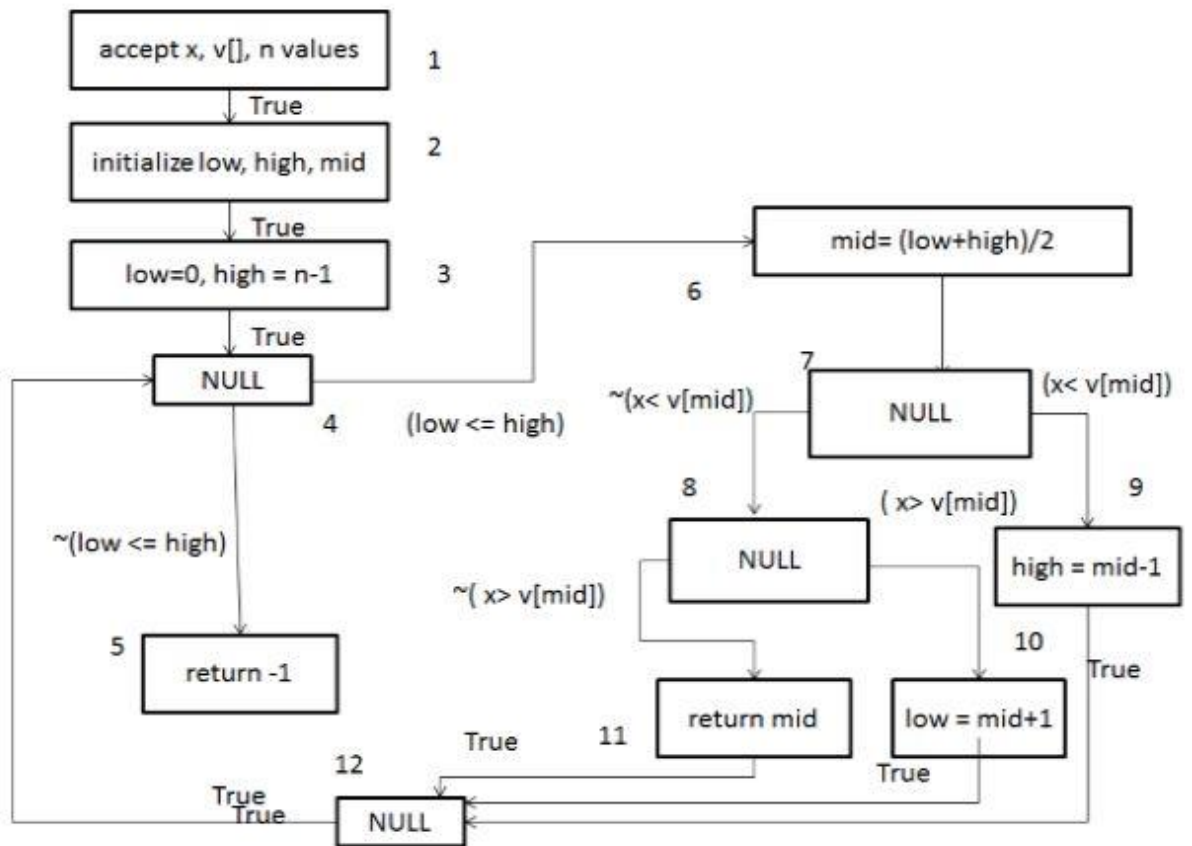


```

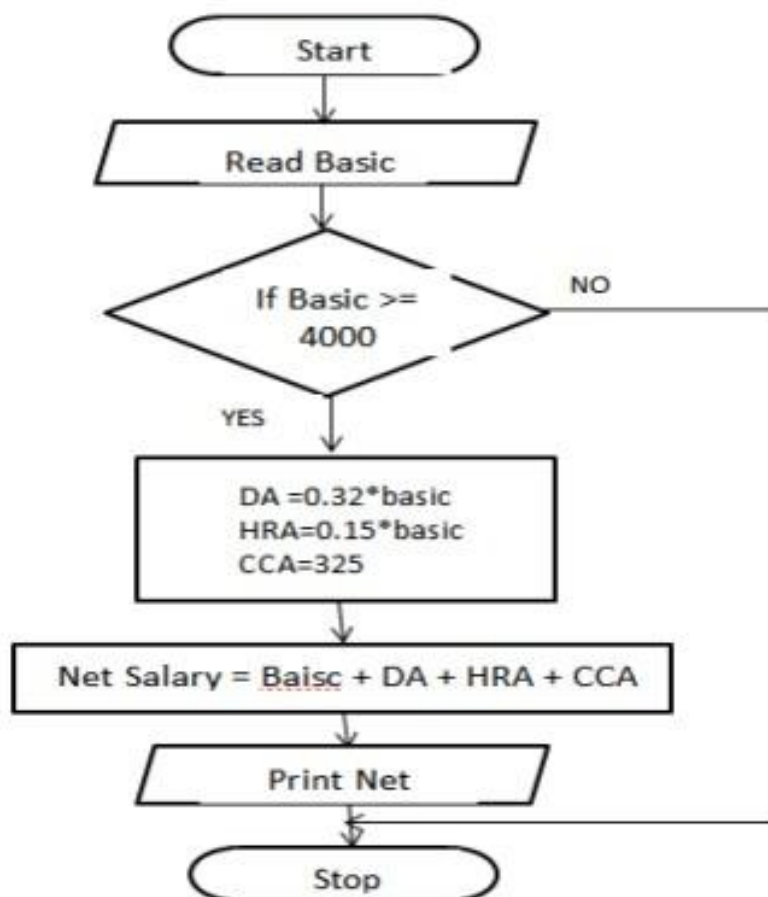
//BinarySearch
int binarysearch (int x, int V[ ], int n) {
int low, high, mid ;
low= 0;
high = n-1 ;
while (low<= high)
{ mid = ( low + high) / 2 ;
if( x < V[mid] )
high = mid - 1;
else if (x > V[mid])
low = mid +1 ;
else
return mid ;
}
return -1 ;
}

