Set A -Answer Key CT-3 - PPS CLAT - 3 SET-A QUESTION PAPER

Programming For Problem Solving (SRM Institute of Science and Technology)

SRM Institute of Science and Technology
Faculty of Engineering and Technology
School of Computing

Set A

DEPARTMENT OF COMPUTING TECHNOLOGIES

SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamilnadu

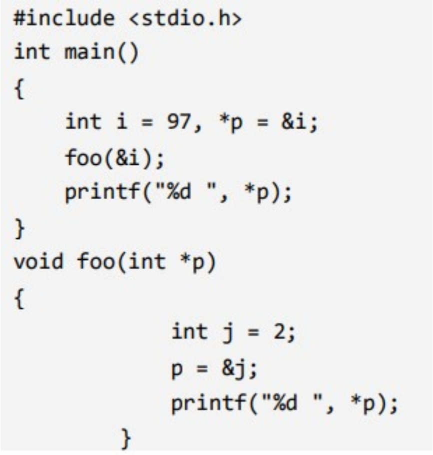
Academic Year: 2021 – 2022 EVEN

Test: CLAT-3 **Date:**
Course Code & Title: 18CSS101J & Programming for Problem Solving **Duration:** 2 periods
Year & Sem: I & II **Max. Marks: 50**

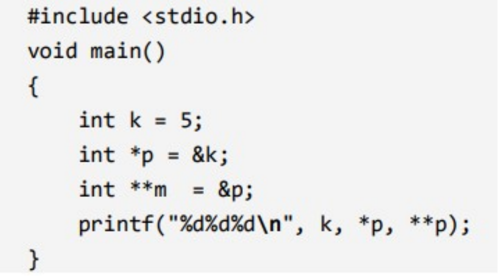
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	-	-	-	-	-	-	-	-	-	-	-	-	3
CO2	3	3	2	-	-	-	-	-	-	-	-	-	-	-	3
CO3	3	3	3	-	-	-	-	-	-	-	-	-	-	-	3
CO4	3	2	3	3	-	-	-	-	-	-	-	-	-	-	3
CO5	3	3	3	-	-	-	-	-	-	-	-	2	-	-	3
CO6	2	2	-	-	-	-	-	-	-	-	-	2	-	-	3

Part – A
(10 x 1 = 10 Marks)
Instruction: Answer all the questions

Q. No	Answer with choice variable	Marks	BL	CO	PO	PI Code
1	Identify the correct syntax to send a 3-dimensional array as a parameter (Assuming declaration int a[5][4][3];) **Answer: a) func(a);**	**1**	**2**	**3**	**1**	**1.7.1**
2	At what stage of building a C program does Pragma work? **Answer: a) Before Compilation**	**1**	**1**	**3**	**1**	**1.7.1**
3	Pick out the incorrect statement with respect to NULL pointer. **Answer: C) datatype *NULL=ptr;**	**1**	**1**	**4**	**1**	**1.7.1**

4	Find the output of this C code
	
	#include <stdio.h>
	int main()
	{
	int i = 97, *p = &i;
	foo(&i);
	printf("%d ", *p);
	}
	void foo(int *p)
	{
	int j = 2;
	p = &j;
	printf("%d ", *p);
	}
	Answer:a) 2 97
		1	**2**	**4**	**1**	**1.7.1**

5	What is the output of this C code?	**1**	**2**	**4**	**1**	**1.7.1**

	
	#include <stdio.h>
	void main()
	{
	    int k = 5;
	    int *p = &k;
	    int **m = &p;
	    printf("%d%d%d\n", k, *p, **p);
	}
	Answer:d) Compile time error					

6	The preprocessor directive used to give additional information to the compiler, beyond which is conveyed in the language is ____________ **Answer: c) #pragma**	**1**	**1**	**5**	**1**	**1.7.1**

7	Predict the output of the following C code
	#include <stdio.h>
	int main()
	{
	    int ary[4] = {1, 2, 3, 4};
	    printf("%d\n", *ary);
	}
	Answer: a)1
		1	**2**	**5**	**1**	**1.7.1**

8	Among the given pointers which of the following cannot be incremented? a) int b) char c) float d) void **Answer :d)Void**	**1**	**1**	**5**	**1**	**1.7.1**
9	The declaration instruction for the variable 'ptr' as the pointer to array of 6 floats is a) float *ptr[6] b) float [6]*ptr c) float(*ptr)[6] d) float(*ptr)(6) **Answer: d) float(*ptr)(6)**	**1**	**2**	**5**	**1**	**1.7.1**
10	The size of a union is determined by the size of the __________ a) First member in the union b) Last member in the union c) Biggest member in the union d) Sum of the sizes of all members **Answer: c) Biggest member in the union**	**1**	**1**	**5**	**1**	**1.7.1**
Part – B **(4 x5 = 20 Marks)** **Instruction : Answer all the questions**						

