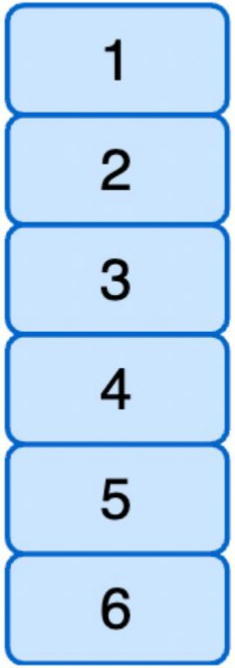
Arrays

Sumaiyah Zahid

Array Definition

Consecutive memory location with same name and type just like consecutive bars of chocolates with same taste and size.

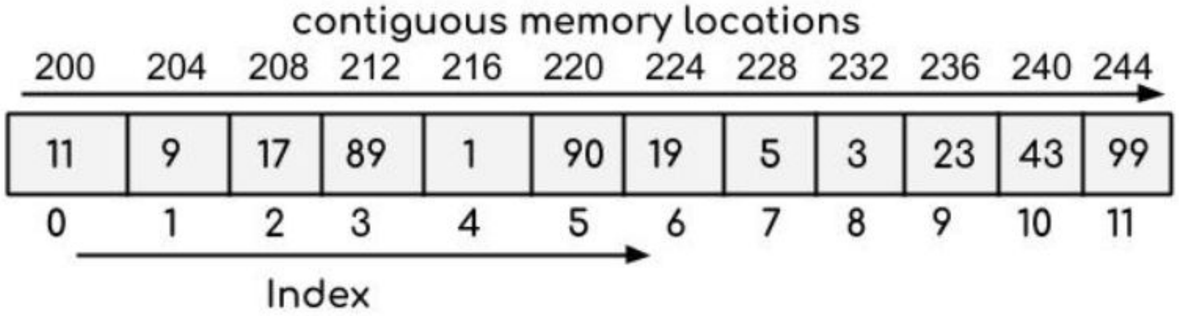


Array Definition

We have houses lanes for 80 yds, 120 yds, 240 yds etc.

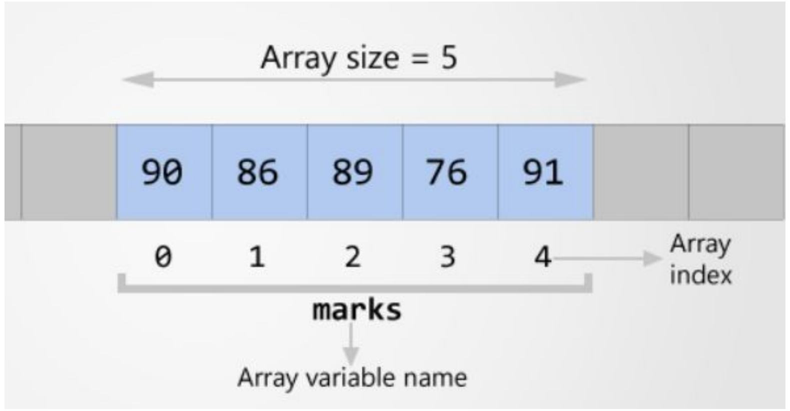
Similarly we have int array , char array, float array etc.

e.g int array->



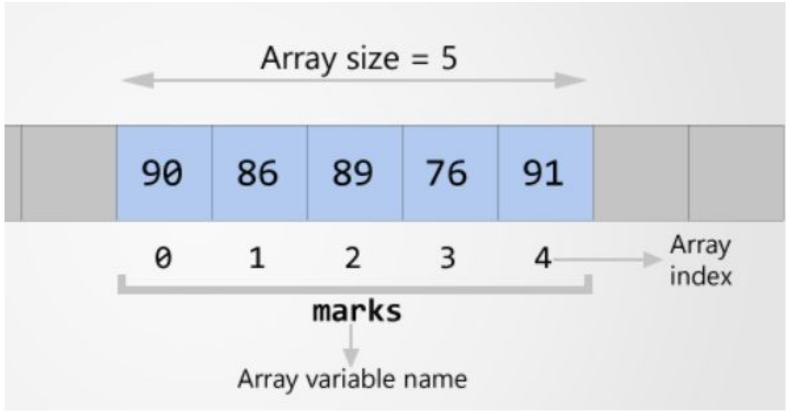
Declaration

```
Datatype ArrayName [NumberOfElements];

int marks[5]; // Declaration

int marks[5]={90,86,89,76,91}; //Declaration & initialization

int marks[]={90,86,89,76,91};


printf(“%d”, marks[2]);
```



Accessing Array Element

Make a c program to add all the marks into another variable sum and display the output.



```c
#include <stdio.h>
int main()
{
    int marks[5]={90,86,89,76,91},i,sum=0;

    for(i=0; i<5; i++)
    {
        sum=sum + marks[i];
    }
    printf("%d", sum);

    return 0;
}
```

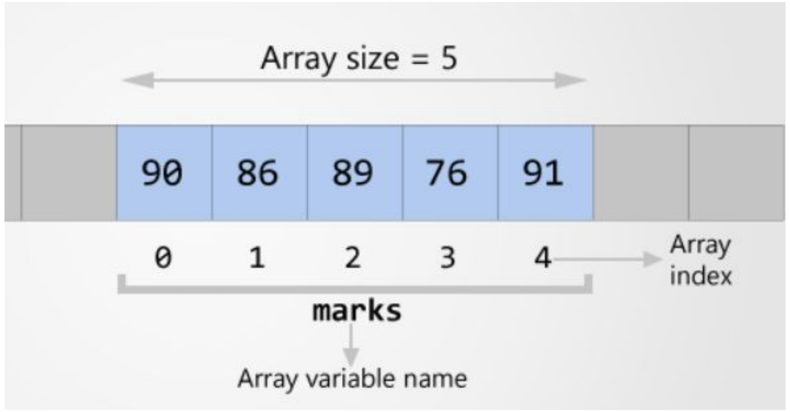
Accessing Array Element

Make a c program to declare an array of size 5 and then take user input to initialize that array.

```c
#include <stdio.h>
int main()
{
    int marks[5],i;

    for(i=0; i<5; i++)
    {
        printf("Enter number at %d location ", i);
        scanf("%d",&marks[i]);
    }
    return 0;
}
```

