Q.WAP a function tthat catch two arrays (l elements),revers it & swap it.
Ex: input X array is (1 to 10), Y array is (10 to 100)
OUTPUT:
Y array is (10 to 1), Y array is (100 to 10)

Ans:-

```
#include<stdio.h>
#include<stdlib.h>
#define N 100
int num,i,x[N],y[N],temp,temp1;

void fun_xy(int x[],int num,int y[],int l)
{
	int j;
	for(i=num-1,j=0;i>j;i--,j++)
	{
		temp=x[i];
		x[i]=x[j];
		x[j]=temp;
	}

	for(i=l,j=0;i>j;i--,j++)
	{
		temp1=y[i];
		y[i]=y[j];
		y[j]=temp1;
	}

	printf("\nswap arrays...\n");

	for(i=0,j=0;i<N,j<N;i++,j++)
	{
		temp=x[i];
		x[i]=y[j];
		y[j]=temp;
	}

}
int main()
{
	printf("enter the no.of x elements ...\n");
	scanf("%d",&num);

	for(i=0;i<num;i++)
	{
		x[i]=i+1;
	}

	int l=N-num;
	int n=num;
	for(i=0;i<=l;i++)
	{
```

```
            y[i]=n++;
        }

        fun_xy(x,num,y,l);

        printf("\nx array\n");
        for(i=0;i<=l;i++)
        {
            printf("%d ",x[i]);
        }

        printf("\ny array\n");
        for(i=0;i<num;i++)
            printf("%d ",y[i]);
        printf("\n");

}
```

Q.WAP convert unsigned cahr into binary & storevalue in array?

Ans:-

```
#include<stdio.h>
#include<stdlib.h>
#define MAX 20
int main()
{
    int i,n,array[32],j=0;
    unsigned char ch[MAX];
    printf("enter the charecter numbers ....\n");
    scanf("%s",ch);

    n=atoi(ch);
    printf("n=%d\n",n);
    for(i=31;i>=0;i--)
    {
        array[j++]=n>>i&1;
    }

    printf("array:");
    for(i=0;i<=31;i++)
        printf("%d ",array[i]);
    printf("\n");


}
```

Q.WAP convert Decimal number into BCD?

Ans:-

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

```
int main()
{
	int num,i,n,n1,rev;
	printf("enter the decimal number ....\n");
	scanf("%d",&num);
	while(num)
	{
		n1=num%10;
		rev=rev*10+n1;
		num=num/10;
	}
	printf("n1=%d\n",rev);
	while(rev)
	{
		n=rev%10;
		for(i=3;i>=0;i--)
		{
			printf("%d",n>>i&1);
		}
		printf(" ");
		rev=rev/10;
	}

	return 0;
}
```

Q.reverse the linked list using recursion?

Ans:-

```
// recursion
#include<stdio.h>
#include<stdlib.h>
typedef struct st
{
	int data;
	struct st *next;
}ST;
ST *hptr,*new ,*ptr;
void create();
void insert();
void display();
void reverse(ST * ptr)
{
	printf("in rec fun...%d\n",ptr->data);
	if(ptr->next == NULL)
	{
		hptr=ptr;
		return;
	}
	reverse(ptr->next);
	q =ptr->next;
	q->next=ptr;
```

```
	ptr->next=NULL;
}
int main()
{
	int i,num;
	hptr=NULL;
	printf("enter the number of ...\n");
	scanf("%d",&num);
	for(i=0;i<num;i++)
		insert();

	reverse(hptr);
	display1();

}
void create()
{
	int n;
	new=(ST *)malloc(1*sizeof(ST));
	new->next=NULL;
	printf("enter the node data...\n");
	scanf("%d",&n);
	new->data=n;

}
void insert()
{
	if(hptr==NULL)
	{
		create();
		hptr=new;
	}
	else
	{
		create();
		new->next=hptr;
		hptr=new;
	}
}
void display()
{
	while(hptr)
	{
		printf("%d ",hptr->data);
		hptr=hptr->next;
	}
}
```

Q.swap the two unsigned int without using third variable?

