

GOVERNMENT ENGINEERING COLLEGE AJMER

B.Tech I Year II Sem (All section)

Subject: Computer Programming-II

Mid Term-I (2017-18)

Duration: 1 Hrs

MM.:20

Note: 1) Attempts 4 questions.

2) Q. No. 1 is compulsory

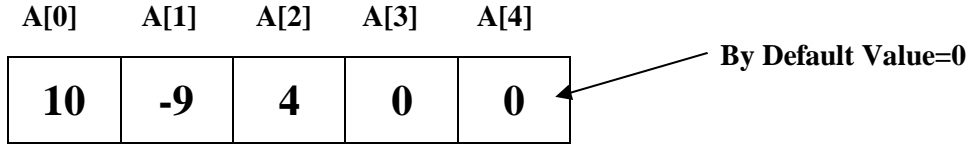
Q. No	CO	Cognitive Domain	Question	Marks
1	CO1	Analysing	<p>List down the output of following code with explanation:</p> <pre>I) void main() { int A[5]={10,-9,4}; printf(“%d,%d\n”, A[1], -2[A]); printf(“ %d, %d”, ++A[3], sizeof(A)); }</pre>	2.5
	CO2	Analysing	<pre>II) void main() { char A[25]=”Engineering”, B[10]=” College”, C[30]; strcpy(C, strcat(A, B)); printf(“%s\n%s\n%s”, A, B, C); }</pre>	2.5
2	CO1	Under-standing	Give the definition of Array? Explain 2-D array with suitable program.	5
3	CO1	Apply	WAP to find first largest, second largest & third largest element from given Array.	5
4	CO2	Apply	<p>WAP to print following output using character array (string).</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <pre>H H E H E L H E L L H E L L O H E L L H E L H E H</pre> </div>	5
5	CO2	Apply	WAP to check whether given String is palindrome or not. Example: “SARAS”	5

-:SOLUTION:-

Q.1 List down the output of following code with explanation:

```
I) void main() {
    int A[5]={10,-9,4};
    printf("%d,%d\n", A[1], -2[A]);
    printf(" %d, %d", ++A[3], sizeof(A));
}
```

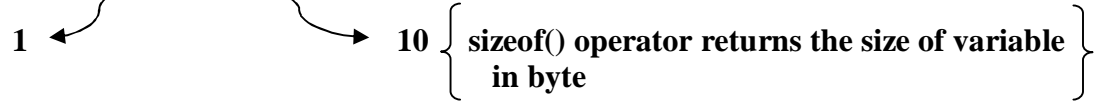
Solution:-



```
printf("%d,%d\n", A[1], -2[A]);
```



```
printf(" %d, %d", ++A[3], sizeof(A));
```



```
Output:
-9, -4
1, 10
```

```
II) void main() {
    char A[25]="Engineering", B[10]=" College", C[30];
    strcpy(C, strcat(A, B));
    printf("%s\n%s\n%s", A, B, C);
}
```

Solution:-

```
strcpy(C, strcat(A, B));
```

1. **strcat(A,B)** function concate/ join two strings.
2. String B (" College") is appends at end of String A ("Engineering").
3. Resultant string is A ("Engineering College")

1. **strcpy(C,A)** function copying content of string A ("Engineering College") into string C.
2. Resultant string is C ("Engineering College").

```
printf("%s\n%s\n%s", A, B, C);
```

Output:-

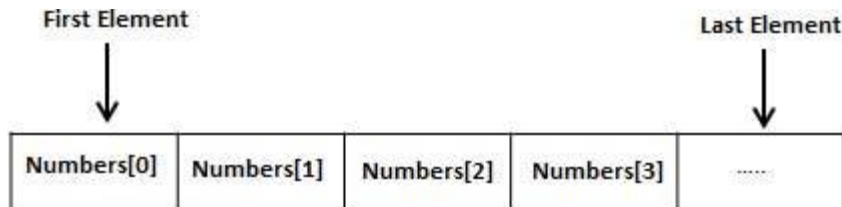
```
Engineering College
College
Engineering College
```

Q.2 Give the definition of Array? Explain 2-D array with suitable program.

Solution:-

Array

1. An array is collection of homogeneous or similar type element.
2. A specific element in an array is accessed by an index
3. All arrays consist of contiguous memory locations. The lowest address corresponds to the first element and the highest address to the last element.



4. The size and type of arrays cannot be changed after its declaration.
5. Index range is start from 0 through size-1

Declaration of Array:-

```
data_type array_name[array_size];
```

eg:- `int a[10];`
`float b[5];`

Arrays are of two types:

1. One-dimensional arrays
2. Multidimensional arrays

2-D array/matrix:

1. In C programming, we can create an array of arrays known as multidimensional array.
2. In 2-D array/matrix there are two dimensions, one for no. of rows and another for no. of columns.

Declaration of 2-D array

```
data_type array_name[size1][size2];
```

eg:- `int a[3][5];`
`float b[5][2];`

Representation of matrix:

`int x[3][4];`

	Column 1	Column 2	Column 3	Column 4
Row 1	<code>x[0][0]</code>	<code>x[0][1]</code>	<code>x[0][2]</code>	<code>x[0][3]</code>
Row 2	<code>x[1][0]</code>	<code>x[1][1]</code>	<code>x[1][2]</code>	<code>x[1][3]</code>
Row 3	<code>x[2][0]</code>	<code>x[2][1]</code>	<code>x[2][2]</code>	<code>x[2][3]</code>

Program example for scanning & printing elements of 2-D array/matrix:

```
#include<stdio.h>
#include<stdio.h>
void main()
{
    int A[3][2];
    int i, j;
    /*scanning of elements*/
    for(i=0; i<3; i++) {
        for(j=0; j<2; j++) {
            printf("Enter value for A[%d][%d]:", i, j);
            scanf("%d", &A[i][j]);
        }
    }
    //Displaying array elements
    printf("Two Dimensional array elements:\n");
    for(i=0; i<3; i++) {
        for(j=0; j<2; j++) {
            printf("%d\t", A[i][j]);
        }
        printf("\n");
    }
}
```

Output :

```
Enter value for A[0][0]:1
Enter value for A[0][1]:2
Enter value for A[1][0]:3
Enter value for A[1][1]:4
Enter value for A[2][0]:5
Enter value for A[2][1]:6
Two Dimensional array elements:
1    2
3    4
5    6
```

Q.3 WAP to find first largest, second largest & third largest element from given Array.

