

Problem Solving Through Programming in C

Tutorial Session 6

Prof. Anupam Basu
Dept. of Computer Science & Engg.
IIT Kharagpur

Siddhant Mohapatra
PMRF Scholar
IIT Madras

Arrays

Q. In C, the placement of elements of a 2D array is:

- a) Row-wise
- b) Column-wise
- c) Diagonal-wise
- d) Bottom to top wise

C - row major

Fortran - column major

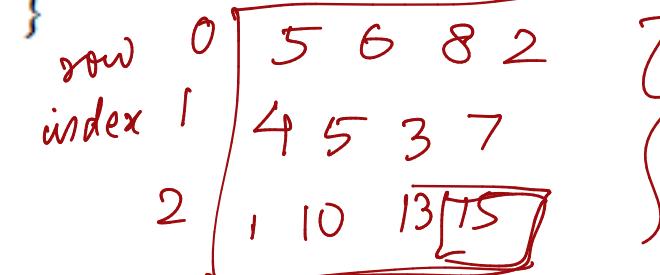
Python - row column major



Q. What will be the output?

```
#include <stdio.h>
int main()
{
    int disp[3][4] = {{5, 6, 8, 2}, {4, 5, 3, 7}, {1, 10, 13, 15}};
    printf("%d\n", disp[2][3]);
    return 0;
}
```

0 1 2 3 column index



12x1

15

2D of arr.

3x4

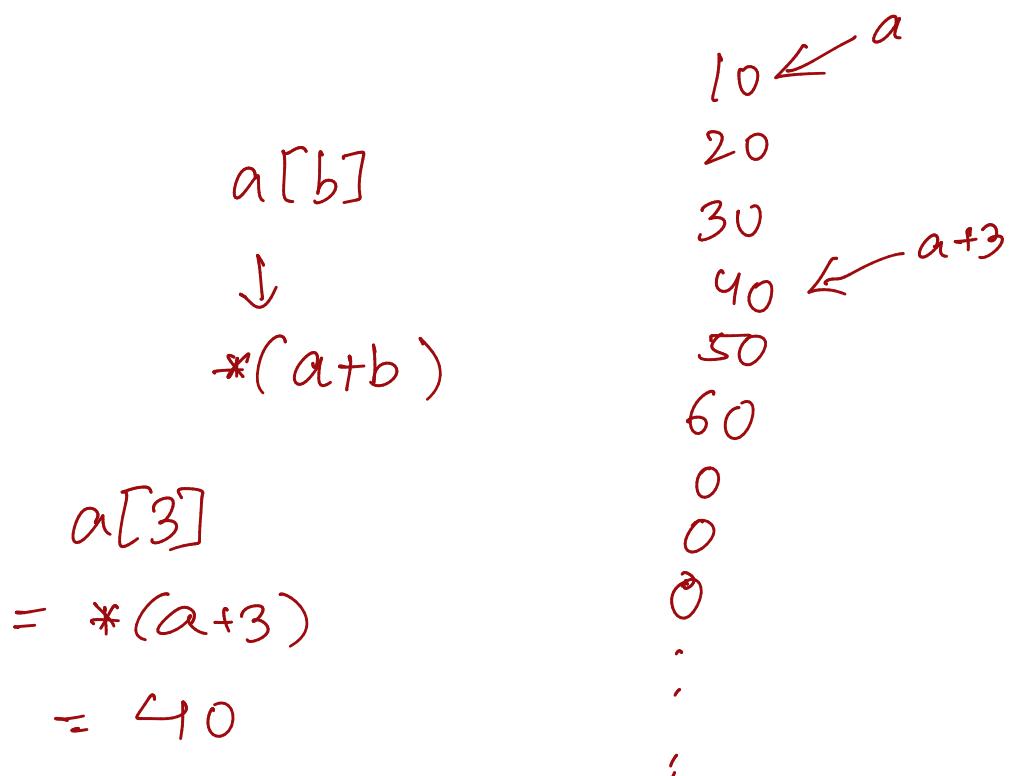
Arrays

Q. What will be printed after execution of the following code?

```
#include<stdio.h>
int main()
{
    0 1 2 3 4 5
    int a[20] = {10, 20, 30, 40, 50, 60}; 0 0 0 0 . .
    printf("%d", 3[a]);
    return 0;
}
```

Ans

$$\begin{aligned}
 & 3[a] \\
 &= *(3+a) \\
 &= *(a+3) \\
 &= a[3]
 \end{aligned}$$



I-LOVE-C\n
in
null
character

Arrays and strings

strcpy (dest, source)
strcat ↪

sizeof - returns no. of bytes occupied by any variable.