Linear Search an Array

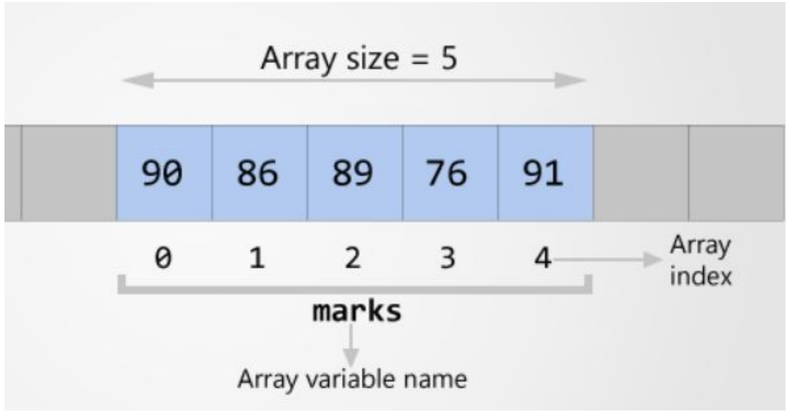
Make a c program to find the highest marks and display the output.



```c
#include <stdio.h>
int main()
{
   int marks[5]={90,86,89,76,91},i,high=0;

   for(i=0; i<5; i++)
   {
       if(marks[i] > high )
       {
           high=marks[i];
       }

   }
   printf("%d", high);

   return 0;
}
```

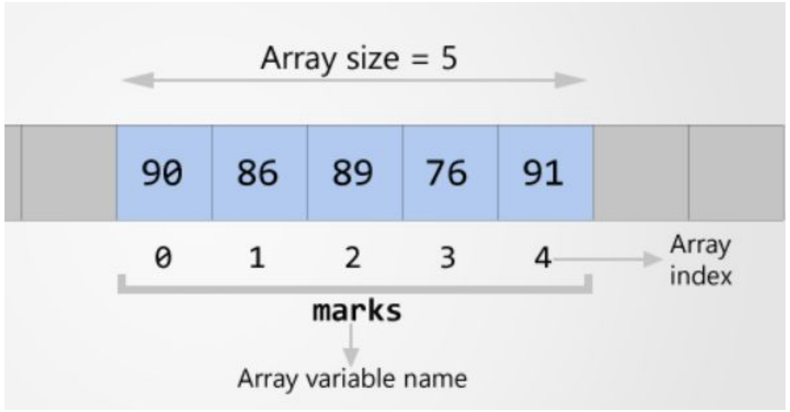
Home Assignment

Make a c program to find the lowest marks and display the output.

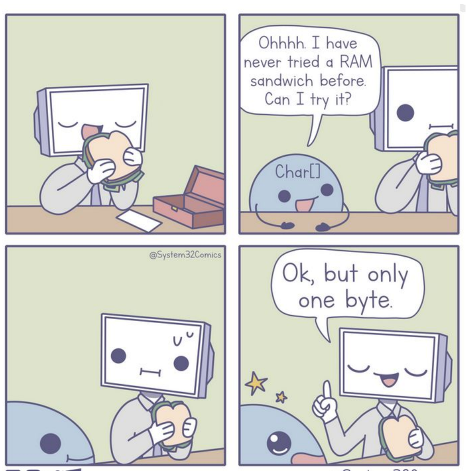


Linear Search an Array

Make a c program to find the marks=86 and display the index of array where it exist.



```c
#include <stdio.h>
int main()
{
   int marks[5]={90,86,89,76,91},i;

   for(i=0; i<5; i++)
   {
      if(marks[i] == 86 )
       {
          break;
       }
   }
   printf("%d", i);
   return 0;
}
```


Ohhhh. I have never tried a RAM sandwich before. Can I try it?
Char[]
@System32Comics
Ok, but only one byte.

STRINGS

Char Array

```
char name[30]="FAST NUCES"; //10 characters
int i;
for(i=0; i<10; i++)
{
     printf("%c",name[i]); // FAST NUCES will print
}
printf("%s",name); // FAST NUCES will print
```

String \ Char array

```
char name[]={‘F’,’A’,’S’,’T’,’ ‘ ,’N’, ‘U’, ‘\0’};

char name[]="FAST NU";


\0 = Null character or String Terminator


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

   { printf("%c",name[i]);}
```

String

```c
#include <stdio.h>
int main()
{
    char name[20];
    scanf("%s", name); // No need of &name
    printf("Your name is %s", name);
    return 0; // Array name alone works as a base address
}
```

Tasks

Given a string of “Fast Nuces !(1234)”

- Find out count of letter s
- Find out count of capital and small letters
- Find the count of special characters
- Find the count of digits.

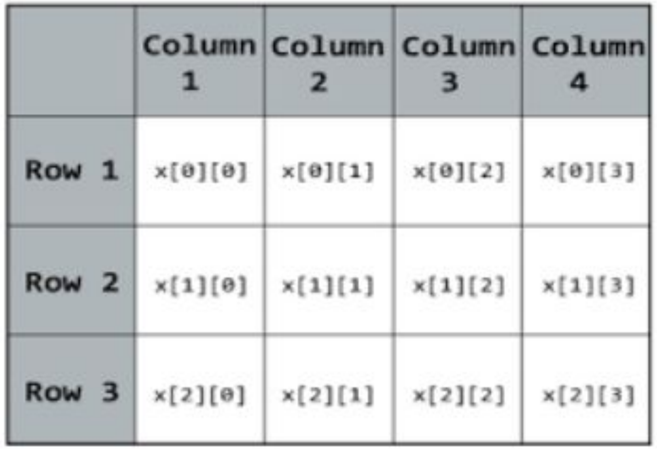
Tasks

Write a program to change the case of all the alphabets in an array of strings.

