

[This question paper contains 4 printed pages.]

Sr. No. of Question Paper : 12860
Unique Paper Code : 2344001101
Name of the Paper : Programming Using C+
Name of the Course : Generic Elective : I
Semester : I
Duration : 3 Hours

Maximum Marks : 90

Instructions for Candidates

1. Write your Roll No. on the top immediately on receipt of this question paper.
2. **Section-A** is compulsory.
3. Attempt any four Questions from **Section B**.
4. Parts of question must be answered together.

Section-A

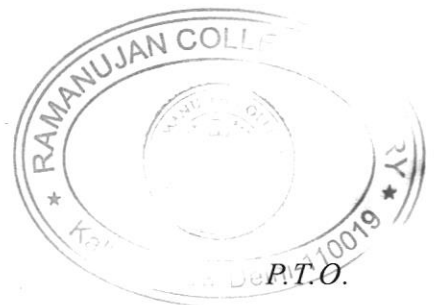
1. (a) Explain dynamic initialization of a variable with a suitable example. **3**
- (b) Differentiate with the help of an example **6**
 - (i) Static and dynamic polymorphism.
 - (ii) Break and continue statement
- (c) Find out the error(s) in following code snippets after execution giving **10** reasons:

```
(i) #include <iostream>
using namespace std;
int main()
{
    const int x = 50;
    x = 30;
    cout << x;
    return 0;
}
```

```
(ii) #include <iostream>
using namespace std;
int fun(int x, int y)
{
    return x + y;
}
double fun(int x, int y)
{
    return x * y;
}
int main()
{
    cout << fun(5,10);
    return 0;
}
```

```
(iii) #include <iostream>
using namespace std;

int main()
{
    ifstream file;
    file.open("data.txt");
    file << "Hello, World";
    file.close();
    return 3;
}
```



```
(iv) #include <iostream>
using namespace std;
int add(x, y)
{
    return x + y;
}
```

```
int main()
{
    int z = add(3, 4);
    cout << z;
    return 0;
}
```

```
(v) #include <iostream>
using namespace std;
class Student
{
public:
    Student(int age, string course)
    {
        this->age = age;
        this->course = course;
    }
    void display()
    {
        cout << "Age: " << age << ", Course: "
        << course << endl;
    }
private:
    int age;
    string course;
};
int main()
{
    Student S1;
    S1.display();

    return 0;
}
```

(d) Convert the following recursive function to iterative version:

5

```
int func(int n)
{
    if (n <= 1)
        return n;
    return func(n - 1) + func(n - 2);
}
```

(e) Write a C++ program to swap two numbers using function template.

6

Section-B

2. (a) Write a C++ program to print the following pattern for n rows: 5
 *
 **

- (b) Which of the following identifiers are valid? Give reasons. 5
 (i) total_1
 (ii) @value
 (iii) _number
 (iv) int
 (v) EmployeeID
- (c) Observe the code below and answer the question: 5

```
class A {
public:
    A() { cout << "A Constructor\n"; }
    ~A() { cout << "A Destructor\n"; }
};

class B : public A {
public:
    B() { cout << "B Constructor\n"; }
    ~B() { cout << "B Destructor\n"; }
};

int main() {
    B obj;
}
```

 (i) What will be the output?
 (ii) Explain the order of constructor and destructor execution.
3. (a) Write a while loop to display perfect square up to 100. 5
- (b) Write the output of the following code fragment after execution: 5

```
int i = 1, j = 2, r = 0;
do
{
    r = i + j;
    i++;
    if(i == 3)
        continue;
    cout << r << " ";
} while(i <= 5);
```
- (c) Write a function largest(int a, int b, int c) that finds greatest of three given numbers. Write a main function that reads three integers a, b, c from the user and uses largest to print greatest of three numbers. 5
4. (a) Write a C++ program to create a class BankAccount with member functions to deposit, withdraw and display balance. Define a default constructor for this class with 0 balance. Define functions to take input and display the balance of the customers (objects). 8

- (b) Write a C++ program that defines a template function `min` which accepts two parameters of any type and returns the minimum of the two values. Test the function with `int`, `float`, and `char` types. 7
5. (a) Write a program in C++ to create a class named `Calculator` with member functions `addition`, `subtraction`, `multiplication`, and `division`. Use objects of the class to perform arithmetic operations on two numbers. 8
- (b) Describe the concept of 'nesting of member functions'. Explain with the help of a C++ program. 7
6. (a) Design a C++ program to demonstrate multilevel inheritance using the following classes: 10
`class School (Base class) with attributes: school_name and location.`
`class Grade (Intermediate class): Inherits from School and adds attributes: grade_name and teacher.`
`class Student (derived class): inherits from Grade and adds attributes: student_name and marks.`
 Use parameterized constructors to initialize all the class attributes. Create an object of `Student`, and display details from all the classes: `School`, `Grade`, and `Student`.
- (b) Write a C++ program to print factorial of a number using recursion. 5
7. (a) Write the output of the following code: 5
- ```
#include <iostream>
using namespace std;
int main()
{
 int a, b, r;
 a = 10;
 b = 0;
 try
 {
 if (b == 0)
 throw 0;
 r = a / b;
 cout << "Result = " << r << endl;
 }
 catch (int arg)
 {
 cout << "Exception Caught" << endl;
 }
 cout << "End of the Program";
 return 0;
}
```
- (b) Write a program that reads a text file `check.txt` and prints the total number of integers in it. 6
- (c) Is runtime polymorphism possible without inheritance? Justify your answer. 4