Q. What will be the output?

```
#include<stdio.h>
#include<string.h>
int main()
{
    char str1[20] = "Programming", str2[20] = "Language";
    printf("%s", strcpy(str2, strcat(str1, str2)));
    return 0;
}
```

format for printing string data

str1 = "ProgrammingLanguage"

- a) Programming
- b) Language
- c) ProgrammingLanguage
- d) LanguageProgramming

Q. What will be the output?

```
#include <stdio.h>
int main()
{
    char str1[] = "I-LOVE-C";
    char str2[] = {'I', 'L', 'O', 'V', 'E', ' ', 'C'};
    int n1 = sizeof(str1)/sizeof(str1[0]);
    int n2 = sizeof(str2)/sizeof(str2[0]);
    printf("n1=%d, n2=%d", n1, n2);
    return 0;
}
```

n2 = 8/1 = 8

n1 = 9/1 = 9

- a) n1=8, n2=8
- b) n1=9, n2=9
- c) n1=8, n2=9
- d) n1=9, n2=8

Arrays and strings

$\text{strcmp}(\text{str1}, \text{str2})$
 = 0 equal
 $\neq 0$ not equal (\neq)

Q. What will be the output of the program?

```
#include<stdio.h>
int main()
{
  char str[] = "Array\0String";
  printf("%s", str);
  return 0;
}
```

"null character"

- a) Array
- b) String
- c) Array String
- d) Array\0String
- e) Compilation error

Q. What is the output of the code snippet?

```
char str1[] = "abcd", str2[] = "abCd";
char str4[] = "ab", str5[] = "CD";

int r = strcmp(str1, str2);
printf("%d, ", r);
r = strcmp(strcat(str4, str5), str2); abfd abfd 68 - 100 = -32
printf("%d\n", r);
```

- a) 1, 0
- b) 0, 0
- c) 0, 32
- d) 32, 32
- e) 32, -32

$$\begin{array}{ccc}
 \text{str1} & \text{abcd} \\
 \text{str2} & \underline{\text{abCd}} \\
 & \underline{\quad\quad\quad} \\
 & \neq
 \end{array}
 \quad
 \begin{array}{l}
 99 - 67 = \underline{\underline{32}}
 \end{array}$$

`fgets(str, buffer size)`

assignment \0
 0 1 2 3 4 5 6 7 8 9 \0
 $i=10$ loop exits
 $i=9$ \0 assignment

$i=8$ \0 tsignmena
 $i=7$ \0 tnigmessa
 $i=6$ \0 tnegnmissa $i=5$
\0tnemngissa $i=4$
\0tnevnmissa $i=3$

Arrays and strings

Q. char str1[20], str2[20];
 printf("Enter a string twice: ");
 scanf("%s", str1); // Print 1
 gets(str2); // Print 2
 ↪ dangerous function (possibility of buffer overflow)

Is there any difference between Print 1 and Print 2?

str1 \Rightarrow Programming

str2 \Rightarrow Programming in C

printf("%s", str)

puts(str)

Q. What will be the output?

```
#include<stdio.h>
#include<string.h>
int main()
{
    char p[] = "assignment";
    char t;
    int i, j;
    for(i=0, j=strlen(p); i<j; i++)
    {
        j = 10; ~
        t = p[i];
        p[i] = p[j-i];
        p[j-i] = t;
    }
    printf("%s", p);
    return 0;
}
```

length of string (except '\0')
 0\tnigmessa $i=2$
\0tsignmena $i=1$
\0ssignment $i=0$

