Chapter 7

Arrays & Strings

Er. Shiva K. Shrestha (HoD)

Department of Computer Engineering,

Khwopa College of Engineering

Arrays

- 1D Arrays
- Declaration of 1D Arrays
- Global Declaration
- Local Declaration
- Initialization of Arrays
- Compile Time Initialization
- Run Time Initialization
- Accessing the Array Elements
  - Assigning values to array elements,
  - Displaying the array elements

- 2D Arrays
- Elements of 2D Arrays
- Initialization of 2D Arrays
- Passing Array Element to a Function
- Passing Array to a Function
- Passing 2D Array to a Function
- Returning Array Element from a Function
- Returning Array from a Function

One Dimensional Array

- Declaration of One-Dimensional Arrays
    - [storage_class] data_type variable_name[size];
- Global Declaration

    char string[50]; /* Global Declaration */

    void main(){

    ....

    }
- Local Declaration

    void main(){

    int age[] = {21,22,19,20}; /* Local Declaration & Initialization */

    ....

    }

Initialization of Arrays

1. Compile Time Initialization
   ▶ int number[3]={0,0,0};
2. Run Time Initialization
   ▶ int x[3];
   ▶ scanf("%d%d%d",&x[0],&x[1],&x[2]);

Accessing the Array Elements

i. Assigning values to array elements,

✓   int a[5], b[3];

✓   a=0; /* wrong*/

✓   b=a; /* wrong*/

✓   if(a<b); /* wrong*/

✓   {…} /* wrong*/

Another Example
float marks[3];
marks[0] = 90.05;
marks[1] = 88.77;
marks[2] = 78.12;
marks[3] = 90.77; /* Invalid */

Compile Time Initialization

```c
char name[] = "Ram";
/*
Same as
char name[] = {'R','a','m','\0'};
*/
```

```
C7_1.C                    ×
 1 #include<stdio.h>
 2 #include<conio.h>
 3 int main(){
 4     char name[] = {'R','a','m','\0'};
 5     char name2[] = "Khwopa";
 6     printf("%s", name);
 7     printf("\n%s", name2);
 8     getch();
 9     return 0;
10 }
```

```
C:\Users\ErSKS\
Ram
Khwopa
```

Run Time Initialization

Program to count negative, positive & zeros from entered numbers

```c
C7_2.C                    ×
1  #include<stdio.h>
2  #include<conio.h>
3  int main(){
4      int a[50],n,count_n=0,count_z=0,count_p=0,i;
5      printf("Enter size of array: ");
6      scanf("%d",&n);
7      printf("Enter the elements of the array\n");
8      for(i=0;i<n;i++){
9          scanf("%d",&a[i]);
10     }
11     for(i=0;i<n;i++){
12         if(a[i]<0){
13             count_n++;
14         }else if(a[i]>0){
15             count_p++;
16         }else{
17             count_z++;
18         }
19     }
20     printf("Negative = %d\n",count_n);
21     printf("Zero = %d\n",count_z);
22     printf("Positive = %d.\n",count_p);
23     getch();
24     return 0;
25 }
```

```
C:\Users\ErSKS\Google Drive (c.khwopa@gmail.com)
Enter size of array: 7
Enter the elements of the array
1 0 3 -8 5 6 -2
Negative = 2
Zero = 1
Positive = 4.
```

2D Array

[storage_class] data_type array_name[row_size][column_size];

Declaration of 2D Array

    int m[3][3];

Initialization of 2D Array

    int a[3][3]={{5,6,7},{4,3,8},{1,5,9}};

       is equivalent to

a[0][0]=5; a[0][1]=6; a[0][2]=7;

a[1][0]=4; a[1][1]=3; a[1][2]=8;

a[2][0]=1; a[2][1]=5; a[2][2]=9;