Solution:-

Method-I: without sorting

```
#include<stdio.h>
#include<stdio.h>
void main()
{
    int a[10], n;
    int large1, large2 ,large3, i;
    clrscr();
    printf("enter number of elements you want in array(max=10)");
    scanf("%d", &n);
    printf("enter elements");
    for (i = 0; i < n; i++) {
        scanf("%d", &a[i]);
    }
    large1 = a[0];
    for (i = 1; i < n; i++) {
        if (a[i] > large1) {
            large1 = a[i];
        }
    }
    large2 = a[0];
    for (i = 1; i < n; i++) {
        if (a[i] > large2 && a[i] < large1){
            large2 = a[i];
        }
    }
    large3 = a[0];
    for (i = 1; i < n; i++) {
        if (a[i] > large3 && a[i] < large2){
            large3 = a[i];
        }
    }
    printf("First largest number is %d\n ", large1);
    printf("Second largest number is %d\n ", large2);
    printf("Third largest number is %d\n ", large3);
    getch();
}
```

Method-II: without sorting but in single loop

```
#include<stdio.h>
#include<stdio.h>
void main()
{
    int a[10], n;
    int large1, large2 ,large3, i;
    clrscr();
    printf("enter number of elements you want in array(max=10)");
    scanf("%d", &n);
    printf("enter elements");
    for (i = 0; i < n; i++) {
        scanf("%d", &a[i]);
    }
    large1 = large2= large3= a[0];
    for (i = 1; i < n; i++) {
        if (a[i] > large1) {
            large3=large2;
            large2=large1;
            large1 = a[i];
        }
        else if (a[i] > large2) {
            large3=large2;
            large2=a[i];
        }
        else if (a[i] > large3) {
            large3=a[i];
        }
    }
    printf("First largest number is %d\n ", large1);
    printf("Second largest number is %d\n ", large2);
    printf("Third largest number is %d\n ", large3);
    getch();
}
```

Method-III: using sorting method

```
#include<stdio.h>
#include<stdio.h>
void main()
{
    int a[10], n;
    int i,j,temp;
    clrscr();
    printf("enter number of elements you want in array(max=10)");
    scanf("%d", &n);
    printf("enter elements");
    for (i = 0; i < n; i++) {
        scanf("%d", &a[i]);
    }
    /*sorting process(bubble sorting)*/
    for(i=0;i<n-1;i++)
    {
        for(j=0;j<n-1-i;j++)
        {
            if(a[j]>a[j+1])
            {
                temp=a[j];
                a[j]=a[j+1];
                a[j+1]=temp;
            }
        }
    }

    printf("First largest number is %d\n ", a[n-1]);
    printf("Second largest number is %d\n ", a[n-2]);
    printf("Third largest number is %d\n ", a[n-3]);
    getch();
}
```

Q.4 WAP to print following output using character array (string).

```
H
H E
H E L
H E L L
H E L L O
H E L L
H E L
H E
H
```

Solution:-Method-I(using Formatted control string)

```
#include<stdio.h>
#include<conio.h>
void main()
{
    char string[10]="HELLO";
    int i;
    clrscr();
    for(i=1;i<=5;i++)
        printf("%-5.*s\n", i, string);

    for(i=4;i>=1;i--)
        printf("%-5.*s\n", i, string);

    getch();
}
```

Method-II

```
#include<stdio.h>
#include<conio.h>
void main()
{
    char string[10]="HELLO";
    int i,j;
    clrscr();
    for(i=1;i<=5;i++) {
        for(j=0;j<i;j++) {
            printf("%c", string[j]);
        }
        printf("\n");
    }
    for(i=4;i>=1;i--) {
        for(j=0;j<i;j++) {
            printf("%c", string[j]);
        }
        printf("\n");
    }
    getch();
}
```


Q.5 WAP to check whether given String is palindrome or not.

Example: "SARAS"

Solution:- Method-I: using string function

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
    char string[10],reverse[10];
    clrscr();
    printf("Enter string: ");
    gets(string);
    strcpy(reverse, string);    /*copying the string*/
    strrev(reverse);    /*reverse the string*/
    /*compare original and reverse string*/
    if(strcmp(string, reverse)==0)
        printf("given string is palindrome");
    else
        printf ("given string is not palindrome");
    getch();
}
```

Method-II: without using string function

```
#include<stdio.h>
#include<conio.h>
void main()
{
    char string[10];
    clrscr();
    int i,j, length,flag=0;
    printf("Enter string: ");
    gets(string);
    /*find the length of string*/
    for(length=0;string[length]!='\0';length++);
    /*compare character of string*/
    for(i=0,j=length-1; i<j; i++,j--) {
        if(string[i]!=string[j])
        {
            flag=1; break;
        }
    }
    if(flag==0)
        printf("given string is palindrome");
    else
        printf ("given string is not palindrome");
    getch(); }
```