- a) assignment
- b) tnemngissa
- c) No output
- d) tttttttttt

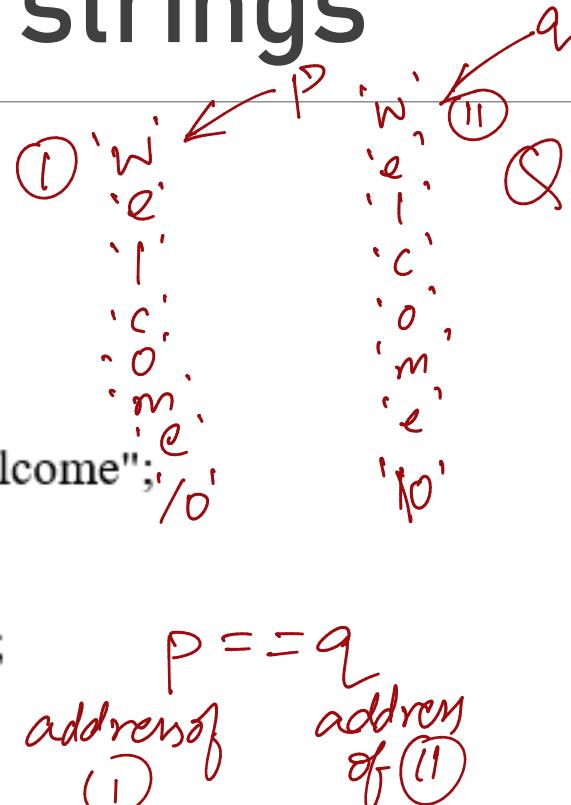
`printf ("%s", p)` → address of 1st element.

$$\begin{aligned} *p &= *(p+0) = p[0] \\ *q &= *(q+0) = q[0] \end{aligned}$$

Arrays and strings

Q. What will be the output?

```
#include<stdio.h>
#include<string.h>
int main()
{
    char p[] = "welcome", q[]="welcome";
    if(p==q)
    {
        printf("Two strings are equal");
    }
    return 0;
}
```



- a) Two strings are equal
- b) Two strings are not equal
- c) No output
- d) Compilation error

$\text{if } (*p == *q) \text{ true}$

Q. What will be the output?

```
# include <stdio.h>
int main()
{
    int i, m=1, num[6] = {1,2,3,4,5,6};
    for(i=0; i<=5; i++)
        m=m*num[i];
    printf(" %d", m);
    return 0;
}
```

720

} product of all elements
of the array.

* arr → value of index 0 element
arr[0]