11	Write a macro MAX (a, b) to find the maximum of a and b
	Answer:
	#include <stdio.h>
	#define MAX(a,b) ((a>b)?a:b)
	int main()
	{
	    int a,b,max;
	    printf("Enter first number: ");
	    scanf("%d",&a);
	    printf("Enter second number: ");
	    scanf("%d",&b);
		5	**3**	**3**	**2**	**2.5.2**

	max=MAX(a,b);
	printf("Maximum number is:%d\n",max);
	return 0;
	}					

12	If ‘a’ is pointing to ‘num’ and ‘b’ is pointing to ‘a’, Can you access the value of ‘num’ with ‘b’. Justify your answer.
	Answer:
	Yes. We can access the value of ‘num’ with ‘b’.
	Justification:
	The given statements says *a=&num and **b=&a. That means we are using Double pointer ‘b’ to store the address of ‘a’ which is in turn a pointer to store the address of ‘num’. With the help of the instruction printf(“%d”,**b); we can print the value of ‘num’
	Example:
	#include <stdio.h>
	int main()
	{
	int num=5;int *a=&num,**b=&a;
	printf("num is: %d\n",**b);
	return 0;
	}
		5	**4**	**4**	**2**	**2.5.2**

13	Differentiate between malloc(),calloc() and realloc(). Malloc() This function only carries unuseful/garbage values Faster allocation time than calloc() Do not initialize the memory. Less secure than calloc() More time-efficient. Returns the starting address. Creates and assign only a single block of memory Calloc() This function only carries unuseful/garbage values Faster allocation time than calloc() Do not initialize the memory. Less secure than calloc() More time-efficient. Returns the starting address. Creates and assign only a single block of memory	**5**	**3**	**5**	**2**	**2.5.2**
14	Raja wants to add two distances in inches and feet using unions. Write a logic to implement the above scenario **#include <stdio.h>**	**5**	**4**	**5**	**2**	**2.5.2**

```
Union Distance {
  int feet;
  float inch;
} d1, d2, result;

int main() {
  // take first distance input
  printf("Enter 1st distance\n");
  printf("Enter feet: ");
  scanf("%d", &d1.feet);
  printf("Enter inch: ");
  scanf("%f", &d1.inch);

  // take second distance input
  printf("\nEnter 2nd distance\n");
  printf("Enter feet: ");
  scanf("%d", &d2.feet);
  printf("Enter inch: ");
  scanf("%f", &d2.inch);

  // adding distances
  result.feet = d1.feet + d2.feet;
  result.inch = d1.inch + d2.inch;

  // convert inches to feet if greater than 12
  while (result.inch >= 12.0) {
    result.inch = result.inch - 12.0;
    ++result.feet;
  }
   printf("\nSum of distances = %d\'-%.1f\"", result.feet,
result.inch);
  return 0;
}
```

Part – C

(2x10 = 20 Marks)

Instruction : Answer all the questions

15 a	Elaborate on passing array elements to Functions with an example program. **Answer:** ✶ Arrays can be passed as a parameter to a function. ✶ When it is being passed, the name of the array is enough as an argument instead passing it as an entire array. ✶ Since the name of the array itself holds the address, no need to include & in the function call. ✶ Since it passes address, in case of any modification occurs in called function automatically reflects back in the calling function. ✶ Multidimensional arrays are also allowed to be passed as arguments to functions. • The first dimension is omitted	**10**	**3**	**3**	**2**	**2.5.2**

when a multi dimensional array is used as a formal parameter in a function.

Formal and Actual Parameters One-Dimensional Array:
Actual Parameters: int a[10];
Function call: add(a); here, a is the base address of the array add is the function
Formal Parameters: within the function body

```
void add(int a[])
{
....
}
```

int a[] - is a formal parameter, since it is a on dimensional array size doesn't matter.

Formal and Actual Parameters in Multi Dimensional Array:
Actual Parameters: int a[10][10];
Function call: add(a); here, a is the base address of the array 'a' b is the base address of the array 'b' add is the function
Formal Parameters: within the function body void add(int a[][10])

```
{
....
}
```

int a[][10] - is a formal parameter, the first dimension is omitted only column value is taken into the account.

Passing Single element of an array:
Example :
int a[5]={0,1,2,3,4};
Function call:
add(a[1]);
Here, add is the function name, a[1] is the second element of the array, value 1 is passed to the array.

