Computer Science & Engineering Department
I. I. T. Kharagpur

Compilers Laboratory: CS39003

3rd Year CSE: 5th Semester

Assignment - 1: Annotating Assembly Marks: *10*
Assign Date: *20^{nd} July, 2016* Submit Date: *23:55, 26^{th} July, 2016*

1. Translate the following C program using GCC/Linux to the assembly language program of x86-64 (Intel 64-bit processor).

   cc -Wall -S <*file name*>.c

   Do not give any optimization option. The *file name* should be ass1_*roll*.c where *roll* is your roll number.

   Write comments in the assembly language code corresponding to the program <*file name*>.s. Comments should explain the corresponding assembly language instructions and also should clearly show the connection between the C program and the assembly language program.

```
/*
 * ass1.c Generate assembly code of x86-64 and comment
 */

#include <stdio.h>
#define MAXSIZE 100

void inst_sort(int num[],int n);
int bsearch(int num[],int n,int item);
int insert(int num[],int n,int item);

int main()
{
        int n, a[MAXSIZE], item, i, loc;

        printf("Enter how many elements you want:\n");
        scanf("%d", &n);

        printf("Enter the %d elements:\n", n);
        for(i = 0; i < n; i++) scanf("%d", &a[i]);

        inst_sort(a,n);

        printf("\nEnter the item to  search\n");
        scanf("%d", &item);

        loc=bsearch(a,n,item);

        if (item == a[loc]) {
                printf("\n%d found in position: %d\n", item, loc + 1);
        } else {
                loc=insert(a,n,item);
                n++;
                printf("\n%d inserted in position: %d\n", item, loc + 1);
        }

        printf("The list of %d elements:\n", n);
        for(i = 0; i < n; i++) printf("%6d", a[i]);
        printf("\n");
        return 0;
}

void inst_sort(int num[],int n)
{
        int i,j,k;
        for(j=1;j<n;j++) {
                k=num[j];
                for(i=j-1;i>=0 && k<num[i];i--) num[i+1]=num[i];
                num[i+1]=k;
        }
}


int bsearch(int a[],int n,int item)
{
        int mid, top, bottom;

        bottom = 1;
        top = n;

        do {
                mid = (bottom + top) / 2;
                if (item < a[mid])
                        top = mid - 1;
                else if (item > a[mid])
                        bottom = mid + 1;
        } while (item != a[mid] && bottom <= top);

        return mid;
}

int insert(int num[],int n,int k)
{
        int i;

        for(i=n-1;i>=0 && k<num[i];i--) num[i+1]=num[i];
        num[i+1]=k;

        return (i+1);
}
```

2. The commented assembly language program should remain syntactically correct.
3. Intel assembly language manual and other reading materials are available in Moodle.