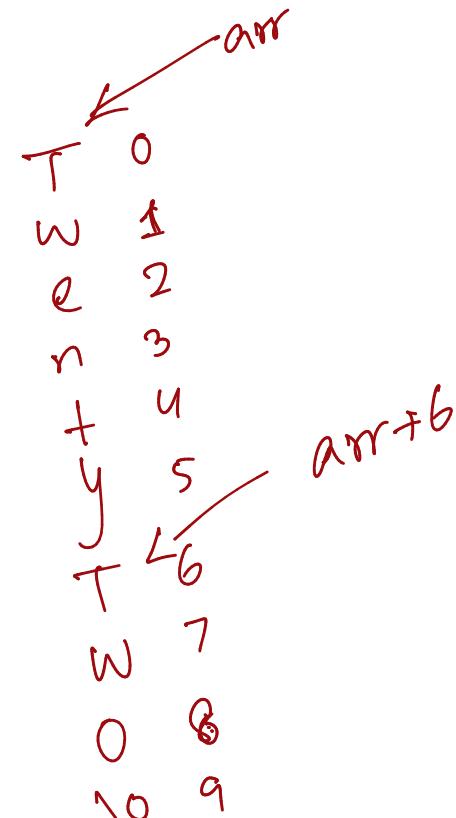
arr → address of index 0 element

Arrays and strings

Q. What would you put in place of "****" to print "Two"?

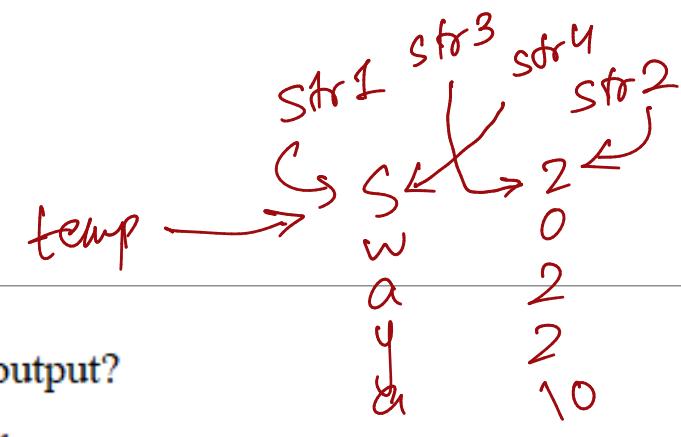
```
#include <stdio.h>
int main()
{
    char arr[] = "TwentyTwo";
    printf("%s", ****);
    return 0;
}
```

- a) arr
- b) arr+5
- c) arr+6
- d) Not possible



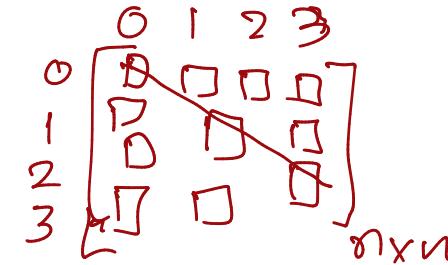
Q. What will be the output?

```
#include <stdio.h>
void swap(char *str1, char *str2)
{
    char *temp = str1; formal parameters
    str1 = str2;
    str2 = temp; actual parameters
}
int main()
{
    char *str1 = "Swayam";
    char *str2 = "2022";
    swap(str1, str2);
    printf("str1 is %s, str2 is %s", str1, str2);
    return 0;
}
```



Arrays and strings

$$\begin{matrix} \text{no. of elements} \\ \text{above diagonal} \\ \text{or to} \end{matrix} = \frac{(n^2 - n)}{2}$$



$$a[0][1] = a[1][0]$$

Write a C program to check if a 2D matrix is symmetric.

$m \times n$
inputs from user

if ($m \neq n$) not symmetric

else

{ user input all the elements $m \times n$ arr

count = 0;

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

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

 if (arr[i][j] == arr[j][i]) count++;

if (count == $((n^2 - n)/2)$)

symmetric

else not symmetric

}

Arrays and strings

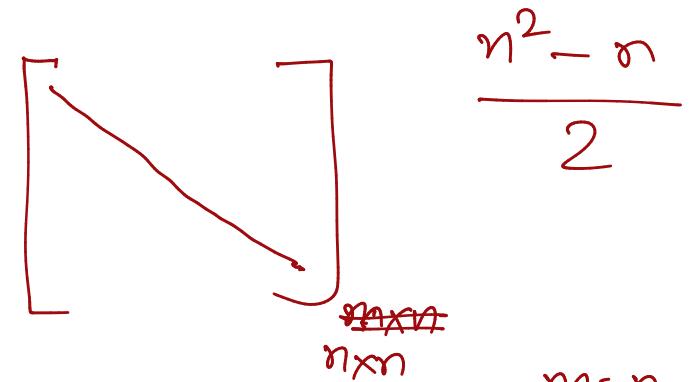
Write a C program to check if a 2D matrix is symmetric.

```
#include <stdio.h>
int main(){
int m, n, i, j;
printf("Enter matrix dimensions:\n");
scanf("%d %d", &m, &n);
if(m!=n) printf("Matrix is not symmetric");
else{
    int mat[m][n], count=0;
    printf("Enter the elements of the matrix:");
    for(i=0;i<m;i++)
    for(j=0;j<n;j++) scanf("%d", &mat[m][n]);

    for(i=0;i<m;i++)
    for(j=i+1;j<n;j++) {
        if(mat[i][j] == mat[j][i]) count++;
    }
}
```

$\Rightarrow \frac{(n^2 - n)}{2}$

```
    if(count==n(m*n-n)/2)
        printf("Matrix is symmetric");
    else
        printf("Matrix is not symmetric");
    }
return 0;
}
```



no. of diag terms = n

$$m = n$$