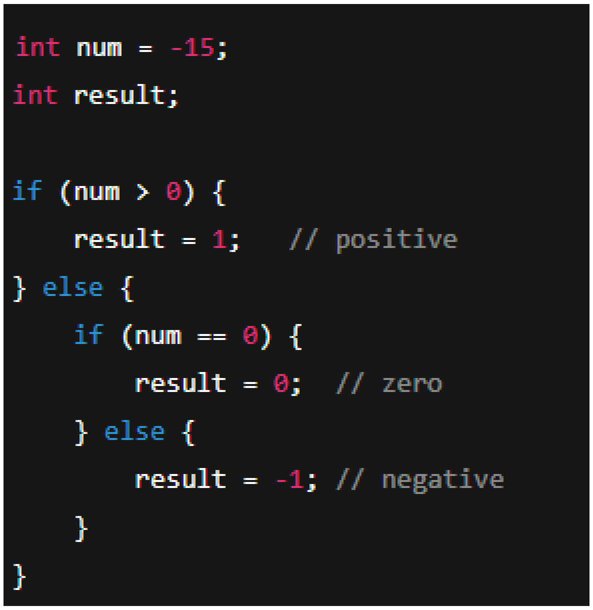
Activity 4: Convert the following C++ codes in assembly language.

A) Maximum Value



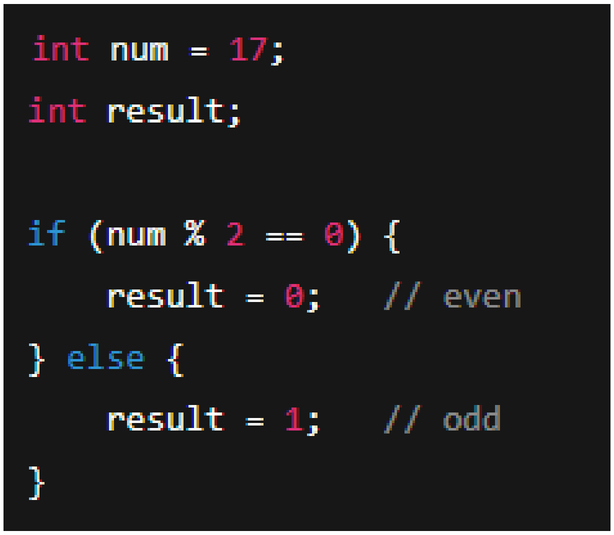
```
int a = 12, b = 7, c = 20;
int maxVal;

if (a > b) {
    if (a > c) {
        maxVal = a;
    } else {
        maxVal = c;
    }
} else {
    if (b > c) {
        maxVal = b;
    } else {
        maxVal = c;
    }
}
```

B) Signed Value



```
int num = -15;
int result;

if (num > 0) {
    result = 1;   // positive
} else {
    if (num == 0) {
        result = 0;  // zero
    } else {
        result = -1; // negative
    }
}
```

C) Even /Odd Number



```
int num = 17;
int result;

if (num % 2 == 0) {
    result = 0;   // even
} else {
    result = 1;   // odd
}
```