Passing an entire array to a function:

```
int main()
{
int a[50],i,n;
scanf("%d",&n);
for(i=1;i<=n;i++)
scanf("%d",&a[i]);
add(a,n);
for(i=1;i<=n;i++)
printf("%d\n",a[i]);
return 0;
}
void add(int a[],int n)
{
int r;
for(r=1;r<=n;r++)
```

	{
	a[r]=a[r]+5;
	}
	}					

	OR					

b	Develop a program in C to print a string in reverse using a pointer
	Answer:
	#include <stdio.h>
	#include <stdlib.h>
	#include<string.h>
	int main()
	{
	   char str1[100],str2[100];
	   char *p1,*p2;
	   printf("Enter a String\n");
	   gets(str1);
	   p1=str1+strlen(str1)-1;
	   p2=str2;
	   while(p1>=str1)
	   {
	      *p2=*p1;
	      p2++;
	      p1--;
	   }
	   *p2='\0';
	   printf("Original String: %s\n",str1);
	   printf("Reverse String: %s",str2);
	   return 0;
	}
		10	**3**	**4**	**2**	**2.5.2**

16 a	Rahul wants to check whether two files named A.TXT and B.TXT are identical or not. Use suitable component to solve the above problem.
	
	#include<stdio.h>
	
	int main() {
	  FILE *fp1, *fp2;
	  int ch1, ch2;
	  char fname1[40], fname2[40];
	
	  printf("Enter name of first file :");
	  gets(fname1);
	
	  printf("Enter name of second file:");
	  gets(fname2);
	
	  fp1 = fopen(fname1, "r");
	  fp2 = fopen(fname2, "r");
	
	  if (fp1 == NULL) {
	    printf("Cannot open %s for reading ", fname1);
	    exit(1);
	  } else if (fp2 == NULL) {
	    printf("Cannot open %s for reading ", fname2);
	    exit(1);
	  } else {
	    ch1 = getc(fp1);
	    ch2 = getc(fp2);
		10	**4**	**5**	**2**	**2.5.2**

	while ((ch1 != EOF) && (ch2 != EOF) && (ch1 == ch2))
	{
	ch1 = getc(fp1);
	ch2 = getc(fp2);
	}
	
	if (ch1 == ch2)
	printf("Files are identical n");
	else if (ch1 != ch2)
	printf("Files are Not identical n");
	
	fclose(fp1);
	fclose(fp2);
	}
	return (0);
	}					

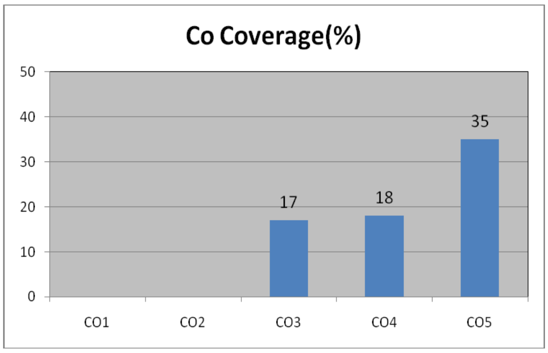
	OR					

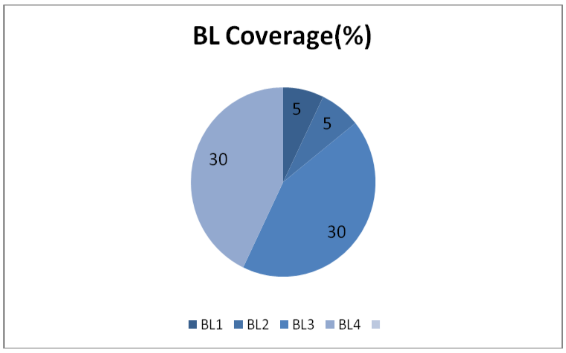
b	A company wants to maintain employee database name, age, salary, date of birth and Aadhar no. Write a program to read and display the above employee information using suitable method.
	
	include <stdio.h>
	
	/*structure declaration*/
	struct employee{
	char name[30];
	int age;
	float salary;
	char Aadharno[30];
	int dob;
	
	};
	
	int main()
	{
	
	struct employee emp;
	
	printf("\nEnter details :\n");
	printf("Name ?:");
	gets(emp.name);
	printf("age:");
	scanf("%d",&emp.age);
	printf("Salary :");
	scanf("%f",&emp.salary);
	printf("Aadharno :");
	gets(emp.Aadharno);
	printf("dob");
	scanf("%d",&dob);
	
	printf("\nEntered detail is:");
		10	**4**	**5**	**2**	**2.5.2**

	printf("Name: %s" ,emp.name);
	printf("age: %d" ,emp.age);
	printf("Salary: %f\n",emp.salary);
	printf("Aadharno:%s" ,emp.Aadharno);
	return 0;
	}					

***Performance Indicators are available separately for Computer Science and Engineering in AICTE examination reforms policy.**

Course Outcome (CO) and Bloom's level (BL) Coverage in Questions





Approved by the Audit Professor/Course Coordinator