DHARMSINH DESAI UNIVERSITY, NADIAD
FACULTY OF TECHNOLOGY
B.TECH. SEMESTER I [CE/EC/IT]
SUBJECT: (ESC-201) PROGRAMMING FOR PROBLEM SOLVING - I

Examination	: **Third Sessional**	**Seat No.**	32	:
Date	: **18/01/2023**	**Day**		: **Wednesday**
Time	: **03:45 p.m. to 05:00 p.m.**	**Max. Marks**		: **36**

INSTRUCTIONS:

1. Figures to the right indicate maximum marks for that question.
2. The symbols used carry their usual meanings.
3. Assume suitable data, if required & mention them clearly.
4. Draw neat sketches wherever necessary.

Q.1 **Do as directed.** **[12]**

CO2 E (a) What will be the output of the following program? [2]

```
#include <stdio.h>
int main () {
        int a[4][5] = { {1, 2, 3, 4, 5},
                        {6, 7, 8, 9, 10},
                        {11, 12, 13, 14, 15},
                        {16, 17, 18, 19, 20} };
        printf("%d\n", *(*(a+**a+1)+2));
        return(0);
}
```

CO5 A (b) Following are different ways of accessing structure members: [2]

(I) p.membername

(II) (*p_ptr).membername

Explain these using examples.

CO5 N (c) Interpret and differentiate the given pairs of declarations: [2]

```
(I)  int *p[5];
     int (*p)[5];

(II) int *p(int,int);
     int (*p)(int,int);
```

CO1 U (d) (I) “You cannot use scanf() with bit-fields”. Is the given statement true or false? Justify your answer. [2]

(II) Assuming the case of multiple members, “Memory occupied by union variable is always lesser than the similar structure variable”. Justify the given statement.

CO1 U (e) What is the difference between a pointer to a constant and constant pointer? [2]

CO3 E (f) What do the following fragments of C-program print? [2]

(I)

```
char c[] = "FACE2023";
char *p =c;
printf("%s", p + p[3] - p[1] + 2) ;
```

(II)

```
char a[] = "WORLD";
int i, j;
for (i = 0, j = 5; i < j; a [i++] = a [j--]);
printf ("%s\n", a+3);
```

Q.2 Attempt ***Any TWO*** from the following questions. **[12]**

CO1 R (a) (I) Explain the Chain of pointers with any suitable C-program as an example. [3]

(II) What are Bit-Fields? What are the several points to be considered when implementing bit-fields in structures? [3]

CO4 C (b) Write a program to calculate the subject-wise and student-wise totals and store them as a part of the structure. Take user input for the number of students and marks scored by each student in each subject. The structure template is given below: [6]

```
struct marks
{
    int sub1;
    int sub2;
    int sub3;
    int total;
};
```

CO4 A (c) Declare and define a function named **exchange.** Write a program to swap two integer values using call by address. [6]

Q.3 Attempt the following questions. [12]

CO5 E (a) What will be the output of the following program? [6]

```
#include <stdio.h>
struct node{
        int z;
        int *p;
};
int main ()
{
        struct node x[3], *y;
        int b[3] = {100,200,300};
        y = &x[0];
        x[0].z = 40;
        x[1].z = 50;
        x[2].z = 60;
        x[1].p = b;
        int theta = ++y -> z;
        int delta = (++y) -> z;
        int alpha = *y -> p++;
        int beta = (*y -> p)++;
        int gamma = *y++ -> p;
      printf("%d\n%d\n%d\n%d\n%d",theta,delta,alpha,beta,gamma);
        return(0);
}
```

CO4 C (b) Write a C program to create a **date** structure with members for the day, month, and year. Use a function named **isvalid** to determine whether a date inputted by the user is valid. [6]

Input Format: Three integers separated by space denoting date, month, and year.

Output Format: Display 1 if the date is valid, 0 otherwise.

Example:

(1) Input: 28 02 2023
    Output: 1

(2) Input: 150 12 2022
    Output: 0

(3) Input: 29 02 2021
    Output: 0

In Example (3), Output is 0 because 2021 is not a leap year.

OR

Q.3 Attempt the following questions. **[12]**

CO5 E (a) Consider the size of int = 4 bytes. What does the following program print? [6]

```
#include <stdio.h>
int main () {
   int x[] = {600,500,400,300,200,100};
   int *y[] = {x, x+1, x+2, x+3, x+4, x+5};
   int **z = y+1;
   z++;
   printf("%ld %ld %d\n",z-y,*z-x,**z);
   *z++;
   printf("%ld %ld %d\n",z-y,*z-x,**z);
   ++*z;
   printf("%ld %ld %d\n",z-y,*z-x,**z);
   ++**z;
   printf("%ld %ld %d\n",z-y,*z-x,**z);
   return(0);
}
```

CO4 C (b) [6]

```
typedef struct
{
   int id;
   char name[20];
   int marks[3];
   int total_marks;
}student;
```

Write a C program which uses an array of above given structure to scan the data of students. Take the number of students as the user input and then scan the information of each student (in sequence - id, name marks of three subjects), each field will be separated by space. Display the information of students sorted in ascending order based on total marks. When the total marks of two or more students are the same, then in output they should appear in the same sequence as input.

Bloom's Taxonomy levels : R-Remembering, U- Understanding, A-Applying, N-Analyzing, E- Evaluating, C-Creating