Er. Shiva K.
2019-12-24

a[0][0]	a[0][1]	a[0][2]
5	6	7
a[1][0]	a[1][1]	a[1][2]
4	3	8
a[2][0]	a[2][1]	a[2][2]
1	5	9

Accessing 2D Array Element

int b[4][4] = {

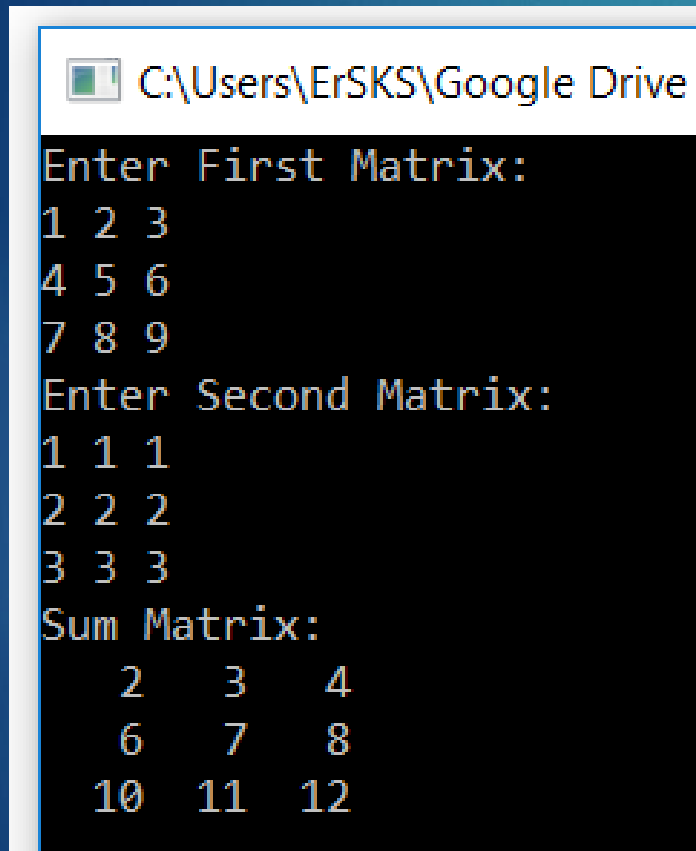{5,6,7,8},

{1,2,3,4},

{8,5,2,0},

{7,5,2,0}

};

b[2][1] = ?

b[0][3] = ?

b[3][3] = ?

b[1][1] = ?

5	6	7	8
1	2	3	4
8	5	2	0
7	5	3	4

WAP to read m*n matrices and display their sum.

```
Enter First Matrix:
1 2 3
4 5 6
7 8 9
Enter Second Matrix:
1 1 1
2 2 2
3 3 3
Sum Matrix:
   2   3   4
   6   7   8
  10  11  12
```
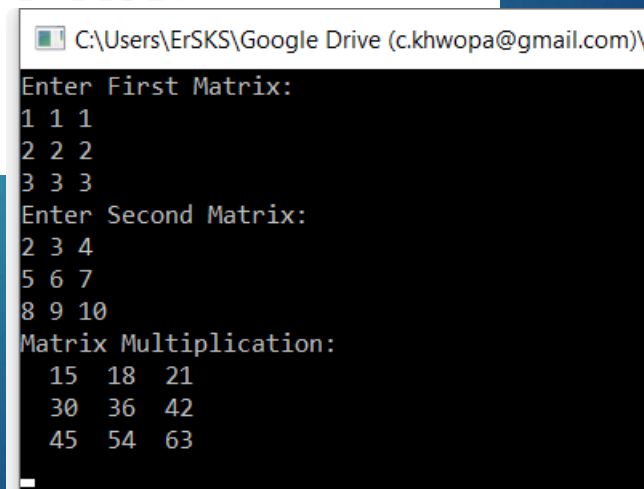
C:\Users\ErSKS\Google Drive

