UNIT 6

ARRAY

A. <u>One-Dimensional Array</u>

1. WAP that reads 10 float numbers from user and displays the entered numbers in the screen.

```c
#include<stdio.h>
int main(){
        float num[10];
        printf("Enter 10 numbers:");
        for(int i=0;i<10;i++){
                scanf("%f",&num[i]);
        }
        printf("Entered numbers are:\n");
        for(int i=0;i<10;i++){
                printf("num[%d] = %f\n",i,num[i]);
        }
        return 0;
}
```

B. <u>Two-Dimensional Array</u>

2. WAP to read a matrix of size M*N from user and display it on screen.

```c
#include<stdio.h>
int main(){
        int a[10][10],i,j,M,N;
        printf("Enter rows of matrix:");
        scanf("%d",&M);
        printf("Enter columns of matrix:");
        scanf("%d",&N);
        printf("Enter the elements of matrix:\n");
        for(i=0;i<M;i++){
                for(j=0;j<N;j++){
                        scanf("%d",&a[i][j]);
                }
        }
        printf("Entered matrix is:\n");
        for(i=0;i<M;i++){
                for(j=0;j<N;j++){
                        printf("%d\t",a[i][j]);
                }
                printf("\n");
        }
        return 0;
}
```

3. WAP to read two M*N matrices and display their sum or difference.

```c
#include<stdio.h>

#define M 3

#define N 3

int main(){

        int a[M][N],b[M][N],i,j;

        printf("Enter the elements of 1st matrix:\n");

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

                for(j=0;j<N;j++){

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

                }

        }

        printf("The 1st matrix is:\n");

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

                for(j=0;j<N;j++){

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

                }

                printf("\n");

        }

        printf("Enter the elements of 2nd matrix:\n");

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

                for(j=0;j<N;j++){

                        scanf("%d",&b[i][j]);
```

```
        }

    }

    printf("The 2nd matrix is:\n");

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

        for(j=0;j<N;j++){

            printf("%d\t",b[i][j]);

        }

        printf("\n");

    }

    printf("The sum of matrices is:\n");

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

        for(j=0;j<N;j++){

            printf("%d\t",a[i][j]+b[i][j]);

        }

        printf("\n");

    }

    return 0;

}
```

4. WAP to find transpose of a matrix.

```c
#include<stdio.h>

#define M 3

#define N 3

int main(){

        int a[M][N],b[M][N],i,j;

        printf("Enter the elements matrix:\n");

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

                for(j=0;j<N;j++){

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

                }

        }

        printf("The matrix to be transposed is:\n");

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

                for(j=0;j<N;j++){

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

                }

                printf("\n");

        }

        printf("The transposed matrix is:\n");

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

                for(j=0;j<N;j++){

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

                }
```

```
        printf("\n");

    }

    return 0;

}
```

5. WAP to find the sum of squares in a diagonal of a square matrix.

```
#include<stdio.h>

#include<stdlib.h>

int main(){

    int a[10][10],i,j,M,N,sum=0;

    printf("Enter order of matrix:");

    scanf("%d%d",&M,&N);

    if(M!=N){

        printf("Not square matrix.\n");

        exit(0);

    }

    if(M>10 || N>10){

        printf("The order is out of range.\n");

        exit(0);

    }

    printf("Enter the elements matrix:\n");

    for(i=0;i<M;i++){
```

```c
        for(j=0;j<N;j++){

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

                if(i==j){

                        sum+=a[i][j]*a[i][j];

                }

        }

}

printf("Entered matrix is:\n");

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

        for(j=0;j<N;j++){

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

        }

        printf("\n");

}

printf("Sum of squares in a diagonal is %d.",sum);

return 0;

}
```

6. WAP to read two matrices from user and multiply them if possible.

```c
#include<stdio.h>
int main(){
        int a[10][10],b[10][10],s[10][10];
        int m,n,x,y,i,j,k;
        printf("Enter number of rows in 1st matrix:");
        scanf("%d",&m);
        printf("Enter number of columns in 1st matrix:");
        scanf("%d",&n);
        printf("Enter number of rows in 2nd matrix:");
        scanf("%d",&x);
        printf("Enter number of columns in 2nd matrix:");
        scanf("%d",&y);
        if(n!=x){
                printf("Matrix multiplication is not possible.");
        }
        else{
                printf("Enter the elements of 1st matrix:\n");
                for(i=0;i<m;i++){
                        for(j=0;j<n;j++){
                                scanf("%d",&a[i][j]);
                        }
                }
                printf("The 1st matrix is:\n");
```

```c
for(i=0;i<m;i++){

        for(j=0;j<n;j++){

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

        }

        printf("\n");

}

printf("Enter the elements of 2nd matrix:\n");

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

        for(j=0;j<y;j++){

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

        }

}

printf("The 2nd matrix is:\n");

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

        for(j=0;j<y;j++){

                printf("%d\t",b[i][j]);

        }

        printf("\n");

}

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

        for(j=0;j<y;j++){

                s[i][j]=0;

                for(k=0;k<n;k++){

                        s[i][j]=s[i][j]+a[i][k]*b[k][j];
```

```
                    }

                }

            }

            printf("The multiplication of matrices is:\n");

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

                    for(j=0;j<y;j++){

                            printf("%d\t",s[i][j]);

                    }

                    printf("\n");

            }

    }

    return 0;

}
```

7. Write a menu-driven program to input a 3*3 matrix and display the menu.

Menu:

- **Print the input matrix**
- **Sum of even values of elements**
- **Sum of all diagonal elements**
- **Exit**

```
#include<stdio.h>

#define M 3

#define N 3

int main(){

    int a[3][3],sum;
```

```c
int i,j,choice;
printf("Enter the elements of matrix:\n");
for(i=0;i<M;i++){
        for(j=0;j<N;j++){
                scanf("%d",&a[i][j]);
        }
}
printf("\nEnter 1 to display matrix.");
printf("\nEnter 2 to display sum of even values of elements.");
printf("\nEnter 3 to display sum of all diagonal elements.");
printf("\nEnter 4 to exit.");
do{
        printf("\nEnter your choice:");
        scanf("%d",&choice);
        switch(choice){
                case 1:
                        printf("The matrix is:\n");
                                for(i=0;i<M;i++){
                                        for(j=0;j<N;j++){
                                                printf("%d\t",a[i][j]);
                                        }
                                printf("\n");
                                }
                                break;
```

```
case 2:

        sum=0;

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

                for(j=0;j<N;j++){

                        if(a[i][j]%2==0){

                                sum+=a[i][j];

                        }

                }

        }

        printf("Sum of even values of elements = %d",sum);

        break;

case 3:

        sum=0;

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

                for(j=0;j<N;j++){

                        if(i==j){

                                sum+=a[i][j];

                        }

                }

        }

        printf("Sum of all diagonal elements = %d",sum);

        break;

case 4:

        break;
```

```
                default:

                        printf("Invalid option. Please try again.");

                }

        }while(choice!=4);

        return 0;

}
```

8. WAP that accepts the elements of 3*3 matrix and calculate the sum of all elements of the matrix.

```
#include<stdio.h>

#define M 3

#define N 3

int main(){

        int a[M][N],sum=0;

        int i,j;

        printf("Enter the elements of matrix:\n");

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

                for(j=0;j<N;j++){

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

                        sum+=a[i][j];

                }

        }

        printf("Sum of all elements = %d",sum);

        return 0;

}
```

C. Strings

9. WAP to illustrate the use of strcpy() function.

```c
#include<stdio.h>
#include<string.h>
int main(){
        char a[12]="Hello World";
        char b[50];
        strcpy(b,a);
        printf("Copied string = %s",b);
        return 0;
}
```

10. WAP to find the length of a string using library function strlen().

```c
#include<stdio.h>
#include<string.h>
int main(){
        char name[50];
        int length;
        printf("Enter your name:");
        scanf("%s",name);
        length=strlen(name);
        printf("Length of name = %d.",length);
        return 0;
}
```

11. WAP to concatenate two strings using strcat() library functions.

```c
#include<stdio.h>

#include<string.h>

int main(){

        char s1[10]="Happy ";

        char s2[]="New Year";

        printf("s1 = %s\ns2 = %s",s1,s2);

        printf("\nAfter concatenating = %s",strcat(s1,s2));

        return 0;

}
```

12. WAP to reverse a string using library functions:

```c
#include<stdio.h>

#include<string.h>

int main(){

        char string[100];

        printf("Enter a string:\n");

        gets(string);

        strrev(string);

        printf("String after reversing = %s",string);

        return 0;

}
```

13. WAP to read a string and count the number of vowels and consonants in it.

```c
#include<stdio.h>

#include<string.h>

int main(){

    char string[100];

    int i,vowels=0,consonants=0;

    printf("Enter a string:\n");

    gets(string);

    strlwr(string);

    for(i=0;string[i]!='\0';i++){

    if(string[i]=='a'||string[i]=='e'||string[i]=='i'||string[i]=='o'||string[i]=='u'){

            vowels++;

        }
        else{

            consonants++;

        }

    }

    printf("Number of vowels = %d\nNumber of consonants =
%d",vowels,consonants);

    return 0;

}
```

14. WAP to check whether a given string is a palindrome or not.

```c
#include<stdio.h>

#include<string.h>

int main(){

        char string1[100],string2[100];

        printf("Enter a string:\n");

        gets(string1);

        strcpy(string2,string1);

        strrev(string1);

        if(strcmp(string1,string2)==0){

                printf("Palindrome.");

        }

        else{

                printf("Not Palindrome.");

        }

        return 0;

}
```

15. WAP to convert a string to uppercase using strupr() library functions.

```c
#include<stdio.h>
#include<string.h>
int main(){
        char string[100];
        printf("Enter a string:\n");
        gets(string);
        strupr(string);
        printf("String in uppercase = %s",string);
        return 0;
}
```

16. WAP to convert a string to lowercase using strlwr() library function.

```c
#include<stdio.h>
#include<string.h>
int main(){
        char string[100];
        printf("Enter a string in uppercase:\n");
        gets(string);
        strlwr(string);
        printf("String in lowercase = %s",string);
        return 0;
}
```

17. WAP to compare two strings using strcmp() library function.

```c
#include<stdio.h>

#include<string.h>

int main(){

        char string1[100],string2[100];

        int result;

        printf("Enter 1st string:\n");

        gets(string1);

        printf("Enter 2nd string:\n");

        gets(string2);

        result=strcmp(string1,string2);

        if(result==0){

                printf("Two strings are identical.");

        }

        else if(result>0){

                printf("%s is greater than %s.",string1,string2);

        }

        else{

                printf("%s is greater than %s.",string2,string1);

        }

        return 0;

}
```