Write a program that counts the no. of upper and lower case letters in an array of strings

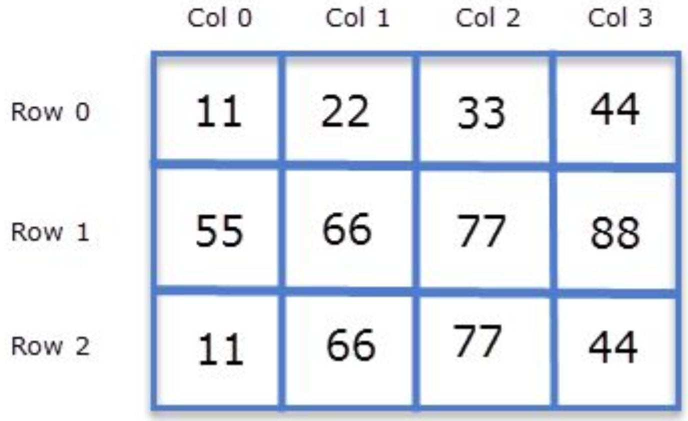
2D Array

```
int array [2][3];
int array [row][column];
int array[3][4] = {
    {10, 11, 12, 13},
    {14, 15, 16, 17},
    {18, 19, 20, 21},
};
```

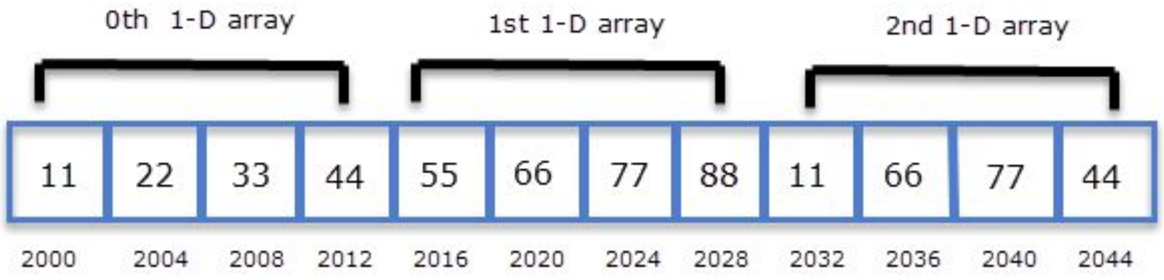


	Column 1	Column 2	Column 3	Column 4
Row 1	x[0][0]	x[0][1]	x[0][2]	x[0][3]
Row 2	x[1][0]	x[1][1]	x[1][2]	x[1][3]
Row 3	x[2][0]	x[2][1]	x[2][2]	x[2][3]

2D Array

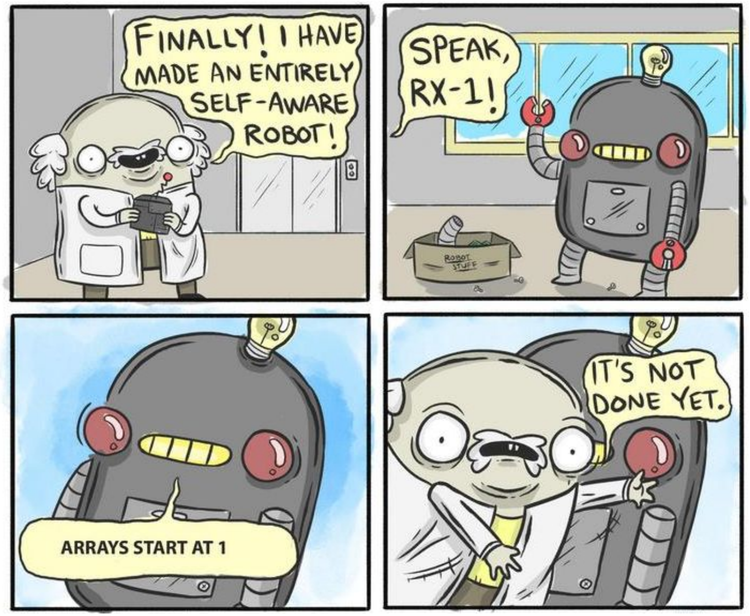


	Col 0	Col 1	Col 2	Col 3
Row 0	11	22	33	44
Row 1	55	66	77	88
Row 2	11	66	77	44



2D Array

Make a c program to declare a 2D-array of 3 rows and 4 columns and then take user input to initialize that array.



```c
#include <stdio.h>
int main()
{
    int array[3][4], i,j;

    for(i=0; i<3; i++)
    {
        for(j=0; j<4; j++)
        {
            scanf("%d", &array[i][j]);
        }
    }
    return 0;
}
```

2D Array

Make a c program to add two matrix and store there result in 3rd matrix.

```c
#include <stdio.h>
int main()
{
   int array1[3][4] = {
    {10, 11, 12, 13},
    {14, 15, 16, 17},
    {18, 19, 20, 21} };

    int array2[3][4] = {
    {10, 11, 12, 13},
    {14, 15, 16, 17},
    {18, 19, 20, 21} };

    int  array3[3][4],i,j;
   for(i=0; i<3; i++)
   {
      for(j=0; j<4; j++)
    {
        array3[i][j]=array1[i][j] + array2[i][j];
    }
   }
   for(i=0; i<3; i++)
   {
    for(j=0; j<4; j++)
    {
        printf("%d ",array3[i][j]);
    }
    printf("\n");
   }
   return 0;
}
```

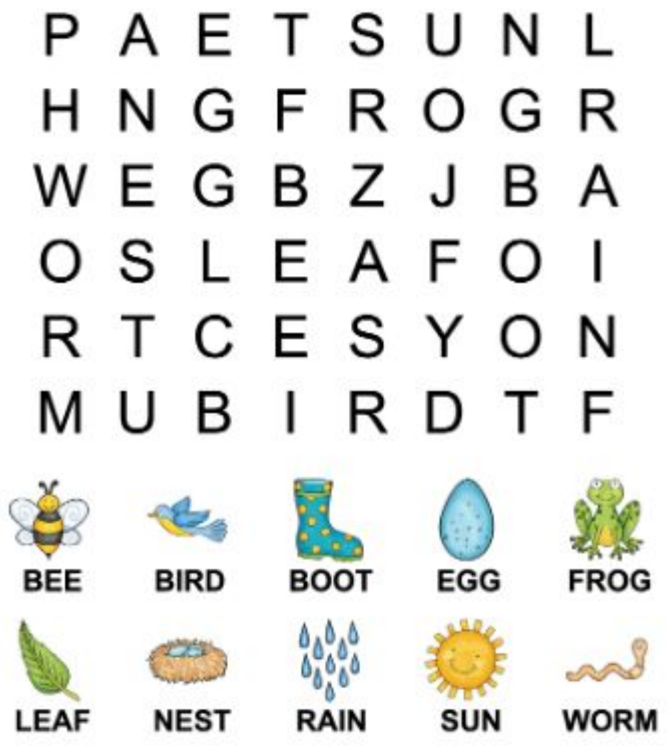
Home Assignment

Make a c program to multiply two matrix and store there result in 3rd matrix.