C7_3.C

```c
#include<stdio.h>
#include<conio.h>
int main(){
    int m=3, n=3;
    int a[m][n], b[m][n], sum[m][n], i, j;
    printf("Enter First Matrix:\n");
    for (i=0; i<m; i++){
        for (j=0; j<n; j++){
            scanf("%d", &a[i][j]);
        }
    }
    printf("Enter Second Matrix:\n");
    for (i=0; i<m; i++){
        for (j=0; j<n; j++){
            scanf("%d", &b[i][j]);
        }
    }
    printf("Sum Matrix:\n");
    for (i=0; i<m; i++){
        for (j=0; j<n; j++){
            sum[i][j] = a[i][j] + b[i][j];
            printf("%4d", sum[i][j]);
        }
        printf("\n");
    }
    getch();
    return 0;
}
```

Assignment

▶ WAP to read two m*n matrices and display their product.

```
18        printf("Matrix Multiplication:\n");
19        for (i=0; i<m; i++){
20            for (j=0; j<n; j++){
21                mult[i][j] = 0;
22                for (k = 0; k < n; ++k) {
23                    mult[i][j] += a[i][k] * b[k][j];
24                }
25                printf("%4d", mult[i][j]);
26            }
27            printf("\n");
28        }
```

```
C:\Users\ErSKS\Google Drive (c.khwopa@gmail.com)\
Enter First Matrix:
1 1 1
2 2 2
3 3 3
Enter Second Matrix:
2 3 4
5 6 7
8 9 10
Matrix Multiplication:
  15  18  21
  30  36  42
  45  54  63
```

