```c
#include<stdio.h>
// Bubble sorting
int main() {
    int arr[5],i,j,c;
    printf("__Bubble sorting implementation__\n");
    printf("Enter the elements of the array:\n");
    for(c=0;c<5;c++) {
        scanf("%d",&arr[c]);
    }

    for(i=0;i<4;i++) {
        for(j=0;j<4-i;j++) {
            if(arr[j]>arr[j+1]) {
                int t=arr[j];
                arr[j]=arr[j+1];
                arr[j+1]=t;
            }
        }
    }
    printf("Sorted array in ascending order :\n");
    for(c=0;c<5;c++) {
        printf("%d ",arr[c]);
    }
    return 0;
}
```

```
// stack implementation using array

#include<stdio.h>
#define N 5
int stack_array[N];
int top=-1;

int main() {
    int choice,data;
    printf("___Stack operation using array___\n");

    while(1) {
    printf("1.Push\n 2.Pop\n 3.Peek\n 4.Display\n 5.Exit\n");
    printf("Enter your choice:\n");
    scanf("%d",&choice);

    switch (choice)
    {
    case 1:
         printf("enter the element to be pushed:");
         scanf("%d",&data);
         push(data);
        break;
    case 2:
        pop();
        break;
    case 3:
        printf("Top of the element is: %d\n",peek());
        break;
    case 4:
        display();
        break;
    case 5:
        printf("Exiting...\n");
        exit(0);
        break;

    default:
        printf("Enter a valid choice!\n");
        break;
    }
    }
    return 0;
}

void push(int data) {
    if(top==N-1) {
        printf("Overflow!");
        return ;
    }
    else {
        top=top+1;
        stack_array[top]=data;
        printf("Element: %d is inserted\n",data);
    }
}

void pop() {
    if(top==-1) {
```

```
        printf("Underflow!");
    } else {
        top=top-1;
    }
}

int peek() {
    if(top==-1){
        printf("Underflow!");
        return 0;
}
else {
    return stack_array[top];
}
}

void display() {
    if(top==-1){
        printf("Stack is empty!");
        return ;
} else {
    for(int i=top;i>=0;i--) {
        printf("Elements are: %d",stack_array[i]);
    }
    printf("\n");
}
}
```

```
#include<stdio.h>
// Binary search implementation
int binary_search(int arr[],int n,int x){
    int l=0,r=n-1,mid;
    while(l<=r) {

        mid=(l+r)/2;
        // 1 cond^n:
        if(x==arr[mid]) {
           return mid;
        }
        // 2 cond^n
        else if(x<arr[mid]) {
            r=mid-1;
        }
        // 3 cond^n
        else {
            l=mid+1;
        }
    }
    return -1;

}

int main() {
    int arr[]={10,15,20,25,30,35,40,45};
    int n=8;
    int x;
    printf("Enter the element to be searched:");
    scanf("%d",&x);
    int result= binary_search(arr,8,x);
    if(result==-1) {
        printf("Data not found!\n");
    }
    else{
        printf("Data found at : %d index\n",result);
        printf("And Data= %d",x);
    }
         return 0;
     }
```

```
 // arrays practice questions.

#include<stdio.h>
int main() {
int arr[]={1,2,3,4,5};
printf("%d\n",*(arr+2));
printf("%d\n",(arr+5));
return 0;
}
```

```c
#include<stdio.h>
int main() {
    float radius;
    printf("enter radius :");
    scanf("%f", &radius);

    printf("area is : %f",3.14*radius*radius);
    return 0;
}
```

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

int main() {
    int x,y,z;
    printf("Enter first number :");
    scanf("%d", &x);
    printf("Enter second number :");
    scanf("%d", &y);
    printf("Enter third number :");
    scanf("%d", &z);

    int Average;
    printf("average is : %d \n",x+y+z/3);
    return 0;
}
```

```
#include<stdio.h>
int main() {
    printf("Hello world");
    return 0;
}
```