PART I – SHORT ANSWER

|  |
| --- |
| 1. 13+21%4-2 => 12 |
| 1. ((5+1)/((2+2+2)/2)) = 2 |
| 1. What are logical errors? When and why does it occur?   Sources:  <https://www.studysmarter.co.uk/explanations/computer-science/computer-programming/logical-error/#:~:text=the%20compilation%20process.-,A%20logical%20error%20in%20C%20is%20a%20mistake%20in%20the,t%20what%20the%20programmer%20expected>.  Logical error 🡪 semantic errors   * Mistake in implementation logic (output is not what was expected) |
| 1. Rules for declaring variables in C++ |
| 1. What is type casting? Give 2 examples   Type casting: |
| 1. Explain 2 forms of decrement operator   Pre-decrement operator    Post-decrement operator |
| 1. C++ functions   Syntax for function Prototype, Call and Body |
| 1. 2D array – syntax |
| 1. Loop |
| 1. Difference between variable declaration & variable initialisation |
| 1. Program 🡪 input 3 numbers and displays maximum number   Version 1: |
|  |
|  |
|  |
| 1. Recursive function: |

PART II – CODING

FUNCTIONS ONLY

|  |
| --- |
|  |
| #include <iostream>  using namespace std;  // Function to convert Fahrenheit to Celsius  namespace convertTempSpace {  float convertToFah(float celsius) {  float resultFahrenheit;  resultFahrenheit = celsius \* 1.8 + 32;  return resultFahrenheit;  }    float convertToCel(float fahrenheit){  float resultCelsius;  resultCelsius = (fahrenheit - 32) / 1.8;  return resultCelsius;  }  }  int main() {  //write a loop to show F to C equivalent from 0 ~ 100 C  for (int i = 0; i <= 100; i++){  cout << "Celsius " << i << " --> Fahrenheit : " << convertTempSpace::convertToFah(i) << endl;  }  //write a loop to show C to F equivalent from 32 ~ 212 F  for (int j = 32; j <= 212; j++){  cout << "Fahrenheit " << j << " --> Celsius : " << convertTempSpace::convertToCel(j) << endl;  }  return 0;  } |
| 1. Matrix-matrix multiplication |
| 1. The greatest common divisor (GCD) of two integers is the largest integer that evenly divides each of the two numbers. Write function gcd that returns the greatest common divisor of two integers.   #include <iostream>  using namespace std;  int gcd(int a, int b) {  if (b == 0)  return a;  return gcd(b, a % b);  }  int main(){  int a = 98, b = 56;  cout << "GCD of " << a << " and " << b << " = " << gcd(a,b);  return 0;  } |
| 1. Take string input from user 🡪 find particular character given by user 🡪 find number of times it appears in the string   #include <iostream>  #include <bits/stdc++.h>  #include <string>  using namespace std;  //takes in string input and its length + char to find  int findNumOfChar(string strInput, char findChar){  int counter = 0;  for (int i = 0; i < strInput.length(); i++){  if (strInput[i] == findChar){  counter++;  }  }  return counter;  }  int main(){  string str;  char findCharVal;  cout << "Enter the string: ";  cin >> str;  cout << "Enter a character to find: ";  cin >> findCharVal;  cout << "String: " << str << "; character to find: " << findCharVal << endl;    int numberOfOccurrences = findNumOfChar(str, findCharVal);  cout << "Number of occurrences of " << findCharVal << " in " << str << " = " << numberOfOccurrences << endl;  return 0;  } |