```



```
//net salary of an employee.
```

```
#include <stdio.h>
int main() {
    // Declare variables
    float basic, DA, HRA, CCA, netSalary;
    printf("Enter the basic salary: ");
    scanf("%f", &basic);
    if (basic >= 4000) {
        // Step 3.1: Calculate DA
        DA = 0.32 * basic;
        HRA = 0.15 * basic;
        CCA = 325;
        netSalary = basic + DA + HRA + CCA;
        printf("Net Salary: %.2f\n", netSalary);
    } else {
        printf("Basic salary should be greater than or equal to
4000. \n");
    }
    return 0;
}
```



```

//Selection sort

#include<stdio.h>
#include<conio.h>
#define MAX 100

void main()
{
    int data[MAX];
    int i, j, temp, size, sort;

    clrscr();

    printf("Enter the size of the Array. n");
    scanf("%d", &size);

    printf("Size of the Array : %d n", size);
    printf("Enter elements of Array: n");
    for(i=0; i<size; i++)
        scanf("%d", &data[i]);

    printf("Array is as follows: n");
    for(i=0; i<size; i++)
        printf("%d t", data[i]);

    printf("n 1. Sort in Ascending n 2. sort in Descending n");
    printf("Enter your choice: ");
    scanf("%d", &sort);
    if(sort==1)
    {
        for(i=0; i<size; i++)
        {
            for(j=i+1; j<size; j++)
            {
                if(data[i]>data[j])
                {
                    temp=data[i];
                    data[i]=data[j];
                    data[j]=temp;
                }
            }
        }
    }
    if(sort==2)
    {
        for(i=0; i<size; i++)
        {
            for(j=i+1; j<size; j++)
            {