Make a c program to transpose a matrix.

2D Char Array

Make a c program to find words in a puzzle.



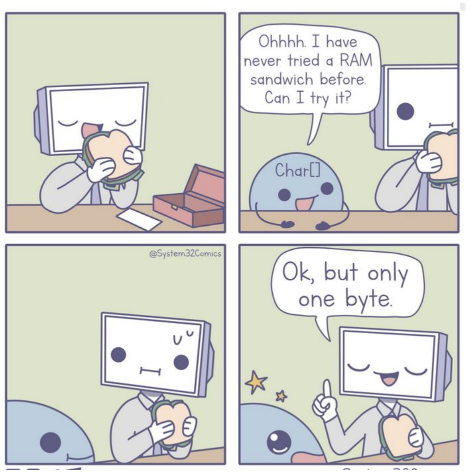
P	A	E	T	S	U	N	L
H	N	G	F	R	O	G	R
W	E	G	B	Z	J	B	A
O	S	L	E	A	F	O	I
R	T	C	E	S	Y	O	N
M	U	B	I	R	D	T	F

Arrays Practice Problem

Write a Program to check whether a given matrix is an identity matrix or not.
Write a Program to find whether the given is the matrix is diagonal or not.
Write a Program to display an upper triangular matrix.
Write a Program to check whether a matrix is symmetric or not.
Write a Program to find the sum of an upper triangular matrix.
Write a Program to find the maximum element in the 2D matrix.
Write a Program to find the position of an element in a 2d array or Matrix.
https://www.examveda.com/c-program/practice-mcq-question-on-arrays-and-strings/

Multi-Dimensional Array

```
int Ferrero [2][2][4];

int array [z][x][y];

int array[2][3][4] ;
 {
   { {0,1,2,3}, {4,5,6,7}, {8,9,10,11} },
   { {12,13,14,15}, {16,17,18,19}, {20,21,22,23} }
 };
```


Ohhhh. I have never tried a RAM sandwich before. Can I try it?
Char[]
@System32Comics
Ok, but only one byte.

STRINGS

String \ Char array

```
char name[]={‘F’,’A’,’S’,’T’,’ ‘ ,’N’, ‘U’, ‘\0’};
char name[]="FAST NU";


\0 = Null character or String Terminator


for(i=0; i<10; i++)
   { printf("%c",name[i]);}
```

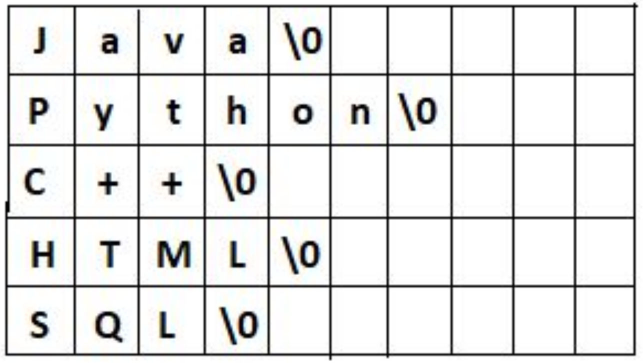
String

```
#include <stdio.h>
int main()
{
    char name[20];
    scanf("%s", name); // No need of &name
    printf("Your name is %s", name);
    return 0; // Array name alone works as a base address
}
```

Gets and Puts

```
#include <stdio.h>
int main()
{
    char name[20];
    puts("Enter your name");
    gets(name);
    puts(name);
    return 0;
}
```

2D Strings



J	a	v	a	\0					
P	y	t	h	o	n	\0			
C	+	+	\0						
H	T	M	L	\0					
S	Q	L	\0						

```
char language[5][10] =
{
  {'J','a','v','a','\0'},   {'P','y','t','h','o','n','\0'},
  {'C','+','+','\0'},  {'H','T','M','L','\0'},
  {'S','Q','L','\0'} };


char language[5][10] = {"Java", "Python", "C++", "HTML",
"SQL"};
```

2D Strings

```
// it is valid

char language[ ][10] = {"Java", "Python", "C++", "HTML", "SQL"};


// invalid

char language[ ][ ] = {"Java", "Python", "C++", "HTML", "SQL"};

// invalid

char language[5][ ] = {"Java", "Python", "C++", "HTML", "SQL"};
```

```c
#include <stdio.h>
int main()
{   int i;
    char language[5][10] = {"Java",
"Python", "C++", "HTML", "SQL"};

    for(i=0; i<5; i++)
        printf("%s\n", language[i]);
    return 0;
}
```

```c
#include <stdio.h>
int main()
{  int i;
   char name[5][10];
  for(i=0; i<5; i++)
       scanf("%s", name[i]);

   for(i=0; i<5; i++)
       printf("%s\n", name[i]);
   return 0;
}
```

```c
#include <stdio.h>
int main()
{   int i,j;
    char language[5][10] = {"Java", "Python", "C++", "HTML", "SQL"};

    for(i=0; i<5; i++)
    {
        for(j=0; language[i][j]!='\0'; j++)
        {
            printf("%c", language[i][j]);
        }
       printf("\n");
    }
    return 0;
}
```

HOME ASSIGNMENT

Write a program to change the case of all the alphabets in an array of strings.