| 1. Write a function that perform each of the following array operations:   a) Initialize the 10 elements of integer array counts to zeros.  b) Add 1 to each of the 15 elements of integer array bonus.  c) Read the 12 values of floating-point array monthlyTemperatures from the keyboard. d) Print the five values of integer array bestScores in column format.  **Note:** Pass the array by reference  #include <iostream>  using namespace std;  namespace arrayOperations {  //a. Initialize the 10 elements of integer array counts to zeros.  void initZeroes(int (&arr)[10]){  for (int i = 0; i < 10; ++i){  arr[i] = 0;  }  }    //b) Add 1 to each of the 15 elements of integer array bonus.  void addOnesPlus(int (&arr)[15]){  for(int i = 0; i < 15; ++i){  ++arr[i]; //or arr[i] += 1  }  }    //c) Read the 12 values of floating-point array monthlyTemperatures from the keyboard.  void getTempValues(float (&arr)[12]){  for (int i = 0; i < 12; ++i){  cout << "Enter month " << i+1 << ": " << endl;  cin >> arr[i];  }  }    //d) Print the five values of integer array bestScores in column format.  void printColFormat(int (&arr)[5]){  for (int i = 0; i < 5; ++i){  cout << arr[i] << endl;  }  }  }  int main(){  int arr[10] = {0};  int bonus[15] = {0};  float monthlyTemp[12] = {0};  int bestScores[5] = {3,5,12,50,23};    //arrayOperations::initZeroes(arr);    arrayOperations::getTempValues(monthlyTemp);    //arrayOperations::printColFormat(bestScores);  return 0;  } |
| 1. Define a function called hypotenuse that calculates the length of the hypotenuse of a right triangle when the other two sides are given. The function should take two arguments of type double and return the hypotenuse as a double.   #include <iostream>  #include <cmath>  using namespace std;  double hypotenuse(double a, double b){  double c = sqrt((a\*a)+(b\*b));  return c;  }  int main(){  double a, b, c;  cout << "Enter first angle: " << endl;  cin >> a;  cout << "Enter second angle: " << endl;  cin >> b;  c = hypotenuse(a, b);  cout << "Hypotenuse of " << a << " and " << b << " = " << c << endl;  return 0;  } |
| 1. Write a program that read the distance between two cities in kilometers and change it into Meters, Feet, Inches, Centimeters and Millimeters.   #include <iostream>  using namespace std;  namespace distanceConvert(float km){  float kmToMeter(){  float result = km \* 1000;  return result;  }  float kmToFeet(){  float result = km \* 3280.84;  return result;  }  float kmToInch(){  float result = km \* 39370.1;  return result;  }  float kmToCm(){  float result = km \* 100000;  return result;  }  float kmToMm(){  float result = km \* 1000000;  return result;  }  } |
| 1. Triangle   #include <iostream>  using namespace std;  int main(){  int rows = 5;    for (int i = 1; i <= rows; i++){  //print spaces  for(int j = 0; j < rows - i; ++j){  cout << " ";  }  //print stars  for (int k = 0; k < (2\*i-1); k++){  cout << "\*";  }  cout << endl;  }  return 0;  } |
| 1. Find HCF and LCM of 2 numbers given by user.   #include <iostream>  using namespace std;  int findHCF (int a, int b){  if (b == 0)  return a;  return findHCF(b, a % b);  }  int findLCM (int a, int b){  return (a / findHCF(a,b)\*b);  }  int main(){  int num1, num2;  cout << "Enter first number: ";  cin >> num1;  cout << "Enter second number: ";  cin >> num2;  cout << "HCF = " << findHCF (num1, num2) << "; LCM = " << findLCM (num1, num2) << endl;  return 0;  } |
| 1. In a company an employee is paid as under:   If his basic salary is less than Rs. 1500, then HRA = 10% of basic salary and DA = 90% of basic salary.  If his salary is either equal to or above Rs. 1500, then HRA = Rs. 500 and DA = 98% of basic salary.  If the employee's salary is input by the user write a program to find his gross salary.  #include <iostream>  #include <bits/stdc++.h>  using namespace std;  int main(){  float basicSalary, grossSalary, hra, da;    //basic salary  cout << "Enter basic salary (Rs): ";  cin >> basicSalary;    //if else statement  if(basicSalary < 1500){  hra = 0.1 \* basicSalary;  da = 0.9 \* basicSalary;  } else {  hra = 500;  da = 0.98 \* basicSalary;  }    grossSalary = basicSalary + hra + da;  cout << "Gross salary : Rs. " << grossSalary << endl;  return 0;  } |