```

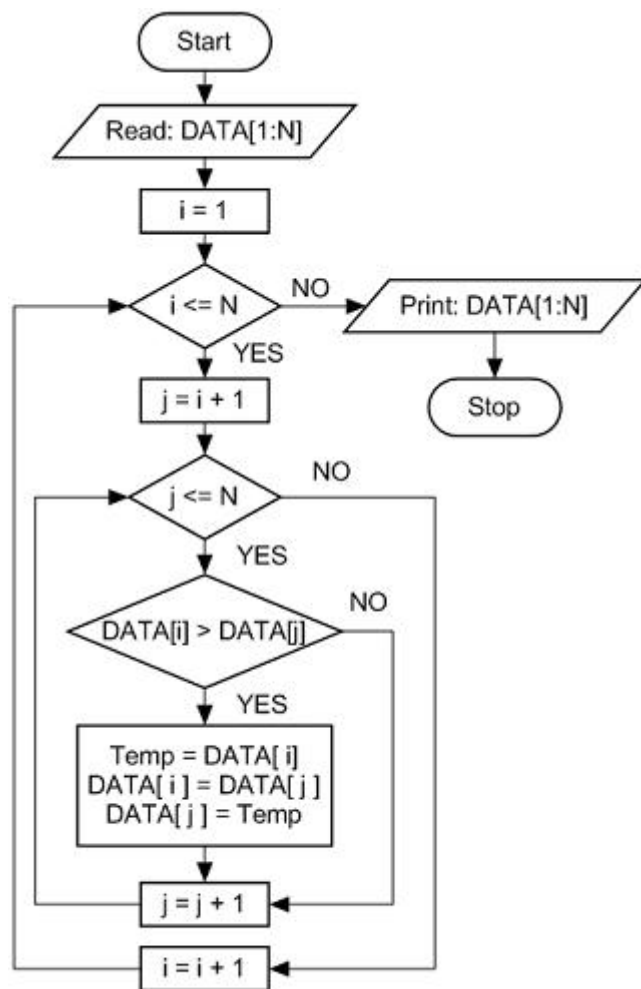
```

        if(data[i]<data[j])
        {
            temp=data[i];
            data[i]=data[j];
            data[j]=temp;
        }
    }

}

printf("\n");
printf("After Sorting Array is as follows: n");
for(i=0; i<size; i++)
    printf(" %d t",data[i]);
getch();
}

```



Selection Sort Flowchart

```

//Bubble sort
#include<stdio.h>
#include<conio.h>
#define MAX 100
void main()
{
    int data[MAX];
    int i, j, temp, size, choice;
    clrscr();
    printf("Enter size of Array: ");
    scanf("%d", &size);
    printf("Enter elements of Array. n");
    // Taking Input for Array.
    for(i=0; i<size; i++)
        scanf("%d", &data[i]);
    printf("Array is as follows: n");
    // Print the Input Array.
    for(i=0; i<size; i++)
        printf("%d t", data[i]);
    printf("n 1. Sort in Ascending n 2. Sort in Descending n");
    printf("Enter your choice: ");
    scanf("%d", &choice);
    // Sorting Array in Ascending order.
    if(choice==1)
    {
        for(i=0; i<size; i++)
        {
            for(j=0; j<size-1; j++)
            {
                if(data[j]>data[j+1])
                {
                    temp=data[j];
                    data[j]=data[j+1];
                    data[j+1]=temp;
                }
            }
        }
    }
    // Sorting Array in Descending order.
    if(choice==2)
    {
        for(i=0; i<size; i++)

