Q1. WAP to increase every student mark by 5 & then print the updated array

#include<stdio.h>

int main(){

int i,marks[10];

for(i=0;i<10;i++)

{

printf("provide marks of student %d\n",i+1);

scanf("%d",&marks[i]);

}

printf("marks before : ");

for(i=0;i<10;i++)

{

printf("%d ",marks[i]);

}

for(i=0;i<10;i++)

{ marks[i]=marks[i]+5;}

printf("\nmarks after increased by 5 : ");

for(i=0;i<10;i++){

printf("%d ",marks[i]);}

return 0;

}

Q2. WAP to print grade of students as per their marks given in an array. (>=75-- A  
grade, 74 to 60--B Grade, 59 to 40--C grade below 40--D grade)

#include<stdio.h>

int main(){

int i,marks[10];

for(i=0;i<10;i++)

{

printf("enter marks of student %d: ",i+1);

scanf("%d",&marks[i]);

}

for(i=0;i<10;i++)

{

printf("%d ",marks[i]);

}

for(i=0;i<10;i++)

{

if(marks[i]>=75)

{printf("\ngrade A");}

else if(marks[i]<75&&marks[i]>=60)

{printf("\ngrade B");}

else if(marks[i]<59&&marks[i]>=40)

{printf("\ngrade C");}

else if(marks[i]<40)

{printf("\ngrade D");}

}

return 0;

}

Q3. WAP to find who scored first “99” in an array marks

#include<stdio.h>

int main(){

int i,marks[10];

for(i=0;i<10;i++)

{

printf("enter marks of student %d: ",i+1);

scanf("%d",&marks[i]);

}

for(i=0;i<10;i++)

{

printf("%d ",marks[i]);

}

for(i=0;i<10;i++)

{

if(marks[i]==99)

{ printf("\nstudent %d scored first 99 marks ",1+i);

break;}

}

return 0;

}

Q4. WAP to find Who & how many students have scored 99 in an array Marks

#include<stdio.h>

int main(){

int a=0,i,marks[10];

for(i=0;i<10;i++)

{

printf("enter marks of student %d: ",i+1);

scanf("%d",&marks[i]);

}

for(i=0;i<10;i++)

{

printf("%d ",marks[i]);

}

for(i=0;i<10;i++)

{

if(marks[i]==99)

{ a++;

printf("\nstudent %d scored 99 marks ",1+i);

}

}

printf("\n%d students scored 99",a);

return 0;

}

Q5. WAP to find sum of all scores in Marks array.

#include<stdio.h>

int main(){

int sum=0,i,marks[10];

for(i=0;i<10;i++)

{

printf("enter marks of student %d: ",i+1);

scanf("%d",&marks[i]);

}

for(i=0;i<10;i++)

{

printf("%d ",marks[i]);

}

for(i=0;i<10;i++)

{

sum=sum+marks[i];

}

printf("\n%d",sum);

return 0;

}

Q6. WAP to find average score of the Marks array

#include<stdio.h>

int main(){

int sum=0,i,marks[10];

int n,average;

printf("how many indexes you want");

scanf("%d",&n);

for(i=0;i<n;i++)

{

printf("enter marks of student %d: ",i+1);

scanf("%d",&marks[i]);

}

for(i=0;i<n;i++)

{

printf("%d ",marks[i]);

}

for(i=0;i<n;i++)

{

sum=sum+marks[i];

}

average=sum/10;

printf("\n%d",average);

return 0;

}

Q7. WAP to check whether score is even or odd in an array.

#include<stdio.h>

int main(){

int sum=0,i,marks[10];

int n,average;

printf("how many indexes you want");

scanf("%d",&n);

for(i=0;i<n;i++)

{

printf("enter marks of student %d: ",i+1);

scanf("%d",&marks[i]);

}

for(i=0;i<n;i++)

{

printf("%d ",marks[i]);

}

for(i=0;i<n;i++)

{

if(marks[i]%2==0)

{

printf("\neven score");

}

else if(marks[i]%2!=0)

{

printf("\nodd score");

}

}

return 0;

}

Q8)   
WAP to find maximum & minimum score in the Marks array.

#include<stdio.h>

int main(){

int i,max,min,marks[5];

for(i=0;i<5;i++)

{

printf("enter marks of student %d\n",i+1);

scanf("%d",&marks[i]);

}

printf("the marks of students are:\n");

for(i=0;i<5;i++)

{

printf("%d ",marks[i]);

}

max=marks[0];

for(i=0;i<5;i++)

{

if(max<marks[i])

max=marks[i];

}

printf("\nmax score is: %d",max);

min=marks[0];

for(i=0;i<5;i++)

{

if(min>marks[i])

min=marks[i];

}

printf("\nmin score is: %d",min);

return 0;

}

Q9. WAP to find a peak element which is not smaller than its neighbours.