Write a program that counts the no. of upper and lower case letters in an array of strings

String.h ------> strlen

```
size_t strlen(const char *str);
```

Computes the length of the string str up to but not including the terminating null character.

Returns the number of characters in the string.

String.h ------> strcmp

```
int strcmp(const char *str1, const char *str2)
```

It compares the two strings and returns an integer value.

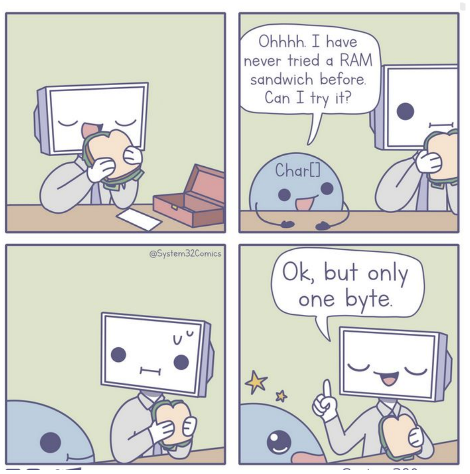
- If Return value < 0 then it indicates str1 is less than str2.
- If Return value > 0 then it indicates str2 is less than str1.
- If Return value = 0 then it indicates str1 is equal to str2.

String.h ------> strncmp

```
int strncmp(const char *str1, const char *str2, size_t n)
```

It compares both the string till n characters or in other words it compares first n characters of both the strings.

```c
#include <stdio.h>
#include<string.h>
int main()
{
   char name1[20]="FAST";
   char name2[20]= "NUCES";
   printf("Length of string is %d and %d \n",
   strlen(name1), strlen(name2));
   int i=strcmp(name1, "FAST");
   int j=strcmp(name1, name2);
   printf("Comparison result is %d %d", i,j);
   return 0;
}
```


Ohhhh. I have never tried a RAM sandwich before. Can I try it?
Char[]
@System32Comics
Ok, but only one byte.

STRINGS FUNCTIONS

String.h ------> strcat

```
char *strcat(char *str1, char *str2)
```

It concatenates two strings and returns the combined string.

String.h ------> strncat

```
char *strncat(char *str1, char *str2, int n)
```

```
It concatenates n characters of str2 to string str1.
```

String.h ------> strcpy

```
char *strcpy(char *str1, char *str2)
```

It copies the string str2 into string str1, including the end character (terminator char ‘\0’).

String.h ------> strncpy

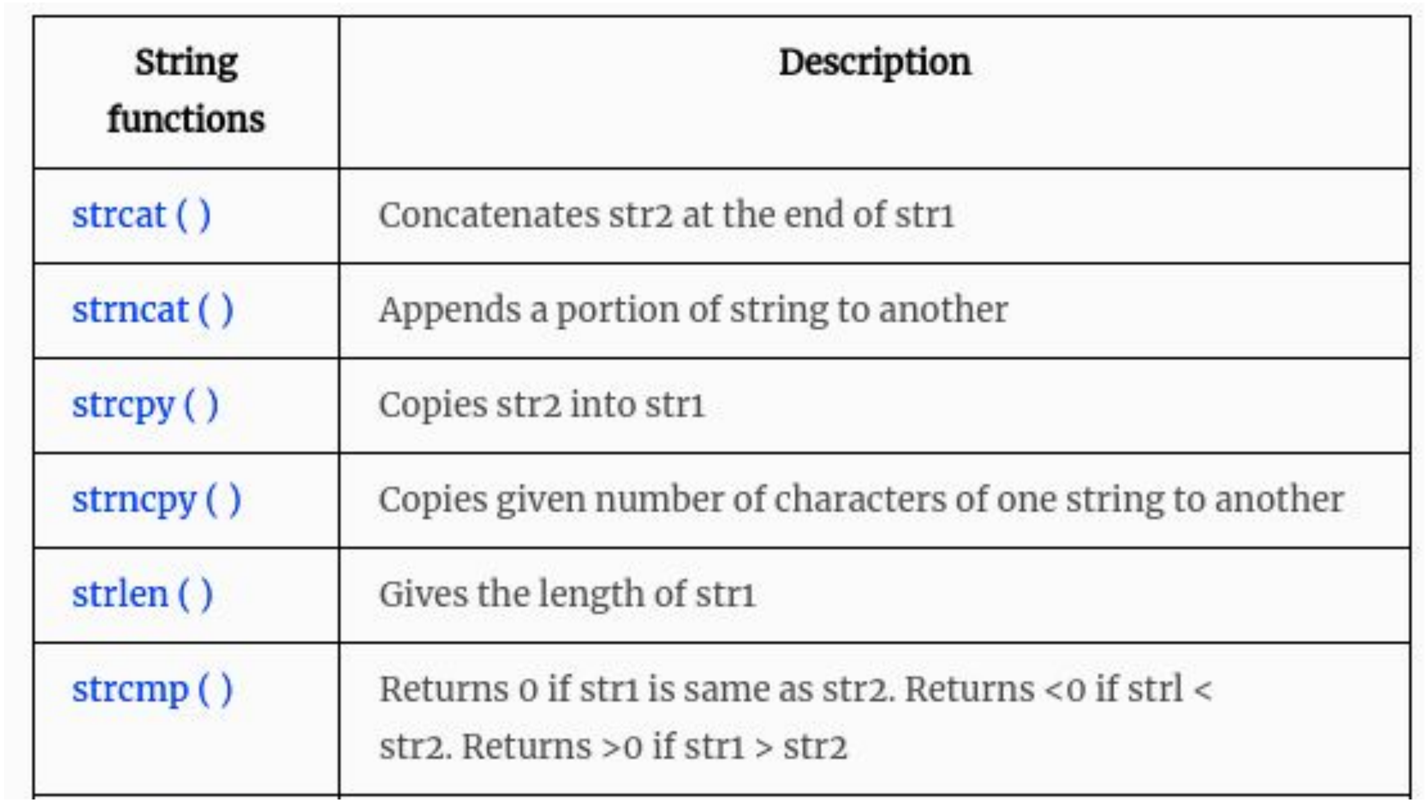
```
char *strncpy(char *str1, char *str2, int n)
```

It copies the n characters of str2 into string str1.