```

```

    {
        for(j=0; j<size-1; j++)
        {
            if(data[j]<data[j+1])
            {
                temp=data[j];
                data[j]=data[j+1];
                data[j+1]=temp;
            }
        }
    }

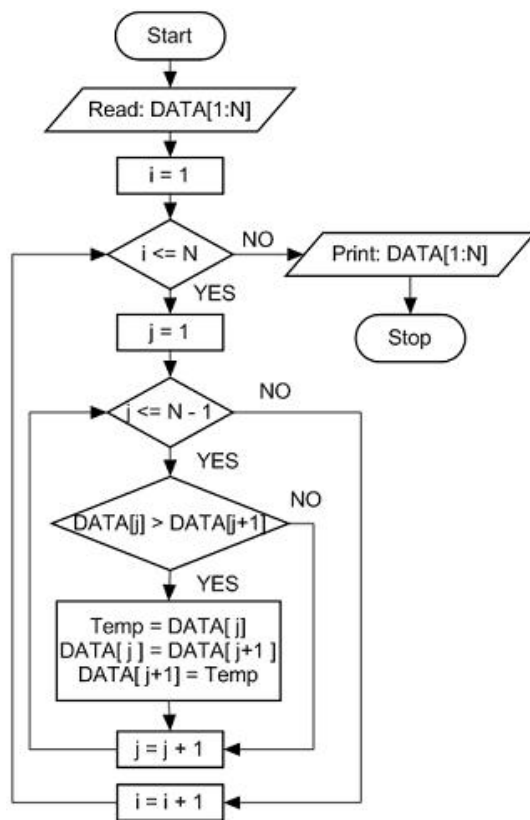
}

// Printing the Output Array.

printf("After Bubble Sort Array is as follows: n");
for(i=0; i<size; i++)
    printf("%d t", data[i]);

getch();
}

```



Bubble Sort Flowchart

```

//Insertion sort
#include <stdio.h>

int main()
{
    int a[6];

    int key;

    int i, j;

    int temp;

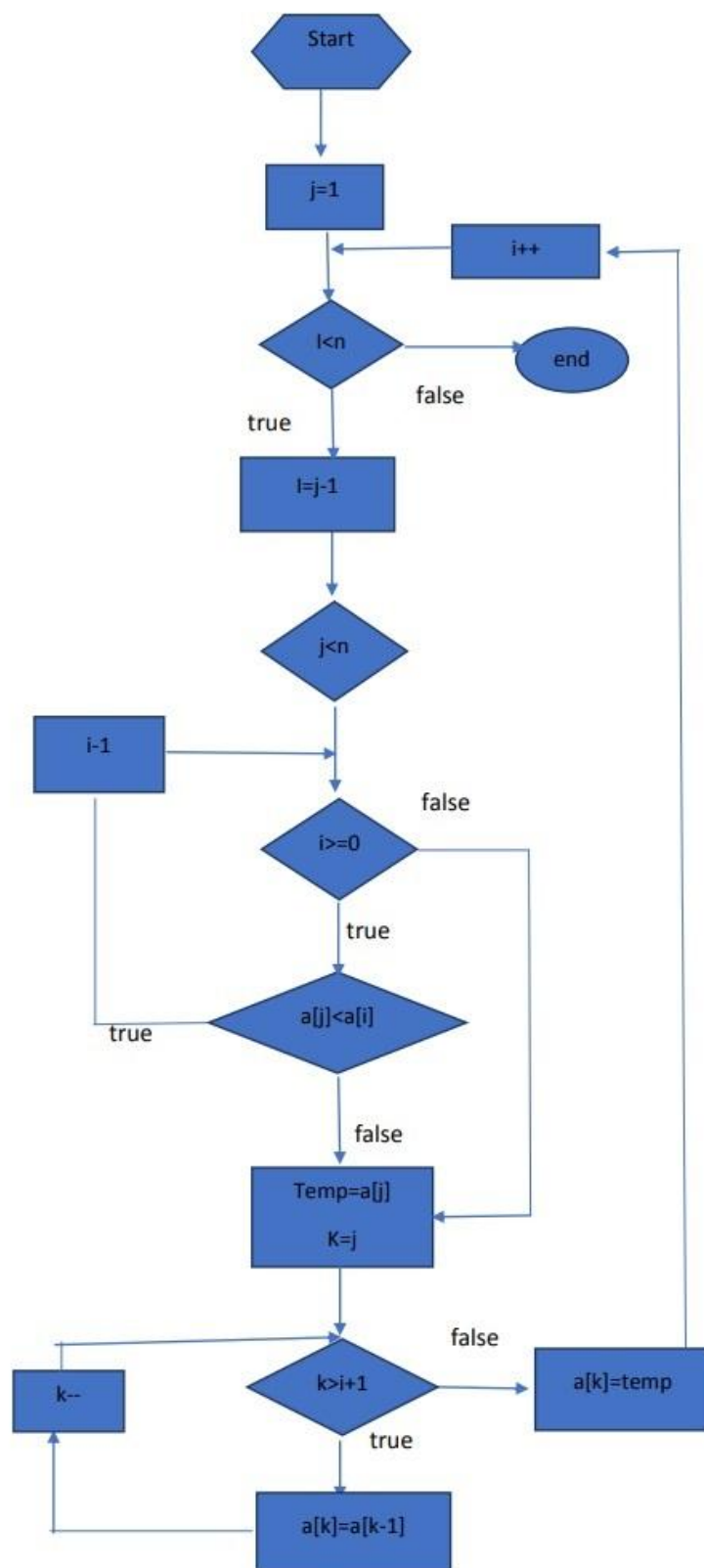

    printf("Enter any six elements to be sorted using insertion sort\n");
    for (i = 0; i < 6; i++) {
        scanf("%d", &a[i]);
    }


    for (j = 1; j < 6; j++) {
        key = a[j];
        i = j - 1;
        while ((i >= 0) && (a[i] >= key)) {
            temp = a[i + 1];
            a[i + 1] = a[i];
            a[i] = temp;
            i = i - 1;
        }
        a[i + 1] = key;
    }


    printf("elements after sorting using insertion sort are \n");
    for (i = 0; i < 6; i++) {
        printf("%d  \n", a[i]);
    }


    return 0;
}

