Implement a program to handle exception it prints appropriate error messages, should anything go wrong. The expected behavior is defined as follows: If the compute function runs fine with the given arguments, then print the result of the function call. If it fails to allocate the memory that it needs, print Not enough memory. If any other standard C++ exception occurs, print Exception: S where S is the exception's error message. If any non-standard exception occurs, print Other Exception

#include <iostream>

#include <stdexcept> // For standard exceptions like runtime\_error

using namespace std;

// Simple function that might throw exceptions

int compute(int a, int b) {

if (a == 0) {

throw bad\_alloc(); // Simulate memory allocation failure

}

if (b == 0) {

throw runtime\_error("Division by zero"); // Simulate a standard exception

}

if (a == -1) {

throw "Non-standard exception"; // Simulate a non-standard exception

}

return a / b;

}

int main() {

int a, b;

cout << "Enter two integers: ";

cin >> a >> b;

try {

int result = compute(a, b);

cout << "Result: " << result << endl;

}

catch (const bad\_alloc&) {

cout << "Not enough memory" << endl; // Handle memory allocation failure

}

catch (const runtime\_error& e) {

cout << "Exception: " << e.what() << endl; // Handle standard exceptions with error message

}

catch (...) {

cout << "Other Exception" << endl; // Catch any other non-standard exception

}

return 0; }

Write a C++ Program for exception handling if age is greater than or equal to 18 then it display message access granted you are old enough otherwise trow an exception.

#include<iostream>  
using namespace std;

Int main() {

try {  
  int age = 15;  
  if (age >= 18) {  
    cout << "Access granted - you are old enough.";  
  } else {  
    throw (age);  
  } }

//variable myNum holds the value of the thrown exception, which is age   
catch (int myNum) {  
  cout << "Access denied - You must be at least 18 years old.\n";  
  cout << "Age is: " << myNum;  
} return 0; }