m*n & p*q
Matrix Multiplication

```c
#include<stdio.h>
#include<conio.h>
int main(){
    int m=10, n=10, p=10, q=10;
    int a[m][n], b[m][n], mult[m][n], i, j, k;
    re: printf("Enter rows & column for 1st matrix: ");
    scanf("%d %d", &m, &n);
    printf("Enter rows & column for 2nd matrix: ");
    scanf("%d %d", &p, &q);
    if(n != p){
        printf("Error! Enter rows & columns again.\n");
        goto re;
    }
    printf("Enter First Matrix:\n");
    for(i=0; i<m; i++){
        for(j=0; j<n; j++){
            scanf("%d", &a[i][j]);
        }
    }
    printf("Enter Second Matrix:\n");
    for (i=0; i<p; i++){
        for (j=0; j<q; j++){
            scanf("%d", &b[i][j]);
        }
    }
    printf("Matrix Multiplication:\n");
    for (i=0; i<m; i++){
        for (j=0; j<q; j++){
            mult[i][j] = 0;
            for (k = 0; k < n; k++) {
                mult[i][j] += a[i][k] * b[k][j];
            }
            printf("%4d", mult[i][j]);
        }
        printf("\n");
    } getch(); return 0;
}
```
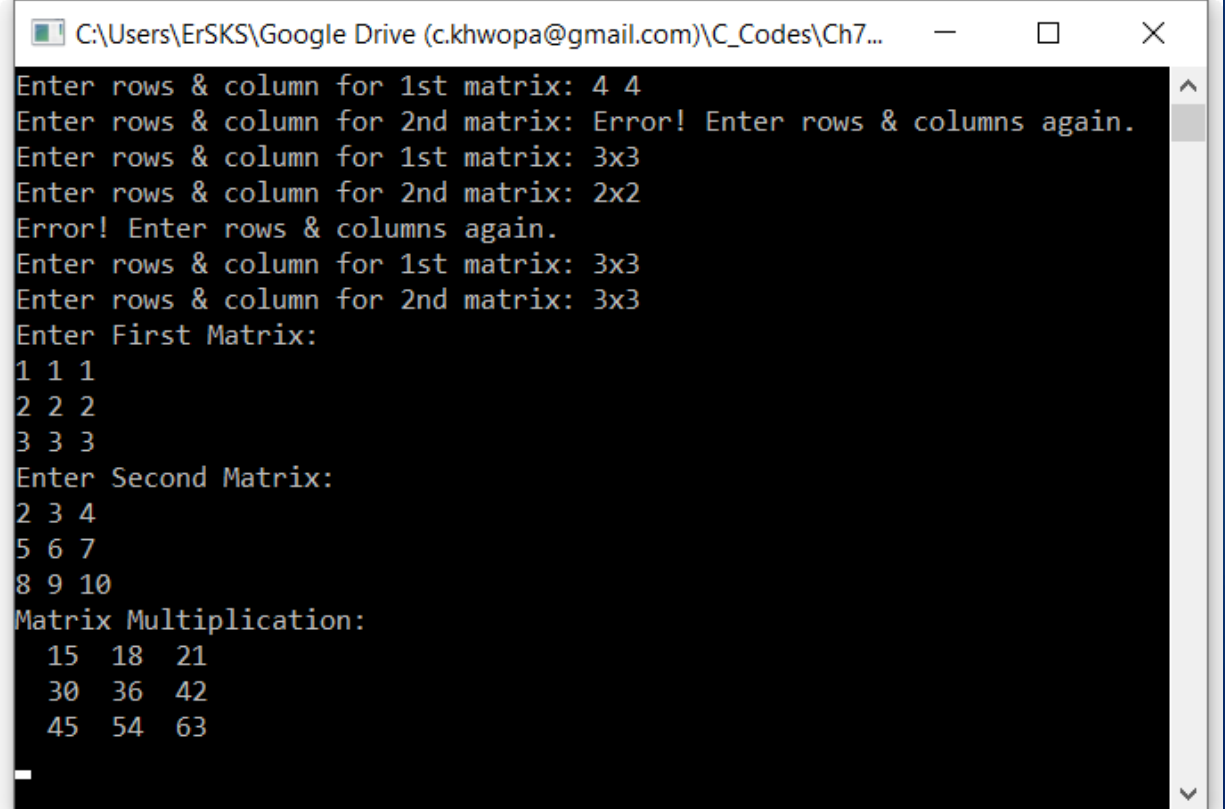
```
C:\Users\ErSKS\Google Drive (c.khwopa@gmail.c...    —

Error! Enter rows & columns again.
Enter rows & column for 1st matrix: 3 3
Enter rows & column for 2nd matrix: 3 3
Enter First Matrix:
1 1 1
2 2 2
3 3 3
Enter Second Matrix:
2 3 4
5 6 7
8 9 10
Matrix Multiplication:
   15   18   21
   30   36   42
   45   54   63
```

scanf ()

Sort Array Element

Er. Shiva K. Shrestha (HoD, Computer Department)
2019-12-24

```c
C7_4_v2.C
1  #include<stdio.h>
2  #include<conio.h>
3  int main(){
4      int a[6]={7,-2,10,9,1,0};
5      int i, j, temp;
6      clrscr();
7      for(i=0;i<5;i++){
8          for(j=0;j<5;j++){
9              if(a[j]>a[j+1]){
10                 temp=a[j];
11                 a[j]=a[j+1];
12                 a[j+1]=temp;
13             }
14         }
15     }
16     printf("Array Elements after Sort: ");
17     for(i=0;i<6;i++){
18         printf("%4d",a[i]);
19     }
20     getch();
21     return 0;
22 }
```

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Pr

Array Elements after Sort: -2   0   1   7   9   10

Passing Array Element to a Function

```
C7_5.C                    ×
1   #include<stdio.h>
2   #include<conio.h>
3   void findSum(int,int);
4   int main(){
5       int a[2]={8,9};
6       findSum(a[0],a[1]);
7       getch();
8       return 0;
9   }
10  void findSum(int a,int b){
11      printf("Sum = %d", (a+b));
12  }
```

