**Que.1) WAP to set all the bits which are having position multiple of 7 from 9th position for given**

**integer using bitwise operator.**

**EX: i/p: 156**

**00000000000000000000000010011100**

**o/p: 00010000001000000100000010011100**

/\*#include<stdio.h>

main()

{

int num,i,j;

printf("enter your number...\n");

scanf("%d",&num);

for(i=31;i>=0;i--)

printf("%d",num>>i&1);

printf("\n");

for(i=1,j=1;i<=31;i++,j++)

{

i=7+7\*j;

num=num|(1<<i);

}

printf("print after..\n");

for(i=31;i>=0;i--)

printf("%d",num>>i&1);

}\*/

**2) WAP to print following pattern.**

**1**

**1 2 1**

**1 2 3 2 1**

**1 2 3 4 3 2 1**

**// using loop…………………**

/\*#include<stdio.h>

int fact(int);

int combi(int ,int);

main()

{

int i,j,r,k,t;

printf("enter your rows...\n");

scanf("%d",&r);

for(i=1;i<=r;i++)

{

k=1;

t=0;

for(j=1;j<=r\*2-1;j++)

{

if(j>=r+1-i && j<=r-1+i && k)

{

printf("%d ",combi(i-1,t));

k=0;

t++;

}

else

{

printf(" ");

k=1;

}

}

printf("\n");

}

}

int fact(int n) // factorial

{

int f=1;

while(n>=1)

{

f=f\*n;

n--;

}

return f;

}

int combi(int n,int r) // calculate nCr here

{

//return (fact(n)/(fact(n-r)\*fact(r)));

return (fact(n)/fact(n-r)/fact(r));

}\*/

**//Using goto………………**

\*#include<stdio.h>

int fact(int );

int combi(int,int);

main()

{

int i,j,r,c,k;

printf("enter your rows..\n");

scanf("%d",&r);

i=0;

L1: i++;

if(i<=r)

{

j=0;

L2:j++;

if(j<=2\*r-1)

{

k=1;

if(j>=r+1-i && j<=r-1+i && k)

{

printf("\* ");

k=0;

}

else

{

printf(" ");

k=1;

}

goto L2;

}

printf("\n");

goto L1;

}

}

int fact(int n)

{

int f=1;

while(n>=1)

{

f=f\*n;

n--;

}

return f;

}

int combi(int n,int r)

{

return (fact(n)/fact(n-r)/fact(r));

}

**3. WAP to check whether a given number is a perfect cube or not.**

**EX: i/p: 125 The number is a perfect Cube of 5**

/\*#include<stdio.h>

int checkcube(int);

main()

{

int num,i;

printf("enter your number...\n");

scanf("%d",&num);

i=checkcube(num); // cube function call

if(i==0)

printf("The number has NOT perfect Cube");

else

printf("%d The number is a perfect Cube of %d",num,i);

}

int checkcube(int num) // cube function definition

{

int i;

for(i=2;i<num;i++)

{

int n=0;

if(num%i==0)

{

n=i\*i\*i;

if(num==n)

{

return i;

break;

}

}

}

return 0;

}\*/

**5) WAP to find the missing number from a given array. There are no duplicates in list.**

**EX: The given array is : 1 3 4 2 5 6 9 8**

**The missing number is : 7**

\*#include<stdio.h>

main()

{

int a[10],ele,i,j,min,k;

ele=sizeof(a)/sizeof(a[0]);

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

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

printf("\n");

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

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

for(i=0;i<ele;i++) //if repeated elements present 1st remove it..

{

for(j=i+1;j<ele;j++)

{

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

{

for(k=j;k<ele;k++)

a[k]=a[k+1];

j--;

ele--;

}

}

}

printf("\n");

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

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

for(i=0;i<ele-1;i++) //sorting the element using selection sort in increasing order

{

min=i;

for(j=i+1;j<ele;j++)

{

if(a[j]<a[min])

min=j;

}

if(min!=i)

{

int temp;

temp=a[i];

a[i]=a[min];

a[min]=temp;

}

}

printf("\n");

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

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

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

{

if(a[i]+1==a[i+1]) // missing number checking

{

}

else

printf("\nmissing number is=%d",a[i]+1);

}

}

**6) WAP to insert two extra elements in given array, increment first element by one and insert**

**before it and decrement last element by one and insert after it using function.**

**EX: The given array : 10 20 30 40 50**

**o/p array: 11 10 20 30 40 50 49**

**NOTE: use this prototyp: int my\_fun(int\*);**

/\*#include<stdio.h>

main()

{

int a[5],i,ele,min,max;

ele=sizeof(a)/sizeof(a[0]);

printf("enter the element...\n");

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

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

min=a[0];

ele++;

for(i=ele-1;i>=0;i--)

{

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

}

a[0]=min+1;

printf("after adding first .......\n");

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

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

max=a[ele-1];

ele++;

a[ele-1]=max-1;

printf("\n");

printf("after adding first and last.......\n");

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

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

}

**8) WAP to Print Fibonacci Series using recursion**

/\*#include<stdio.h>

void fabnoci(int );

main()

{

int num,n;

printf("enter your elements as you want...\n");

scanf("%d",&num);

fabnoci(num);

}

void fabnoci(int num)

{

static int n1=0,n2=1,n,i=1;

if(i<=num)

{

n=n1+n2;

n1=n2;

n2=n;

printf("%d ",n);

}

i++;

fabnoci(num);

}

**9) WAP to check the given number is palindrome or not using recursion.**

**#include<stdio.h>**

int my\_fun(int\*);

main()

{

int num,i;

printf("enter your number...\n");

scanf("%d",&num);

i=my\_fun(&num);

if(i==1)

printf("given number is palindrome=%d",num);

else

printf("given number is NOT palindrome");

}

int my\_fun(int \*p)

{

int \*temp,sum;

\*temp=\*p;

while(\*temp)

{

sum=sum\*10+(\*temp)%10;

\*temp=\*temp/10;

}

if(sum==\*p)

return 1;

else

return 0;

}

**10) WAP to delete given char from given string using recursion.**

**EX: i/p: ”vector india pvt ltd”**

**o/p: “vecor india pv ld”**

**NOTE: use this prototyp: char\* my\_fun(char\*,char);**

#include<Stdio.h>

char\*my\_fun(char \*,char);

main()

{

char s[20],ch,i,\*p;

printf("enter your sentance..\n");

scanf("%[^\n]",s);

printf("enter your charcter...\n");

scanf(" %c",&ch);

p=my\_fun(s,ch);

for(i=0;p[i];i++)

{

if(p[i]>0)

printf("%c",p[i]);

}

//printf("===%s",p);

}

char \*my\_fun(char \*p,char ch)

{

char c,\*q=p;

if(\*q!='\0')

{

if(\*q==ch)

\*q=-1;

c++;

my\_fun(++q,ch);

}

return p;

}