**What is definition of Array:**

An array is a data structure in computer programming that stores a collection of elements, each identified by an index or a key. The elements stored in an array can be of any data type, such as integers, characters, or structures, and they all have the same data type.

In C, arrays are declared by specifying the data type of the elements, the number of elements, and the name of the array. For example:

int numbers[10];

This declares an array named numbers that can store 10 integer values. The elements of an array are accessed by using the array name followed by the index of the element in square brackets, starting from 0. For example:

numbers[0] = 10;

numbers[1] = 20;

Arrays are useful for storing and manipulating large amounts of data, and they provide efficient access to the elements by index.

**Write a C program for matrix addition and subtraction:**

#include<stdio.h>

#define ROW 3

#define COL 3

void printMatrix(int mat[][COL], char name) {

printf("\nMatrix %c\n", name);

for(int i=0; i<ROW; i++) {

for(int j=0; j<COL; j++) {

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

}

printf("\n");

}

}

void addMatrix(int matA[][COL], int matB[][COL], int result[][COL]) {

for(int i=0; i<ROW; i++) {

for(int j=0; j<COL; j++) {

result[i][j] = matA[i][j] + matB[i][j];

}

}

}

void subtractMatrix(int matA[][COL], int matB[][COL], int result[][COL]) {

for(int i=0; i<ROW; i++) {

for(int j=0; j<COL; j++) {

result[i][j] = matA[i][j] - matB[i][j];

}

}

}

int main() {

int matA[ROW][COL] = {{1,2,3}, {4,5,6}, {7,8,9}};

int matB[ROW][COL] = {{9,8,7}, {6,5,4}, {3,2,1}};

int addResult[ROW][COL];

int subtractResult[ROW][COL];

addMatrix(matA, matB, addResult);

subtractMatrix(matA, matB, subtractResult);

printMatrix(matA, 'A');

printMatrix(matB, 'B');

printMatrix(addResult, '+');

printMatrix(subtractResult, '-');

return 0;

}

This program uses two 3x3 matrices matA and matB, and calculates their addition and subtraction results, which are stored in add Result and subtract Result matrices respectively. The printMatrix function prints the given matrix with a specified name, and addMatrix and subtractMatrix functions perform matrix addition and subtraction respectively.

**String: Definition:**

A string is a sequence of characters in computer programming. In C, a string is represented as an array of characters, where the end of the string is marked by a special character called the null terminator '\0'. The null terminator signals the end of the string, and it is not considered as part of the string itself.

Strings in C can be declared and initialized in the following ways:

char greeting[6] = {'H', 'e', 'l', 'l', 'o', '\0'};

char greeting[] = "Hello";

In the first example, the string is declared as an array of characters with a size of 6, and it is initialized with the characters 'H', 'e', 'l', 'l', 'o', and the null terminator '\0'.

In the second example, the string is declared as an array of characters without specifying the size, and it is initialized with the string literal "Hello". The compiler automatically determines the size of the array based on the length of the string literal, including the null terminator.

Strings are commonly used in programs to store and manipulate text data, and they provide many functions and library routines for string manipulation, such as concatenation, comparison, search, and conversion to and from other data types.

A string function is a predefined or user-defined function in computer programming that operates on strings and manipulates their contents. In C, there are many standard string functions provided by the C standard library, such as:

strlen: returns the length of a string, excluding the null terminator

strcpy: copies the contents of one string to another

strcat: concatenates two strings into a single string

strcmp: compares two strings lexicographically and returns an integer indicating their ordering

strstr: searches for a substring within a string and returns a pointer to its first occurrence

sprintf: formats a string and stores the result in a character array

sscanf: reads formatted data from a string and stores it in variables

Here's an example of using the strlen function:

#include <stdio.h>

#include <string.h>

int main() {

char greeting[] = "Hello, World!";

int length = strlen(greeting);

printf("The length of the string is %d\n", length);

return 0;

}

This program declares a string greeting and calculates its length using the strlen function. The length is then printed to the console. The strlen function is declared in the string.h header file, which must be included in the program.