C:\Users\ErSKS\Google Drive
Sum = 17

Passing Array to a Function

```c
C7_6.C
 1  #include<stdio.h>
 2  #include<conio.h>
 3  void findSum(int a[]);
 4  int main(){
 5      int a[3]={8,9,10};
 6      findSum(a);
 7      getch();
 8      return 0;
 9  }
10  void findSum(int x[]){
11      int i, sum=0;
12      for(i=0;i<3;i++){
13          printf("x[%d] = %d\n", i, x[i]);
14          sum += x[i];
15      }
16      printf("\nSum = %d", sum);
17  }
```

```
C:\Users\ErSKS\
x[0] = 8
x[1] = 9
x[2] = 10

Sum = 27
```

Task

- ▶ WAP to add n numbers entered by user. Use the concept of array & function.

Passing 2-D Array to a Function

```
C7_7.C                               ×
 1  #include<stdio.h>
 2  #include<conio.h>
 3  void fxn(int a[3][3]);
 4  int main(){
 5      int i, a[3][3]={{1,2,3},{4,5,6},{7,8,9}};
 6      fxn(a);
 7      getch();
 8      return 0;
 9  }
10  void fxn(int x[3][3]){
11      int i,j;
12      for(i=0;i<3;i++){
13          for(j=0;j<3;j++){
14              printf("x[%d][%d]=%d\t",i,j,x[i][j]);
15          }
16          printf("\n");
17      }
18  }
```

C:\Users\ErSKS\Google Drive (c.khwopa@gmail.com)

```
x[0][0]=1        x[0][1]=2        x[0][2]=3
x[1][0]=4        x[1][1]=5        x[1][2]=6
x[2][0]=7        x[2][1]=8        x[2][2]=9
```

HoD, Computer Department)

```c
/* Returning Array Element from a function */
#include<stdio.h>
int findLargest(int a[]);
int main(){
    int n[] = {5,10,2,-14,15};
    printf("Largest Number = %d",findLargest(n)); return 0;
}
int findLargest(int x[]){
    int i;
    // x[0] is used for holding max value
    for (i=1;i<5;i++){
        if(x[0]<x[i]){
            x[0] = x[i];
        }
    }
    return x[0];
}
```

C:\Users\ErSKS\Google Drive

Largest Number = 15

Returning Array from a Fxn

```
C7_9.C                          ×
 1   /* Program to generate prime numbers */
 2   #include<stdio.h>
 3   #include<conio.h>
 4   int generatePrime(int *, int);
 5   int isPrime(int);
 6   int main(){
 7       int i, prime[100], status=0, n_prime=20;
 8       status = generatePrime(&prime[0],n_prime);
 9       printf("Prime Numbers:\n");
10       for (i = 0; i<n_prime && status == 1; ++i){
11           printf("%d\t", prime[i]);
12       }
13       getch();
14   }
15   int generatePrime(int *el, int n){
16       int i, count=0;
17       for (i=2; ; i++){
18           if(isPrime(i)==1){
19               *el = i;
20               el++;
21               count++;
22               if (count>=n){
23                   break;
24               }
25           }
26       }
27       return 1;
28   }
```

```
29   int isPrime(int n) {
30       int i, countFactor = 0, result = 1;
31       if (n < 2) {
32           return 0;
33       }
34
35       for (i = 1; i < n && result == 1; i++) {
36           if (n % i == 0) {
37               countFactor++;
38           }
39           if (countFactor > 1) {
40               result = 0;
41           }
42       }
43       return result;
44   }
```

```
C:\Users\ErSKS\Google Drive (c.khwopa@gmail.com)\C_...   —   □

2       3       5       7       11      13      17      19
23      29      31      37      41      43      47
53      59      61      67      71
```

Multidimensional Arrays

General Syntax:

▶ [storage_class] data_type array_name [s1][s2][s3]…[sn];

Task

- WAP to read 3*3 matrix & display it to screen.

- Give an example program to add two matrices & store the results in the 3rd matrix.

