|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg. No. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



|  |
| --- |
| SET A |

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRMIST, SCHOOL OF COMPUTING

CYCLE TEST – I

Course Code and Title: 21CSS101J / Programming for Problem Solving Year / Sem: I / I

Max. Marks: 10 Duration: 10 Minutes Date: 10.10.2023

|  |
| --- |
| For Multiple Choice Questions, pls Shade the answer in Circle |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sl. No** | Part A (10\* 1 = 10 Marks) (Answer ALL Questions) | **Marks** | **CO** | **PO** | **BL** | **PI CODE** |
|  | The Operator & is used as  A. Logical AND  B. Bitwise AND  C. Logical OR  D. Bitwise OR  **Answer: B** | 1 | 1 | 1 | 1 | 2.5.2 |
|  | Find the value of x in this C code     1. 3.75 2. Depends on the compiler 3. 3 4. 24   **Answer: D** | 1 | 1 | 2 | 1 | 2.5.2 |
| 1. 5 | Identify the one of the following is true for the variable name in C.   1. Variable names cannot start with a digit 2. Variable can be of any length 3. They can contain alphanumeric characters as well as special characters. 4. Reserved word can be used as a variable name   **Answer: A** | 1 | 1 | 1 | 1 | 2.5.2 |
| 1. 6 | The size of the () operator is a \_\_\_\_ type of operator used to calculate  the size of the data types  **Answer. Unary** | 1 | 1 | 1 | 1 | 2.5.2 |
|  | A binding is \_\_\_\_\_\_\_\_\_\_ if it first occurs \_\_\_\_\_\_\_\_execution or can change during the execution of the program.  **Answer: dynamic, during** | 1 | 1 | 2 | 1 | 2.5.2 |
|  | Which of the following are constants or values in programming languages? (Select two.)  A) if  B) 3.14  C) while  D) True  Answer: B & D | 1 | 2 | 2 | 1 | 2.5.2 |
|  | Match the following:  Variables - Stores a collection of elements of the same data type.  Char - A sequence of characters.  Arrays - Stores a collection of elements of the same data type.  Strings - A data type that stores the Character  Answer: 1 – C, 2 – D, 3- A, 4 - B | 1 | 2 | 2 | 1 | 2.5.2 |
|  | Which of the following statements is used for making decisions in most programming languages?  A) while  B) for  C) switch  D) do-while  **Answer: C** | 1 | 2 | 2 | 1 | 2.5.2 |
|  | How do you exit a while loop prematurely within its body?  A) Use the break statement.  B) Use the continue statement.  C) Use the return statement.  D) You cannot exit a while loop prematurely.  **Answer: A** | 1 | 2 | 2 | 1 | 2.5.2 |
|  | In most programming languages, how are array elements accessed?  A) Using a loop construct.  B) By specifying the element's memory address.  C) Using the array name followed by an index in square brackets.  D) By using the getElement() function.  **Answer: C** | 1 | 2 | 2 | 1 | 2.5.2 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg. No. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