```




```
// C program to find largest of two numbers
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int num1, num2, largest;
```

```
    /*Input two numbers*/
```

```
    printf("Enter two numbers:\n");
```

```
    scanf("%d%d", &num1, &num2);
```

```
    /*check if a is greater than b*/
```

```
    if (num1 > num2)
```

```
        largest = num1;
```

```
    else
```

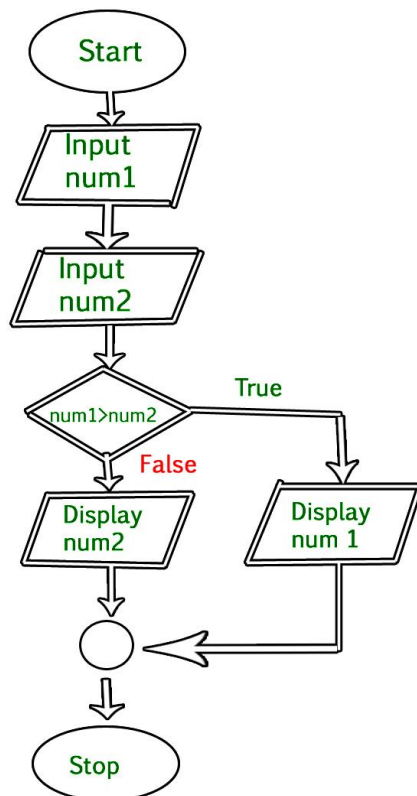
```
        largest = num2;
```

```
    /*Print the largest number*/
```

```
    printf("%d", largest);
```

```
    return 0;
```

```
}
```



```
//Find Maximum of 3 Numbers
#include <stdio.h>

int main() {
    // Declare variables
    int N1, N2, N3, Max;

    // Input: Get three numbers from the user
    printf("Enter three numbers: ");
    scanf("%d %d %d", &N1, &N2, &N3);

    // Find Maximum Number
