PES University, Bengaluru

(Established under Karnataka Act 16 of 2013)

END SEMESTER ASSESSMENT (ESA) - JULY - 2023

UE22CS151B - Problem Solving With C

Total Marks : 100.0

1.a. Draw a clear picture that shows the Program Development Life Cycle (**PDLC**) of a C Program. (4.0 Marks)

1.b. **Mention the outputs of below code snippets separately.**

i) int n=559, a;   printf("%d", a = printf("%d", a = printf("%d", n)));

ii) int auto = 8;   printf("%d",auto);

iii) printf("%d", -10?10:0);

iv) int a;   printf("%d",a = 6 | (8 == 8 == 8));

v) int c=11,d7;   printf("%d",c);

vi) int a = 100;   a == 100 || ++a == 101;   printf("%d", a);   (6.0 Marks)

1.c. Write a C Program to count the number of digits in a number taken through user input and also print the reverse of the number. Print the count of digits in the number as well.

Sample output:
Enter a number: 1234
The reverse of the number is 4321.
The number has 4 digits. (5.0 Marks)

1.d. i) **State True or False:**
a) ** is an operator in C.
b) C is both compiled and interpreted language.
c) There are multiple a.exe files in one folder.

ii) How many bytes does **sizeof('\n')** occupy?

iii) **Printf()** instead of **printf()** leads to ________ error. (5.0 Marks)

2.a. Write a C function **my_strcpy()** that accepts two strings as arguments, and emulates **strcpy()** in the **string.h** file. Test this function with the client code

Sample Output:

```
Enter Str1:
Exam Over
Str1: Exam Over
Str2: Exam Over
```

(6.0 Marks)

2.b. **Find the output of the following program.**

```
#include<stdio.h>
int main()
{
char str[] = "BEST";
int i;
for(i=0; str[i]; i++)
printf("%c %c %c %c\n", str[i], *(str+i), *(i+str), i[str]+2);
}
```

(4.0 Marks)

2.c. **i) Define pointers with an example program.**

ii) Find the output of the following C code

```
#include<stdio.h>
int what(int num,int res);
int main()
{
int a = 121;
printf("%d\n",what(a,0));
return 0;
}
int what(int num,int res)
{
if(num==0)
return res;
else
return what(num/10,res+(num%10));
}
```

(4.0 Marks)

2.d. Write a C function that returns the biggest element from an integer array **arr** with **n** elements. In the main(), call the function to the test.

For Example: If the array elements are **{9,4,5,7,2,3,19,6,1}**, the function returns the biggest element which is **19** in this case. (6.0 Marks)

3.a. **i) Find the output of the following code.**

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
int *p1 = (int*)malloc(sizeof(int));
*p1 = 300;
printf("%d ", *p1);
int *p2 = p1;
printf(" %d ", *p2);
*p2 = 777;
printf("%d ",*p1);
printf("%d",*p2);
free(p1);
p1 = NULL;
p2 = NULL;
}
```

ii)
To avoid dangling pointers after free() is used, ________ is assigned to the pointer.
Dereferencing a NULL pointer results in __________. (4.0 Marks)

3.b. **The expected output is 5001 and Cricket separated by a tab space. Find the errors in the below program and write a correct version of the program to get the expected output.**

```
#include<stdio.h>
#include<stdlib.h>

struct SPORT
{
int s_no;
char sport_name[100];
};
int main()
{
struct SPORT sp;
sp = calloc(sizeof(struct SPORT));
sp.s_no = 5001;
sp->sport_name = "Cricket";
printf("%d\t%s\n",sp.s_no,sp.sport_name);
return 0;
}
```

(5.0 Marks)

3.c. In XYZ company there are 3 salesmen. Each salesman sells 2 items. Write a C program using **two dimensional arrays** to display the **total sales by each salesman.**

Sample output:

```
Enter the data:
Enter the sales of 2 items sold by the sales man: 0
10 10
Enter the sales of 2 items sold by the sales man: 1
20 20
Enter the sales of 2 items sold by the sales man: 2
30 30
Total sales by salesman 0 = 20
Total sales by salesman 1 = 40
Total sales by salesman 2 = 60
```

(5.0 Marks)

3.d. Given the structure declaration and the client code, define the function **insert_end** to add nodes to the end of the linked list and define the **display** function as well to print the data in the nodes.

Sample output:

```
Enter the element:
30
Enter the element:
20
Enter the element:
10
30 20 10
```

```
typedef struct Node
{
   int data;
   struct Node *link;
}Node;

int main()
{
   Node *head = NULL;
   int element;
   for(int i=0;i<3;i++)
   {
   printf("Enter the element:\n");
   scanf("%d",&element);
   head = insert_end(head,element);
   }
   display(head);
}
```

(6.0 Marks)

4.a. The file **sample.txt** exists with some data. Write a C program to find the number of the characters in the text file using **fgetc()** function. (5.0 Marks)

4.b. Given a sorted array of integers, write a **recursive** function which searches for a given integer using binary search approach and returns the index of it, returns -1 otherwise. Use the below function declaration to define the function.

int binary_search(int *array, int key, int start, int end);

array: pointer to the array where the key has to be searched.
key : element to be searched
start: 0
end : size of the array – 1 (6.0 Marks)

4.c. i) A function **opr** takes two integer parameters and returns an integer. Write a valid function pointer declaration that can store the address of opr.

ii)
Find the output of the following code.

```
#include <stdio.h>
#include<string.h>
int main()
{
char line[100]="Hello,All,Friends";
printf("%s ",strtok(line,","));
printf("%s ",strtok(NULL,","));
printf("%s ",strtok(line,","));
}
```

(4.0 Marks)

4.d. Write short notes on array of pointers to structures with an example program.
(2 marks - Definition, 3 marks - Example program)

(5.0 Marks)

5.a. Give brief notes on the following keywords suitable code snippets.
i) volatile
ii) register

(4.0 Marks)

5.b. **Find the output of the following.**

```
#include <stdio.h>
enum cars{TATA=1,BMW=4,KIA,MG=7};
int main()
{
enum cars c;
c=KIA;
printf("%d ",KIA);
switch(c)
{
case TATA:printf("TATA");break;
case BMW:printf("BMW");break;
case KIA:printf("KIA");break;
case MG:printf("MG");break;
}
printf(" %d ",TATA);
printf("%d ",MG);
}
```

(4.0 Marks)

5.c. **Find the output of the following C programs.**

i)

```
#include<stdio.h>
int main()
{ char a = 'q';        char b = 'p';         const char *c = &b;        *c = 'z';
printf("%c",*c);        return 0;
}
```

ii)

```
#include<stdio.h>
int main()
{  int i = 999;        int j = 777;        int* const p = &i;         *p = j;
  printf("%d\n",*p);        return 0;}
```

iii)

```
#include<stdio.h>
int main()
{     printf("%d",sizeof(short) >= sizeof(int));     return 0;       }
```

iv)

```
#include<stdio.h>
#include<stddef.h>
union A
{   double x;      int y;      float z;   };
int main()
{   printf("%lu ",offsetof(union A,z));
   }
```

v)

```
 #include<stdio.h>
void fun();
int main()
{     fun();        fun();        return 0;        }

void fun()
{ static int a = -5;        a--;        printf("%d\t",a);        }
```

(6.0 Marks)

5.d. **i) Mention any three preprocessor directives with its purpose.**

ii) Find the output of the following code.

```
#include<stdio.h>
#define MAX 6
#define fun(a,b) a*b
int main()
{
    printf("%d\t",fun(MAX,7+7));
    #undef MAX
    int MAX = 77;
    printf("%d\t",MAX);
    #define MAX 9
    printf("%d",MAX);
}
```

(6.0 Marks)