COLLEGE OF ENGINEERING AND TECHNOLOGY

SRMIST , SCHOOL OF COMPUTING

CYCLE TEST – I

Course Code and Title: 21CSS101J / Programming for Problem-Solving

Max. Marks: 40 Duration: 90 Minutes

Year / Sem: I / I Date: 10.10.2023

Part B (4\* 5 = 20 Marks) Answer ALL Questions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sl.No** | **Questions** | **Marks** | **CO** | **PO** | **BL** | **PI Code** |
| 11 | Ganesh had taken the loan amount of Rs. 50000 for 3 years in the HDFC bank at 3.0% interest. He went to the bank to make the payment of his loan. Therefore,  could you help him to pay the correct amount by  writing a C program to find the simple interest of his loan amount?  Answer :  void main()  {  float amount, rate, simple\_interest;  int months;  printf(“Provide amount and interest in  the format $amount rate%% ”);  scanf(“%f%f%”;, &amount, &rate);  printf(“Enter the period in months : ”);  scanf(“%d”, &months);  simple\_interest = amount \* rate \*  months / 100 ;  printf(“Simple interest : %f\n”;,  simple\_interest);  } | 5 | 1 | 2 | 2 | 2.6.3 |
| 12 | Samyuktha got a good job in an MNC company. She was confused  about the salary credited to her account. To verify if the correct amount of HRA and DA was provided to her, Samyuktha planned to develop software that calculates her Gross salary.  The Salary policy of Samyuktha Company is as follows:  Basic Pay is Rs.46,000/=  HRA is 40% of the basic pay and DA is 50% of the basic pay.  Can you help Samyuktha in the software development by Printing the Gross salary of Samyuktha by adding the HRA and DA to the basic pay?  int main()  {  float hra,da,bp,gs;  bp=46000;  hra=bp\*40/100;  da=bp\*50/100;  gs=hra+da+bp;  printf(“Gross Salary of Samyuktha is %.f”.gs);  return 0;  } | 5 | 1 | 2 | 2 | 2.6.3 |
| 13 | You are given an array of integers. Write a program to find the largest and smallest numbers in the array and calculate their sum. Also, find the average of all the numbers in the array. Provide a C programming code snippet to solve this problem.  Answer:  #include <stdio.h>  int main() {  int arr[] = {12, 45, 6, 78, 23, 54, 9, 87, 2, 51};  int n = sizeof(arr) / sizeof(arr[0]);  // Initialize variables to store the largest, smallest, and sum of elements  int largest = arr[0];  int smallest = arr[0];  int sum = 0;  // Calculate the largest, smallest, and sum of elements  for (int i = 0; i < n; i++) {  if (arr[i] > largest) {  largest = arr[i];  }  if (arr[i] < smallest) {  smallest = arr[i];  }  sum += arr[i];  }  // Calculate the average  double average = (double)sum / n;  // Display the results  printf("Largest Number: %d\n", largest);  printf("Smallest Number: %d\n", smallest);  printf("Sum of Numbers: %d\n", sum);  printf("Average of Numbers: %.2lf\n", average);  return 0;  } | 5 | 2 | 2 | 2 | 2.5.2 |
| 14 | What is a null pointer, and when is it useful in programming? Provide an example where a null pointer is used appropriately.  Answer :  #include <stdio.h>  #include <stdlib.h>  int main() {  int \*ptr = NULL;  // Attempt to allocate memory for an integer  ptr = (int \*)malloc(sizeof(int));  // Check if memory allocation was successful  if (ptr == NULL) {  printf("Memory allocation failed. Exiting program.\n");  return 1; // Exit with an error code  }  // Memory allocation was successful; use the allocated memory  \*ptr = 42;  printf("Value stored at the allocated memory: %d\n", \*ptr);  // Free the allocated memory before exiting  free(ptr);  return 0;  } | 5 | 2 | 2 | 2 | 2.6.3 |

