

Name: Bibek Dhungana

Lab 3 - In Lab Assignment

Due end of the lab session

Acknowledge your collaborators or source of solutions, if any. **Submission by the end of the LAB is required.** Please type your answers, handwritten submission will not be accepted. Do all of the following. A subset of your solutions will be graded.

1. Let us assume,
i=5
j=6
 - a. What will be the value of i and j when the following line is executed?
j = i++;
 - b. What will be the value of i and j when the following line was executed instead?
j=++i;

Part a:

In operator precedence, assignment operation is in high priority than post increment operator. So, first j is assigned to 5(value of i) then i value is increased by 1.

So, after code execution:

i = 6
j = 5

Part b:

The pre-increment operator is in high priority than assignment operator. So, first value of i is increased by 1 and then assigned to j.

So, after code execution:

i = 6
j = 6

2. What will be the output of the following piece of code?

```
int i = 1, j = 1;
for(--i && j++ ; i<10; i+=2)
{
    printf("loop ");
}
```

OUTPUT:

The string loop is printed 5 times because of the for loop. So, output is:
loop loop loop loop loop

3. Explain the output of the following program.

```
#include<stdio.h>
int main(){
    int a = 130;
    char *ptr;
    ptr = (char *)&a;
    printf("%d ",*ptr);
    return 0;
}
```

This means a is integer and assigned with value of 130. This is stored in memory in binary form using 4 bytes (or 2 bytes in some architecture) of memory. We use &a to find the base address of a.

Then, this line ptr = (char *) &a; force compiler to believe ptr is pointing to address of a char.

When we try to dereference ptr using *ptr, it believes ptr is char pointer and only read what ever in that first bytes in the memory and print the result.

So, 130 in binary is 00000000 00000000 00000000 10000010 in 4 bytes architecture. But char only point to 10000010 which is 128. But, signed char can only store value from -128 to 127.

So, the value overflows and -126 is obtained.

If we use unsigned char, we can get 128 as result

In simple word, this is force conversion of int data to char data using pointer. Since signed char only have one byte memory (-128 to 127). It can not hold all the information of int byte and information is lost.

So, output is -126.