Strings

- Declaration and Initializing String Variable
- Reading Strings from Terminal
- Reading a Line of Text
- String Handling Functions
- Passing String to a Function

Declaration and Initializing String Variable

- *Syntax:* char string_name[size];

K	H	W	O	P	A	\0	\0	\0	\0

- char college[10] = "KHWOPA";
- char college[4] = "KHWOPA";   /* Illegal */

Reading a Line of Text

Using scanf() Function

▶ char address[20];

▶ scanf("%s", adress);

```
Enter text:
Libali, Bhaktapur
Computer
Khwopa

word1 = Libali,
word2 = Bhaktapur
word3 = Computer
word4 = Khwopa
```

```c
C7_10.C

1   #include<stdio.h>
2   #include<conio.h>
3   int main(){
4       char word1[40], word2[40], word3[40], word4[40];
5       printf("Enter text: \n");
6       scanf("%s%s", word1, word2);
7       scanf("%s", word3);
8       scanf("%s", word4);
9       printf("\nword1 = %s\nword2 = %s\n", word1, word2);
10      printf("word3 = %s\nword4 = %s\n", word3, word4);
11      getch();
12      return 0;
13  }
```

HoD, Computer Department)

Reading a Line of Text

```
C7_11.C                          ×
1   #include<stdio.h>
2   #include<conio.h>
3   int main(){
4       char line[100];
5       printf("Enter Your Sentence:\n");
6       scanf("%[^\n]",line);
7       printf("%s", line);
8       getch();
9       return 0;
10  }
```

C:\Users\ErSKS\Google Drive (c.khwopa

```
Enter Your Sentence:
Dedicated to People & Country
Dedicated to People & Country
```

Er. Shiva K. Shrestha (HoD, Computer Department)
2019-12-24

ASCII Value & Character

```
C7_12.C                    ×
1  #include<stdio.h>
2  #include<conio.h>
3  int main(){
4      char c;
5      printf("\n\n");
6      for(c=65; c<=122; c++){
7          if( c > 90  &&  c < 97 ){
8              continue;
9          }
10         if(c==97){
11             printf("\n\n");
12         }
13         printf("|%4d - %c ", c, c);
14     }
15     printf("|\n");
16     getch();
17     return 0;
18 }
```

```
C:\Users\ErSKS\Google Drive (c.khwopa@gmail.com...        —    □    ×

|  65 - A |  66 - B |  67 - C |  68 - D |  69 - E |  70 - F
|  71 - G |  72 - H |  73 - I |  74 - J |  75 - K |  76 - L
|  77 - M |  78 - N |  79 - O |  80 - P |  81 - Q |  82 - R
|  83 - S |  84 - T |  85 - U |  86 - V |  87 - W |  88 - X
|  89 - Y |  90 - Z |

|  97 - a |  98 - b |  99 - c | 100 - d | 101 - e | 102 - f
| 103 - g | 104 - h | 105 - i | 106 - j | 107 - k | 108 - l
| 109 - m | 110 - n | 111 - o | 112 - p | 113 - q | 114 - r
| 115 - s | 116 - t | 117 - u | 118 - v | 119 - w | 120 - x
| 121 - y | 122 - z |
```

String Handling Functions

i. strlen() – return length of a string

ii. strcpy() – strcpy(destination_string, source_string)

iii. strcat() – concatenates two strings

iv. strcmp() – compares two strings

    i. Returns 0 if both are same

    ii. Returns less than 0 if 1st string is less than 2nd string

    iii. Returns greater than 0 if 1st string is greater than 2nd string

v. strrev() – reverse given string

String Concatenation

```
C7_13.C                    ×
1  #include<stdio.h>
2  #include<conio.h>
3  #include<string.h>
4▼ int main(){
5      char s1[]="Khwopa", s2[]="Computer";
6      printf("%s\n", strcat(s1,s2));
7      getch();
8      return 0;
9  }
```
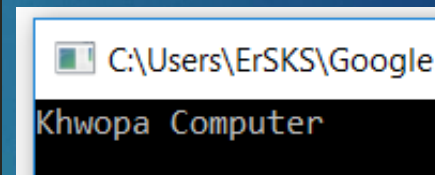
```
C:\Users\ErSKS\...
KhwopaComputer
_
```