**Part C (2 \* 10 = 20 Marks) [NOTE: Qns, 15 to 18 can have sub-divisions]**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sl.No** | **Question** | **Marks** | **CO** | **PO** | **BL** | **PI Code** |
| 15 | i. Develop a C program to read a number from the user and multiply the last digit and the second last digit.  Sample Input:  N = 2869  Output:  6 \* 9 = 54  void main()  {  Int n,ld,sld;  scanf(“%d”,&n);  ld=n%10;  sld=n%100;  printf(“%d”,ld\*sld);  }  ii, Ramesh is playing with a rectangular box (cuboid). He knows the length, width, and height of the box. Help him to find the surface area of the box using a C program.  int main()  {  float l,b,h,s,v;  printf(“n Enter length of cuboid “  scanf(“%f”,&l);  printf(“n Enter breadth of cuboid “);  scanf(“%f”,&b);  printf(“n Enter height of cuboid “);  scanf(“%f”, &h);  s=2\*(l\*b+b\*h+b\*h);  v=l\*b\*h;  printf("\n The surface area of cuboid is '%f'",s);  printf("\n \n The volumer of cuboid is '%f' \n \n",v);  return 0;  } | 10 | 1 | 2 | 2 | 2.6.3 |
| **(OR)** | | | | | | |
| 16 | a) What is the purpose of the else if statement in programming? How does it differ from if and else statements? Provide an example.  b) Write a simple C program that uses else if statements to categorize a student's grade (A, B, C, D, or F) based on their score. Include handling for invalid input  Answer :  #include <stdio.h>  int main() {  int score;  // Input: Ask the user for their score  printf("Enter your score: ");  scanf("%d", &score);  // Check if the input is valid (0 to 100)  if (score >= 0 && score <= 100) {  // Grade categorization using else if statements  if (score >= 90) {  printf("Your grade is A\n");  } else if (score >= 80) {  printf("Your grade is B\n");  } else if (score >= 70) {  printf("Your grade is C\n");  } else if (score >= 60) {  printf("Your grade is D\n");  } else {  printf("Your grade is F\n");  }  } else {  // Invalid input handling  printf("Invalid input! Please enter a score between 0 and 100.\n");  }  return 0;  } | 10 | 1 | 2 | 2 | 2.5.2 |
| 17 | i. How does the break statement affect the execution of a loop? Provide a scenario where using a break is appropriate  ii. Explain the purpose of the continue statement in C programming. How does it differ from break?  Answer:  #include <stdio.h>  int main() {  int i;  printf("Using break statement:\n");  for (i = 1; i <= 10; i++) {  if (i == 5) {  printf("Encountered 5, breaking the loop.\n");  break; // Breaks out of the loop when i equals 5  }  printf("%d ", i);  }  printf("\n\nUsing continue statement:\n");  for (i = 1; i <= 10; i++) {  if (i % 2 == 0) {  printf("Skipping even number %d.\n", i);  continue; // Skips even numbers in the loop  }  printf("%d ", i);  }  return 0;  } | 10 | 2 | 2 | 3 | 2.6.3 |
| **(OR)** | | | | | | |
| 18 | a) Describe a scenario in which you use nested for loops to create a pattern or grid. Provide a code example illustrating this scenario.  b) Create a program that uses nested for loops to print a right-angled triangle pattern of asterisks (\*). Allow the user to specify the triangle's height.  Answer: i)  #include <stdio.h>  int main() {  int rows, cols;  // Input: Ask the user for the number of rows and columns  printf("Enter the number of rows: ");  scanf("%d", &rows);  printf("Enter the number of columns: ");  scanf("%d", &cols);  // Nested for loops to create the grid  for (int i = 0; i < rows; i++) {  for (int j = 0; j < cols; j++) {  printf("\* ");  }  printf("\n"); // Move to the next row after each inner loop  }  return 0;  }  Answer : ii.  #include <stdio.h>  int main() {  int height;  // Input: Ask the user for the height of the triangle  printf("Enter the height of the right-angled triangle: ");  scanf("%d", &height);  // Nested for loops to create the triangle  for (int i = 1; i <= height; i++) {  // Print spaces for alignment  for (int j = 1; j <= height - i; j++) {  printf(" ");  }  // Print asterisks for the current row  for (int k = 1; k <= i; k++) {  printf("\* ");  }  printf("\n"); // Move to the next row  }  return 0;  } | 10 | 2 | 2 | 3 | 2.5.3 |

**Quality Alignment Matrix (QAM)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Question No.** | **BLOOM’S LEVEL Distribution** | | | | | |
| L1 | L2 | L3 | L4 | L5 | L6 |
| PART-A | | | | | | |
| **1** |  |  |  |  |  |  |
| **2** |  |  |  |  |  |  |
| **3** |  |  |  |  |  |  |
| **4** |  |  |  |  |  |  |
| **5** |  |  |  |  |  |  |
| **6** |  |  |  |  |  |  |
| **7** |  |  |  |  |  |  |
| **8** |  |  |  |  |  |  |
| **9** |  |  |  |  |  |  |
| **10** |  |  |  |  |  |  |
| **PART-B** | | | | | | |
| **11** |  |  |  |  |  |  |
| **12** |  |  |  |  |  |  |
| **13** |  |  |  |  |  |  |
| **14** |  |  |  |  |  |  |
| **15** |  |  |  |  |  |  |
| **PART-C** | | | | | | |
| **16** |  |  |  |  |  |  |
| **17** |  |  |  |  |  |  |
| **18** |  |  |  |  |  |  |
| **19** |  |  |  |  |  |  |
| **Total (70)** |  |  |  |  |  |  |
| **%** |  |  |  |  |  |  |

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

**L1 + L2 = %**

**L3 + L4 = %**