Ans:-

```
#include<stdio.h>
int main()
```

```
{
	unsigned int x,y;
	printf("enter the x and y values....\n");
	scanf("%d %d",&x,&y);
	printf("before swp...x=%d y=%d\n",x,y);

	x=x^y;
	y=x^y;
	x=y^x;
	printf("after swp...x=%d y=%d\n",x,y);

}
```

Q.WAP to find Little and Big Endian ?

Ans:-

```
#include<stdio.h>
int main()
{
	unsigned int n=1;
	char *c = (char *)&n;

	if(*c)
		printf("little endian...\n");
	else
		printf("big endian...\n");

	getchar();
	return 0;
}
```

Q.delete the particular data in an array and remaining data shift to left and right most zeros?
Ex: array is (4,3,2,3,3,5)
delete data 3
OUT PUT: array is (4,2,5,0,0,0)

Ans:-

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
	unsigned long int i,j,*array,n,dn;
	printf("enter no.of elements req....\n");
	scanf("%ld",&n);
	array=(long int *)calloc(n,sizeof(long int));
	if(array==NULL)
	{
		perror("calloc");
		return 0;
	}
	printf("enter the elements of array...\n");
	for(i=0;i<n;i++)
```

```
        scanf("%ld",&array[i]);
    printf("array:");
    for(i=0;i<n;i++)
        printf("%ld ",array[i]);
    printf("\nenter the deleted number in an array...\n");
    scanf("%ld",&dn);

    for(i=0;i<n;i++)
    {
        if((array[i]==dn))
        {
            for(j=i;j<n;j++)
                array[j]=array[j+1];
            i=i-1;
        }
    }
    for(i=0;i<n;i++)
    {
        printf("%ld ",array[i]);
    }
}
```

Q.WAP basic calculator program using two inputs are unsigned characters & oprations +,-,*,/ ?
Ans:-

```
#include<stdio.h>
int main()
{
    double num1,num2, res;
    unsigned char op;
    while(1)
    {
        printf("enter the num1 , + or - or * or /,  num2 ....\n");
        scanf("%lf %c %lf",&num1,&op,&num2);
        switch(op)
        {
            case '+':
                res=num1+num2;
                printf("\n%.3lf %c %.3lf=%.3lf\n",num1,op,num2,res);
                break ;
            case '-':
                if(num1>=num2)
                {
                    res=num1-num2;
                    printf("\n%.3lf %c %.3lf=%.3lf\n",num1,op,num2,res);
                }
                else
                {
                    res=num2-num1;
                    printf("\n%.3lf %c %.3lf=-%.3lf\n",num1,op,num2,res);
                }
                break ;
            case '*':
```

```
                res=num1*num2;
                printf("\n%.3lf %c %.3lf=%.3lf\n",num1,op,num2,res);
                break;
        case '/':
                if(num2==0)
                        printf("Not Divisible by 0! Start Again!\n");
                else{
                        res= (float)(num1/num2);
                        printf("\n%.3lf %c %.3lf=%f\n",num1,op,num2,res);
                }
                break;
        default:
                printf("unkown option entered....\n");
                break;

    }


  }

}
```

Q.WAP to find out duplicate elements in an array that array contains 101 elements(1 to 100 & un shorted)

Ans:-

```
#include<stdio.h>
#define MAX 20
int main()
{
    int array[MAX];
    int ar_sum=0;//sum of array elements
    int sum=0; // sum of 1 to max-1
    int i;
    for(i=0;i<MAX;i++)
    {
        scanf("%d",&array[i]);
        ar_sum +=array[i];
    }
    //ar_sum-((no.of elements in array= MAX)*(MAX-1)/2);
    sum=(MAX*(MAX-1))/2;
    printf("\nDuplicate number is %d\n",ar_sum-sum);

}
```

Q.write a function that takes one argument is rows print * and space (1,3,5,.....)

EX:input 3 rows

1space *

3spaces *

5spaces *

.....

Ans:-

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

```
int i,num,j,count=0;
void fun_patt(int num);
int main()
{
	printf("enter the number of rows...\n");
	scanf("%d",&num);
fun_patt(num);

}
void fun_patt(int num)
{
	for(i=1;i>0;i++)
	{
		if(i%2!=0)
		{
			count++;
			for(j=0;j<i;j++)
				printf("-");
			printf("*");
			printf("\n");
			if(count==num)
				break;
		}

	}
}
```