Try These

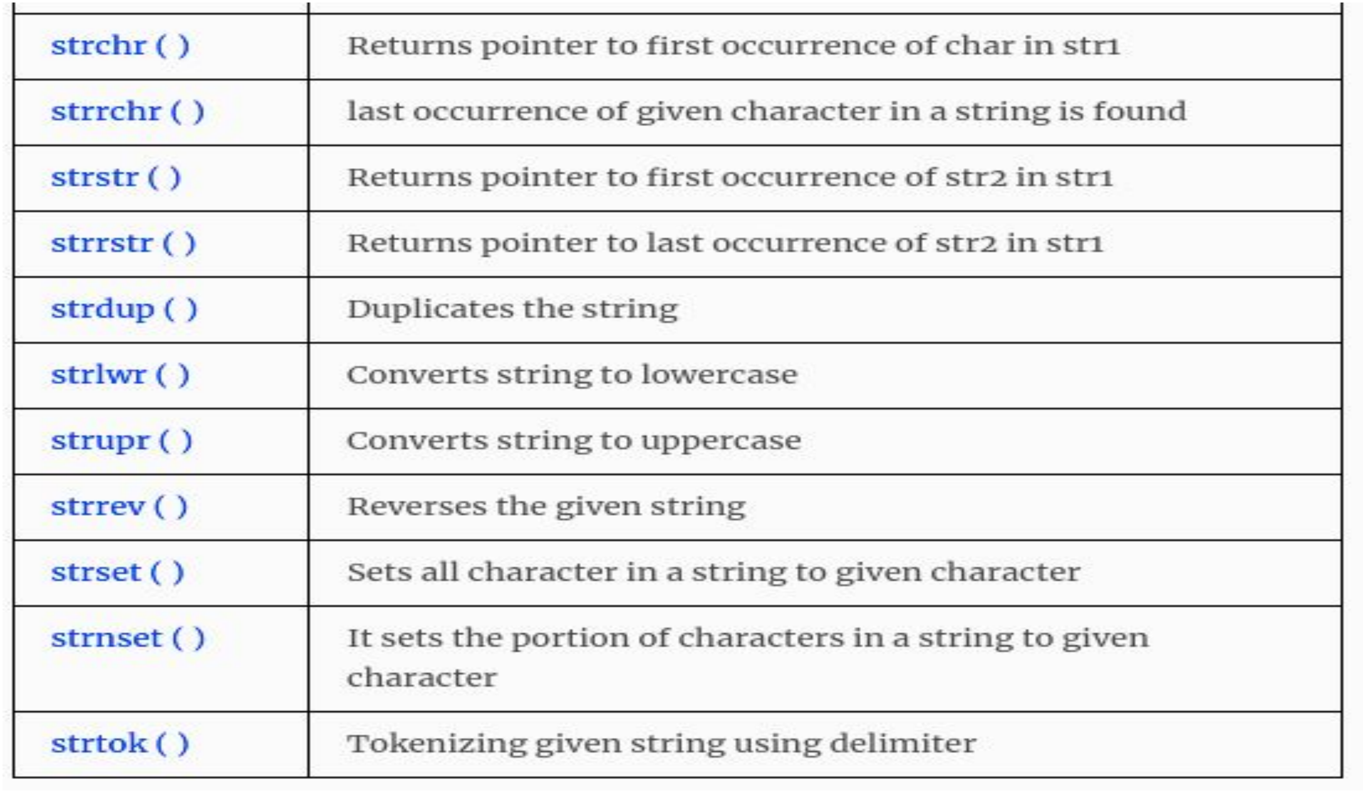
```
strcat(str1, str2)

strncat(str1,str2, 5)

strcpy(str1,str2)

strncpy(str1,str2,5)
```

HTTPS://FRESH2REFRESH.COM/C-PROGRAMMING/C-STRINGS/

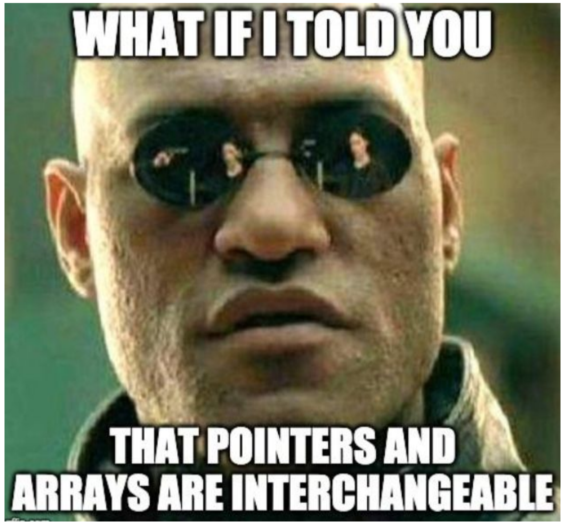


String functions	Description
strcat ( )	Concatenates str2 at the end of str1
strncat ( )	Appends a portion of string to another
strcpy ( )	Copies str2 into str1
strncpy ( )	Copies given number of characters of one string to another
strlen ( )	Gives the length of str1
strcmp ( )	Returns 0 if str1 is same as str2. Returns <0 if strl < str2. Returns >0 if str1 > str2

HTTPS://FRESH2REFRESH.COM/C-PROGRAMMING/C-STRINGS/



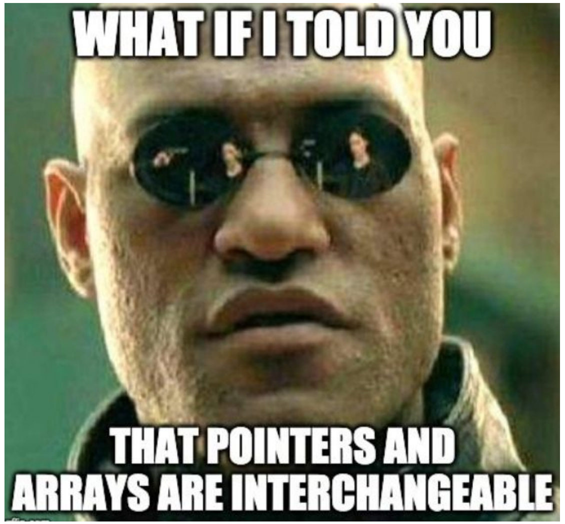
strchr ( )	Returns pointer to first occurrence of char in str1
strrchr ( )	last occurrence of given character in a string is found
strstr ( )	Returns pointer to first occurrence of str2 in str1
strrstr ( )	Returns pointer to last occurrence of str2 in str1
strdup ( )	Duplicates the string
strlwr ( )	Converts string to lowercase
strupr ( )	Converts string to uppercase
strrev ( )	Reverses the given string
strset ( )	Sets all character in a string to given character
strnset ( )	It sets the portion of characters in a string to given character
strtok ( )	Tokenizing given string using delimiter



