|  |  |
| --- | --- |
| **SESSION** | **February/MARCH 2024** |
| **PROGRAM** | **MASTER OF COMPUTER APPLICATIONS (mCA)** |
| **SEMESTER** | **I** |
| **course CODE & NAME** | **DCA6102** |
| **CREDITS** | **4** |
| **nUMBER OF ASSIGNMENTS & Marks** | **02**  **30** |

**Set-Ist**

**1. What is the basic structure of C program? Explain printf() and scanf() functions with an example.**

**Ans:**C is characterized by the ability to write very concise source programs, due in part to the large number of operators included within the language. It has a relatively small instruction set, though actual implementations include extensive library functions which enhance the basic instructions.

There are some variables that are used in more than one function. Such variables are called global variables and are declared in the global declaration section that is outside of all the functions. Every C program must have one main function section. This section contains two parts, declaration part and executable part. The declaration part declares all the variables used

Its Half solved only

Buy Complete assignment from us

**Price – 190/ assignment**

**MUJ Manipal University Complete SolvedAssignments session FEB 2024**

buy cheap assignment help online from us easily

we are here to help you with the best and cheap help

**Contact No – 8791514139 (WhatsApp)**

**OR**

**Mail us- [bestassignment247@gmail.com](mailto:bestassignment247@gmail.com)**

**Our website -** [**www.assignmentsupport.in**](http://www.assignmentsupport.in)

**2. What is function? Discuss about different classification of user defined functions based on parameter passing and return type with examples.**

**Ans:**

User-defined functions in C can be classified based on how parameters are passed to them and what they return.

**Here are the main classifications:**

**Parameter Passing: Call by Value:** In this method, the actual value of the argument is passed to the function. Any modifications made to the parameters inside the function do not affect the original values.

**Call by Reference:** Here, the address of the argument is passed to the function. Any

**3. Explain the different kinds of loops available in C with examples.**

**Ans:**C provides several types of loops to facilitate repetitive execution of a block of code.

**The main types of loops in C are:**

For loop while loop do-while loop

**Let's discuss each of these loops with examples:**

**1. For loop:** The for loop in C is used when the number of iterations is known before the loop starts. It consists of three parts: initialization, condition, and increment/decrement.

**Set-IInd**

**4. What is the purpose of storage class in C? Explain various types of storage class in C.**

**Ans:**In C, storage classes specify the lifetime, scope, and visibility of variables and functions within a program. They determine how and where variables and functions are stored in memory and how they are accessed during program execution. Storage classes provide flexibility in managing memory and control over variable lifetimes, aiding in efficient memory utilization and program organization.

**Various Types of Storage Classes in C:**

**Auto:** The auto storage class is the default storage class for local variables. Variables

**5. Describe the process of dynamic memory allocation in C and elaborate on the different dynamic memory allocation functions in the C programming language.**

**Ans:Dynamic memory** allocation in C refers to the process of allocating memory during program execution, rather than at compile time. This allows programs to dynamically adjust memory usage based on runtime requirements. In C, dynamic memory allocation is typically performed using functions from the standard library <stdlib.h>.

**Process of Dynamic Memory Allocation in C:**

**Allocation:** Memory is allocated dynamically using functions like malloc (), calloc (), or realloc (). These functions return a pointer to the allocated memory block.

**6. (a). What is the difference between structure and union in C?**

**Ans:**

|  |  |
| --- | --- |
| **Structure** | **Union** |
| A structure can store multiple values of the different members | A union stores one value at a time for all of its members |
| A structure’s total size is the sum of the size of every data member | A union’s total size is the size of the largest data member. |
| Users can access or retrieve any member at a time. | You can access or retrieve only one member at a time |
| You can initialize multiple members at a time. | You can initialize only the first member at once. |