#include<stdio.h>

int main(){

int i,arr[5],a;

for(i=0;i<5;i++)

{

printf("enter element %d:\n",i+1);

scanf("%d",&arr[i]);

}

for(i=0;i<5;i++)

{

printf("%d ",arr[i]);

}

for(i=0;i<5;i++)

{

if(arr[i-1]<arr[i]&&arr[i+1]<arr[i])

a=arr[i];

}

printf("\npeak element is %d",a);

return 0;

}

Q10. WAP to count prime numbers in an array.

#include<stdio.h>

int main(){

int i,j,a[5],f,n;

int prime=0;

for(i=0;i<5;i++)

{

printf("enter %d element:",i+1);

scanf("%d",&a[i]);

}

for(i=0;i<5;i++){

n=a[i];

f=0;

if(n<=1){

f=1;}

else{

for(j=2;j\*j<=n;j++){

if(n%j==0){

f=1;

break;

}

}

}

if(f==0){

printf("%d is prime\n",n);

prime++;

}

}

printf("total prime numbers:%d\n",prime);

return 0;

}

Q11. WAP to implement Insert -Front, any position in between & end in an array. Print  
the array before insert & after insert

#include<stdio.h>

int main(){

int i,size=7,a[7],pos,item;

for(i=0;i<7;i++)

{

printf("enter element %d:",i+1);

scanf("%d",&a[i]);

}

printf("array before insertion:");

for(i=0;i<7;i++)

{

printf("%d ",a[i]);

}

printf("\nenter the element you want to insert:");

scanf("%d",&item);

printf("\nenter the position where you want to insert:");

scanf("%d",&pos);

size++;

for(i=size-1;i>=pos;i--)

{

a[i]=a[i-1];

}

a[pos-1]=item;

printf("\narray after insertion:");

for(i=0;i<size;i++)

{

printf("%d ",a[i]);

}

return 0;

}

Q12. WAP to implement delete-Front, any position in between & end in an array. Print  
the array before delete & after delete

#include<stdio.h>

int main(){

int i,a[7],pos;

for(i=0;i<7;i++)

{

printf("enter element %d:",i+1);

scanf("%d",&a[i]);

}

printf("array before deletion:");

for(i=0;i<7;i++)

{

printf("%d ",a[i]);

}

printf("\nenter the position where you want to delete:");

scanf("%d",&pos);

for(i=pos-1;i<7;i++)

{

a[i]=a[i+1];

}

printf("\narray after deletion:");

for(i=0;i<6;i++)

{

printf("%d ",a[i]);

}

return 0;

}

Q13. Given an array, the task is to cyclically rotate the array clockwise by one time.  
Examples:  
Input: arr[] = {1, 2, 3, 4, 5}  
Output: arr[] = {5, 1, 2, 3, 4}  
Input: arr[] = {2, 3, 4, 5, 1}  
Output: {1, 2, 3, 4, 5}

#include<stdio.h>

int main(){

int n,i,j,element;

printf("enter size of array");

scanf("%d",&n);

int arr[n];

for(i=0;i<n;i++)

{

printf("enter element %d:",i+1);

scanf("%d",&arr[i]);

}

printf("array:");

for(i=0;i<n;i++)

{

printf("%d ",arr[i]);

}

i=n-1;

j=n-2;

while(i>0){

int temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

i--;

j--;

}

printf("\narray after rotation:");

for(i=0;i<n;i++)

{

printf("%d ",arr[i]);

}

return 0;

}

Q14. Given an array of n integers. The task is to print the duplicates in the given array.  
If there are no duplicates then print -1.  
Examples:  
Input: {2, 10,10, 100, 2, 10, 11,2,11,2}  
Output: 2 10 11  
Input: {5, 40, 1, 40, 100000, 1, 5, 1}  
Output: 5 40 1

#include <stdio.h>

int main() {

int i,j,a[6],flag=1;

printf("Enter 6 elements of array\n");

for (i=0;i<6;i++)

{scanf("%d", &a[i]); }

printf("Array: ");

for (i=0;i<6;i++)

{printf("%d ", a[i]); }

printf("\n");

for (i=0;i<6;i++)

{for (j=i+1;j<6;j++)

{if (a[i] == a[j])

{printf("%d is a duplicate\n", a[i]);

flag=0;

break;}}}

if(flag==1)

{printf("-1");}

return 0;

}