Arrays as a pointers

Arrays As a Pointers

```
int marks[]={90,86,89,76,91};

printf(“%d”, marks[2]);

printf(“%d”, marks); ????
```



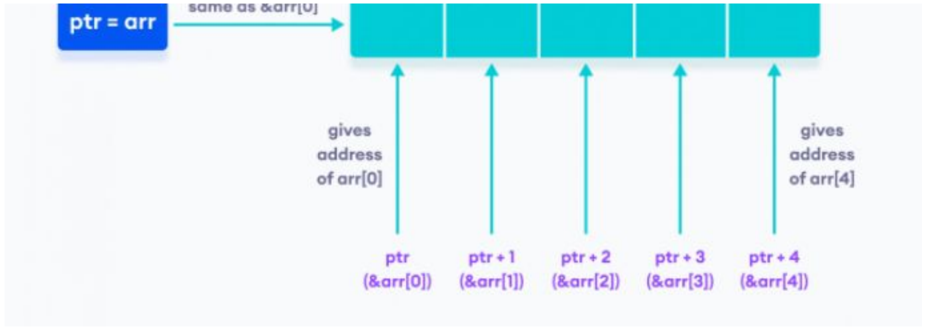
Array name holds the starting address of that array i.e.

```
    marks = & (marks[0])
```

Arrays are not variables, but pointer-variable.

Arrays As a Pointers

```
int arr[]={90,86,89,76,91};

int *ptr=arr;

ptr = arr

arr = &arr[0]
```

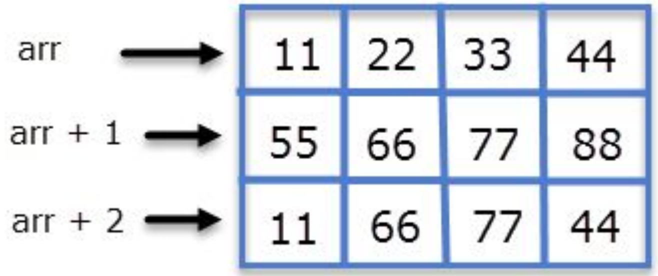


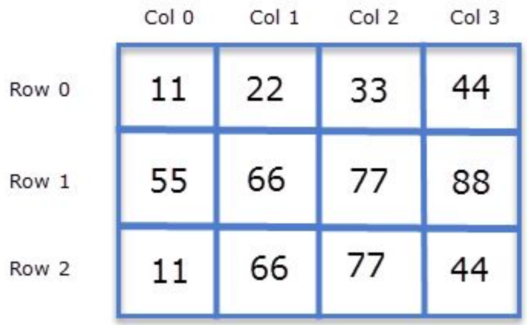
```
arr[0] = *(arr +0) or *(ptr +0)  // Array indexing is actually
arr[1] = *(arr +1) or *(ptr +1)  //dereferencing memory location
arr[i] = *(arr +i) or *(ptr +i)  //with pointer addition
```

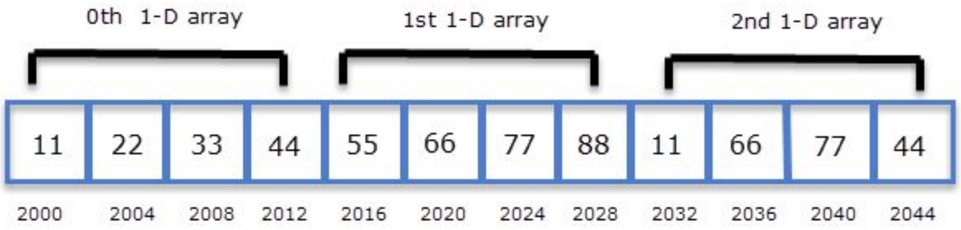
Arrays As a Pointers

```
#include <stdio.h>
int main()
{
 int num[ ] = { 24, 34, 12, 44, 56, 17 } ;
 int i, *j ;
 j = num;
 for ( i = 0 ; i <= 5 ; i++ )
 {
 printf ( "\naddress = %u ", j ) ;
 printf ( "element = %d", *j ) ;
 j++ ; /* increment pointer to point to next location */
 }
}
```