String Concatenation without using string.h

29

2019-12-24 Er. Shiva K. Shrestha (HoD, Computer Department)

```c
#include<stdio.h>
#include<conio.h>
████████████████
int main(){
    int  i, j;
    char  s1[10]="Khwopa", s2[10]="Computer";
    char  str[20];
    /* Copy s1 into str */
    for(i=0; s1[i] != '\0'; i++ ){
      str[i] = s1[i];
    }
    /* End s1 with a space */
    str[i] = ' ' ;
    /* Copy s2 into str */
    for( j = 0 ; s2[j] != '\0' ; j++ ){
      str[i+j+1] = s2[j];
    }
    /* End str with a null character */
    str[i+j+2] = '\0';

    printf("%s\n", str);
    getch();
    return 0;
}
```

C:\Users\ErSKS\Google

Khwopa Computer

Illustrations of String-Handling Functions

C7_15.C

```c
#include<stdio.h>
#include<conio.h>
#include<string.h>
int main(){
    char s1[20], s2[20], s3[20];
    int x, l1, l2, l3;
    printf("Enter Two Strings: ");
    scanf("%s%s", s1, s2);
    /* comparing s1 and s2 */
    x = strcmp(s1, s2);
    if(x != 0){
        printf("Strings are not equal.\nMismatch Difference = %d",x);
        strcat(s1, s2);/*joining s1 & s2, assign result to s1 */
    }
    else{
        printf("Strings are equal.");
    }
    strcpy(s3, s1); /* copying s1 to s3 */
    /* Finding length of strings */
    l1 = strlen(s1);
    l2 = strlen(s2);
    l3 = strlen(s3);
    printf("\ns1 = %s\t length = %d characters\n", s1, l1);
    printf("s2 = %s\t\t length = %d characters\n", s2, l2);
    printf("s3 = %s\t length = %d characters\n", s3, l3);
    getch();
    return 0;
}
```

C:\Users\ErSKS\Google Drive (c.khwopa@gmail.com)\C_Codes\

```
Enter Two Strings: Prashisha Prabesh
Strings are not equal.
Mismatch Difference = 1
s1 = PrashishaPrabesh      length = 16 characters
s2 = Prabesh               length = 7 characters
s3 = PrashishaPrabesh      length = 16 characters
```

Computer Department)

Passing String to a Function

```
C7_16.C

1   #include<stdio.h>
2   #include<conio.h>
3   void function(char a[]){
4       printf("Passed String: %s", a);
5   }
6
7   int main(){
8       char text[] = {"Engineering"};
9       function(text);
10      getch();
11      return 0;
12  }
```

C:\Users\ErSKS\Google Drive (c.khwopa(

Passed String: Engineering

```
C7_3_MatMul_v2_2.C   sum_n_arr_fxn.C   C7_14.C   C7_17.C

1  #include<stdio.h>
2  void function(char a[3][11]){
3      int i; printf("Passed Strings:\n");
4      for (i=0; i<3; i++){
5          printf("String%d = %s\n",i+1,a[i]);
6      }
7  }
8  int main(){
9      char text[3][11] = {"Civil","Computer","Electrical"};
10     function(text);
11     return 0;
12 }
```

```
C:\Users\ErSKS\Google Drive (c.khwopa@gmail.co
Passed Strings:
String1 = Civil
String2 = Computer
String3 = Electrical
```

Q/A**?**

Thank You!

Er. Shiva K. Shrestha

computer.khwopa@gmail.com

Er. Shiva K. Shrestha (HoD, Computer Department)
2019-12-24