1.

```c
#include <stdio.h>
void insertionSort(int arr[] , int n){
    int i , key ,j;
    for(i=1;i<n;i++){
        key = arr[i];
        j=i-1;
        while(j>=0 && arr[j]>=key){
            arr[j+1] = arr[j];
            j--;
        }
        arr[j+1] = key;
    }
}

void printEle(int arr[] , int n){
    int i;
    for(int i=0;i<n;i++){
        printf("%d\t",arr[i]);
    }
    printf("\n");
}
int main(){
    int arr[] = {3, 5, 7, 2, 8, 9, 4};
    int n=7,i;
    printf("BEFORE SOTRING \n");
    printEle(arr,n);
    insertionSort(arr,n);
    printf("AFTER SORTING\n");
    printEle(arr,n);
    return 0;
}
```

2.

```c
#include<stdio.h>

int stack[100], top = -1 , size;

void push(int x){
    if(top == size-1)
    {
        printf("OVERFLOW\n");
        return;
    }
    stack[++top] = x;
}
int maxElem(){
```

```c
    int maxx = stack[0];
    for(int i=1;i<=top;i++){
        if(maxx < stack[i])
            maxx = stack[i];
    }
    return maxx;
}
int main(){

    printf("NO OF VALUES NEED\n");
    scanf("%d", &size);
    int x;
    for (int i = 0; i < size; i++)
    {
        printf("Enter value at position %d :\t",(i+1));
        scanf("%d", &x);
        push(x);
    }

    printf("Maximum elemnt %d\n",maxElem());

    return 0;


}
```

3.

```c
#include<stdio.h>

int stack[100], top = -1 , size;

void push(int x){
    if(top == size-1)
    {
        printf("OVERFLOW\n");
        return;
    }
    stack[++top] = x;
}
int minElem(){
    int minn = stack[0];
    for(int i=1;i<=top;i++){
        if(minn > stack[i])
            minn = stack[i];
    }
    return minn;
}
```

```c
int main(){

    printf("NO OF VALUES NEED\n");
    scanf("%d", &size);
    int x;
    for (int i = 0; i < size; i++)
    {
        printf("Enter value at position %d :\t",(i+1));
        scanf("%d", &x);
        push(x);
    }

    printf("Minimum elemnt %d\n",minElem());

    return 0;


}
```