    Max = N1; // Initialize Max to N1

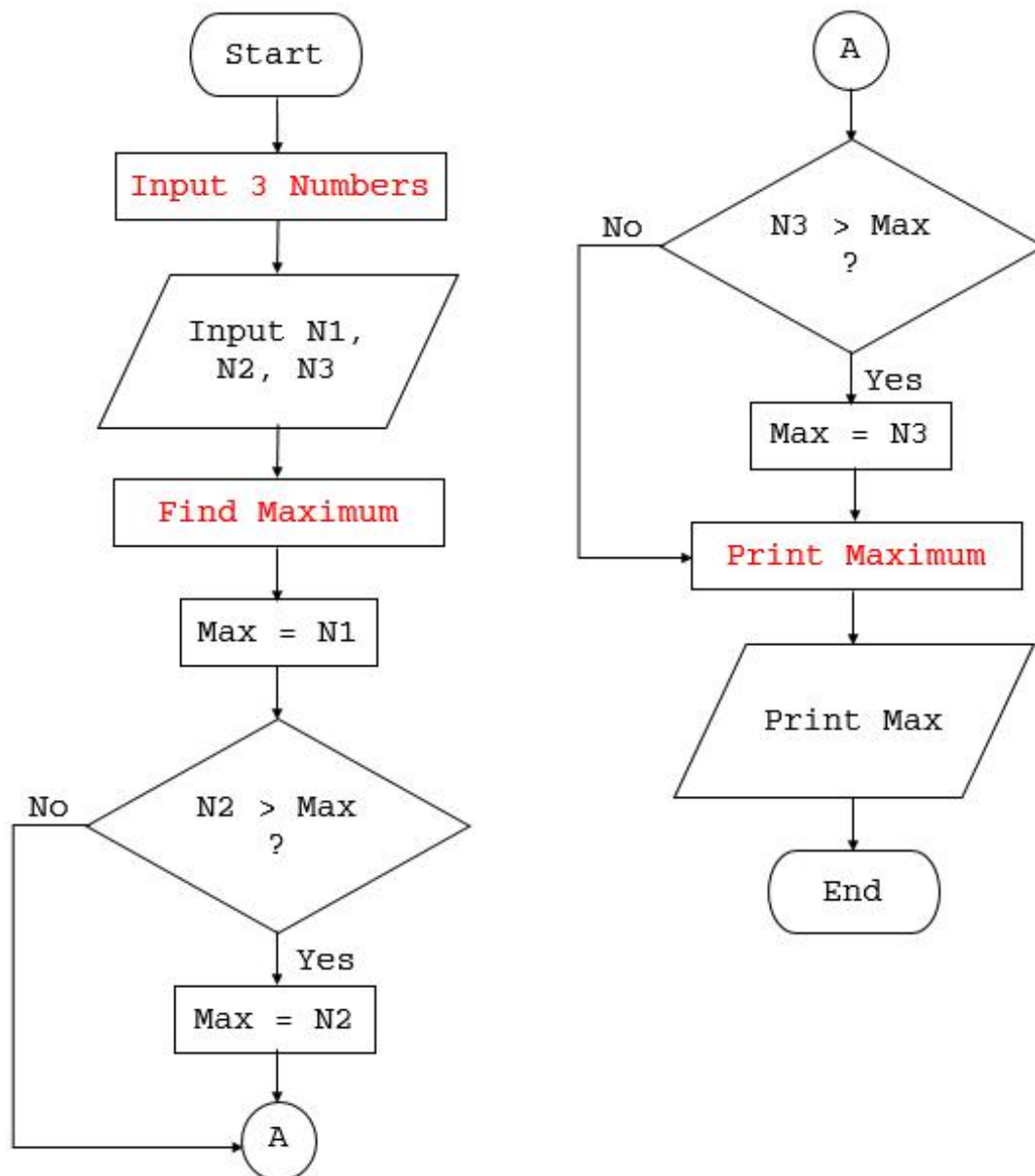
    // Check if N2 is greater than Max
    if (N2 > Max) {
        Max = N2;
    }

    // Check if N3 is greater than Max
    if (N3 > Max) {
        Max = N3;
    }

    // Print Maximum Number
    printf("Maximum number is: %d\n", Max);

    return 0; // Indicates successful execution
}
```

FIND MAX OF 3 NUMBERS



```

//Volume of Cube
#include <stdio.h>
int main() {
    // Declare variables
    float E, Volume;
    // Input: Get the edge length of the cube from the user
    printf("Enter the edge length of the cube: ");
    scanf("%f", &E);
    // Calculate Cube Volume
    Volume = E * E * E;
    // Print Cube Volume
    printf("Volume of the cube with edge length %.2f is: %.2f\n", E,
Volume);

    return 0; // Indicates successful execution
}

```

