ASSESSMENT 3

1. Write a C++ program to find the power of any number

```
#include <iostream>
#include <cmath>
using namespace std;

int main() {
    double base, exponent;
    cout << "Enter base: ";
    cin >> base;
    cout << "Enter exponent: ";
    cin >> exponent;

    double result = pow(base, exponent);
    cout << base << " raised to the power of " << exponent << " is " << result << endl;

    return 0;
}
```

Output :

```
Enter base: 2
Enter exponent: 3
2 raised to the power of 3 is 8
```

2. Write a C++ program to find the most frequent element in an array

```
#include <iostream>
#include <unordered_map>
#include <vector>
using namespace std;

int main() {
    vector<int> arr = {1, 3, 2, 3, 4, 1, 3, 2, 1, 1};
    unordered_map<int, int> freq;
    int mostFrequent = arr[0];
    int maxCount = 1;
```

```
    for (int num : arr) {
        freq[num]++;
        if (freq[num] > maxCount) {
            maxCount = freq[num];
            mostFrequent = num;
        }
    }

     cout << "The most frequent element is: " << mostFrequent << " (Frequency: " << maxCount
<< ")" << endl;
    return 0;
}
```

Output :

The most frequent element is: 1 (Frequency: 4)

3. Develop a C++ program to compute record of 10 students. Read name, regno, mark1, 2, 3 of a student, calculate the average marks and grade

```
#include <iostream>
#include <string>
using namespace std;

class Student {
public:
    string name;
    int regNo;
    int marks[3];
    double average;
    char grade;

    void calculateAverage() {
        average = (marks[0] + marks[1] + marks[2]) / 3.0;
    }

    void calculateGrade() {
```

```
        if (average > 90) {
            grade = 'S';
        } else if (average > 80) {
            grade = 'A';
        } else if (average > 70) {
            grade = 'B';
        } else if (average > 60) {
            grade = 'C';
        } else {
            grade = 'F';
        }
    }
};

int main() {
    Student students[10];
    for (int i = 0; i < 10; i++) {
        cout << "Enter details for student " << i + 1 << ":\n";
        cout << "Name: ";
        cin >> students[i].name;
        cout << "Registration Number: ";
        cin >> students[i].regNo;
        cout << "Marks in 3 subjects: ";
        cin >> students[i].marks[0] >> students[i].marks[1] >> students[i].marks[2];

        students[i].calculateAverage();
        students[i].calculateGrade();
    }

    cout << "\nStudent Records:\n";
    for (int i = 0; i < 10; i++) {
        cout << "Name: " << students[i].name << ", Reg No: " << students[i].regNo
             << ", Average Marks: " << students[i].average << ", Grade: " << students[i].grade <<
endl;
    }

    return 0;
}
```

Output :

```
Enter details for student 1:
Name: Alice
Registration Number: 101
Marks in 3 subjects: 85 90 80
...
Enter details for student 10:
Name: John
Registration Number: 110
Marks in 3 subjects: 70 75 65

Student Records:
Name: Alice, Reg No: 101, Average Marks: 85, Grade: A
...
Name: John, Reg No: 110, Average Marks: 70, Grade: B
```

4. Write a C++ program to demonstrate the use of multiple catch blocks for handling different types of exceptions

```
#include <iostream>
using namespace std;

int main() {
   try {
      int a, b;
      cout << "Enter two numbers: ";
      cin >> a >> b;
      if (b == 0)
         throw runtime_error("Division by zero");
      if (a < 0 || b < 0)
         throw invalid_argument("Negative number error");

      cout << "Result of division: " << a / b << endl;
   } catch (runtime_error &e) {
      cout << "Runtime error: " << e.what() << endl;
   } catch (invalid_argument &e) {
      cout << "Invalid argument: " << e.what() << endl;
   } catch (...) {
```

```
        cout << "Some other exception occurred" << endl;
    }

    return 0;
}
```

Output :

```
Enter two numbers: 10 0
Runtime error: Division by zero
```