2D Array

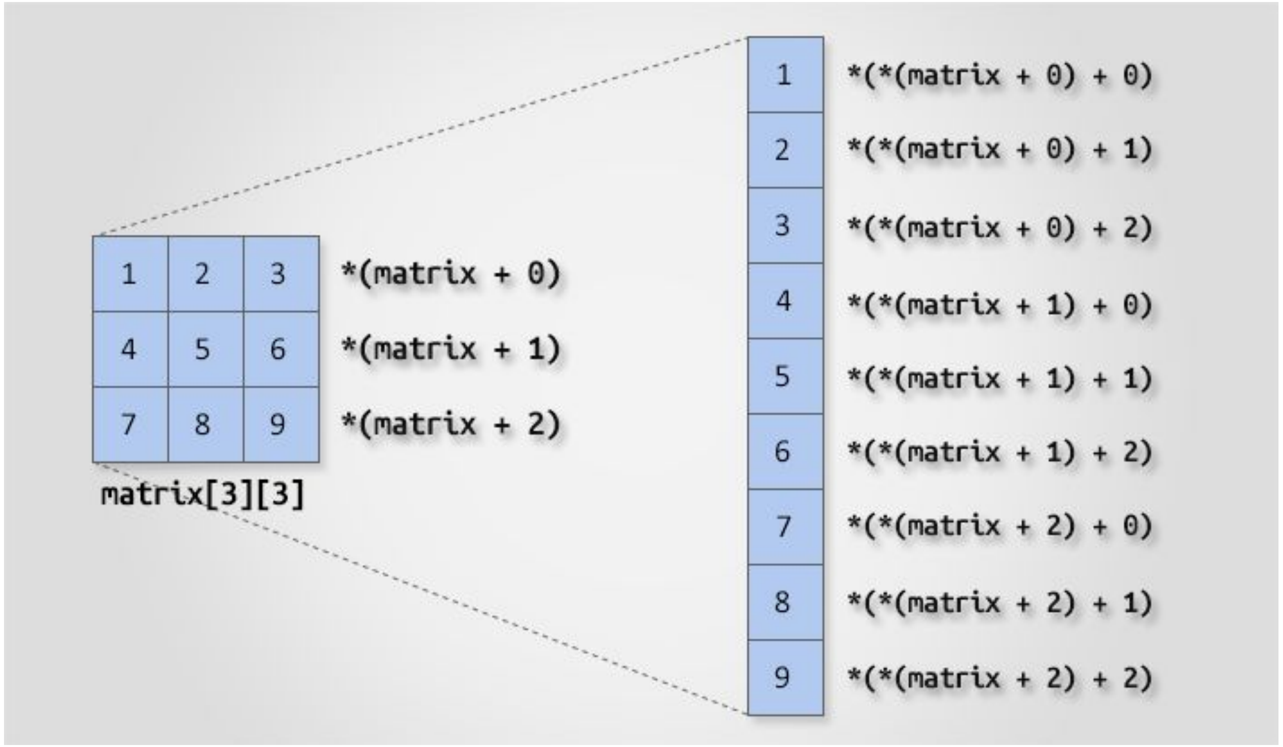






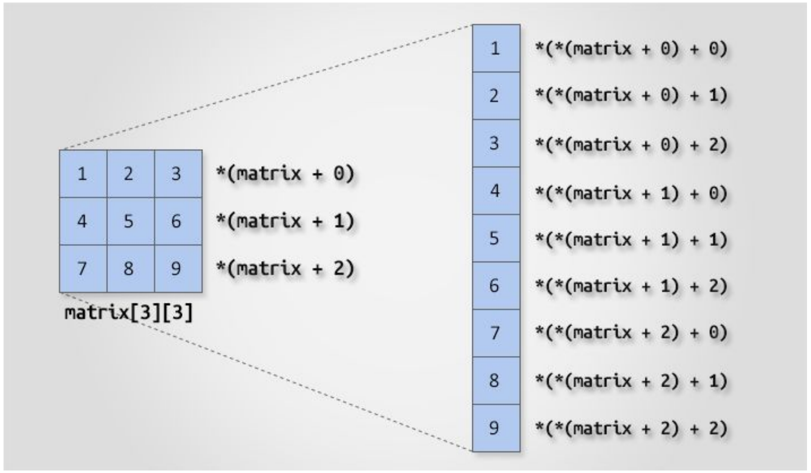
https://overiq.com/c-programming-101/pointers-and-2-d-arrays/

2D Arrays As a Pointers

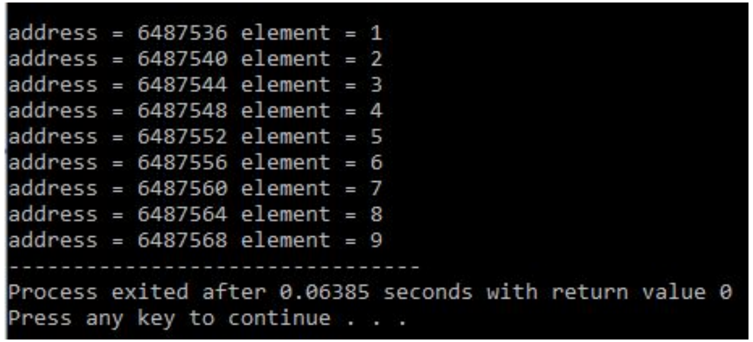


Home Assignment

Make a c program to print 2D matrix using pointers.



2D Arrays As a Pointers



```
address = 6487536 element = 1
address = 6487540 element = 2
address = 6487544 element = 3
address = 6487548 element = 4
address = 6487552 element = 5
address = 6487556 element = 6
address = 6487560 element = 7
address = 6487564 element = 8
address = 6487568 element = 9
--------------------------------
Process exited after 0.06385 seconds with return value 0
Press any key to continue . . .
```

```
#include <stdio.h>
int main()
{
 int num[3][3] = { {1,2,3},{4,5,6},{7,8,9}} ;
 int i, j;
 for ( i = 0 ; i < 3 ; i++ )
 {
      for(j=0; j<3; j++)
      {
           printf ( "\naddress = %u ",  *(num + i)+j ) ;
           printf ( "element = %d", *(*(num + i)+j) ) ;
      }
 }
}
```

Arrays Practice Problem

Write a program that calculates the sum of all the elements in 1D and 2D array using pointers.

Write a program that finds the highest number in a float type array of 20 elements using pointer.

Strings As a Pointers

```
char name[ ] = "FAST" ;
char *ptr ;
ptr = name ; /* store base address of string */
while ( *ptr != `\0' )
{
    printf ( "%c", *ptr ) ;
    ptr++ ;
}
```

Strings As a Pointers

```
char str[ ] = "Hello" ;
char *p = "Hello" ;
```

- We cannot assign a string to another, whereas, we can assign a char pointer to another char pointer.

- Once a string has been defined it cannot be initialized to another set of characters. Unlike strings, such an operation is perfectly valid with char pointers.

Strings As a Pointers

```
char str1[ ] = "Hello" ;
char str2[10] ;
char *s = "Good Morning" ;
char *q ;
str2 = str1 ; /* error */
q = s ; /* works */
-----------------------------

char str1[ ] = "Hello" ;
char *p = "Hello" ;
str1 = "Bye" ; /* error */
p = "Bye" ; /* works */
```