**Assignment 05 Exception**

**1.** Develop a simple calculator program that performs basic arithmetic operations (+, -, \*, /)

on two numbers provided by the user. The program should ask the user for the numbers

and the operator. However, the program should handle the following exceptions:

a. Invalid Number: If the user enters a number that is not valid, catch the exception and

display an error message.

b. Invalid Operator: If the user enters an operator other than "+", "-", "\*", or "/", catch the

exception and display an error message.

c. Division by Zero: If the user tries to divide by zero, catch the exception and display an

error message.

Write a program that performs the requested arithmetic operation and handles the

exceptions as described above.

Calculator.h

#ifndef CALCULATOR\_H

#define CALCULATOR\_H

using *namespace* std;

*double* calculate(*double* *num1*, *double* *num2*, *char* *op*);

#endif

Calculator.cpp

#include "calculator.h"

*double* calculate(*double* *num1*, *double* *num2*, *char* *op*) {

    switch (*op*) {

    case '+':

        return *num1* + *num2*;

    case '-':

        return *num1* - *num2*;

    case '\*':

        return *num1* \* *num2*;

    case '/':

        if (*num2* == 0) {

            throw "Division by zero is not allowed!";

        }

        return *num1* / *num2*;

    default:

        throw "Invalid operator!";

    }

}

Main.cpp

#include <iostream>

#include "calculator.h"

using *namespace* std;

*int* main() {

*double* num1, num2;

*char* op;

*int* x = 1;

    while (x > 0) {

        cout << "Do you want to stop {0 to stop}: ";

        cin >> x;

        try {

            if (x == 0) {

                cout << "Exiting program." << endl;

                break;

            }

            cout << "Enter the first number: ";

            if (!(cin >> num1)) {

                throw "Invalid number entered!";

            }

            cout << "Enter an operator (+, -, \*, /): ";

            cin >> op;

            if (op != '+' && op != '-' && op != '\*' && op != '/') {

                throw "Invalid operator entered!";

            }

            cout << "Enter the second number: ";

            if (!(cin >> num2)) {

                throw "Invalid number entered!";

            }

*double* result = calculate(num1, num2, op);

            cout << "The result is: " << result << endl;

        } catch (const *char* \*errMsg) {

            cout << "Error: " << errMsg << endl;

        } catch (...) {

            cout << "An unknown error occurred." << endl;

        }

    }

    return 0;

}

Output:

PS D:\Fullstack-Java-FirstBit-Solutions\DSA\Assignments\Exception\Calculator> ./main

Do you want to stop {0 to stop}: 1

Enter the first number: 23

Enter an operator (+, -, \*, /): +

Enter the second number: 32

The result is: 55

Do you want to stop {0 to stop}: 1

Enter the first number: 45

Enter an operator (+, -, \*, /): -

Enter the second number: 56

The result is: -11

Do you want to stop {0 to stop}: 1

Enter the first number: 35

Enter an operator (+, -, \*, /): \*8

Enter the second number: The result is: 280

Do you want to stop {0 to stop}: 1

Enter the first number: 23

Enter an operator (+, -, \*, /): /

Enter the second number: 6

The result is: 3.83333

Do you want to stop {0 to stop}: 1

Enter the first number: 34

Enter an operator (+, -, \*, /): /

Enter the second number: 0

Error: Division by zero is not allowed!

Do you want to stop {0 to stop}: 1

Enter the first number: 6

Enter an operator (+, -, \*, /): f

Error: Invalid operator entered!

Do you want to stop {0 to stop}: 0

Exiting program.

PS D:\Fullstack-Java-FirstBit-Solutions\DSA\Assignments\Exception\Calculator>

**2.** Create class television that has members to hold the model number ,screen size and price.

Take a member function to take input from user, If more than 4 digits are entered for model

number, if screen size is smaller than 12 inches or greater than 70 inches or if the price is

negative or greater than 5000 Rs, then throw an exception.

Write a main() that instantiates an object and allows the user to enter and display data. If

exception is caught, replace all data member values with zero

Television.h

#include <stdexcept>

#include <iostream>

using *namespace* std;

*class* Television {

*private:*

*int* modelNumber;

*int* screenSize;

*float* price;

*int* countDigits(*int* *number*);

*public:*

    Television();

*void* inputDetails();

*void* displayDetails() const;

};

Television.cpp

#include "television.h"

Television::Television() : modelNumber(0), screenSize(0), price(0.0) {}

*int* Television::countDigits(*int* *number*) {

*int* count = 0;

    while (*number* != 0) {

        count++;

*number* /= 10;

    }

    return count;

}

*void* Television::inputDetails() {

    try {

        cout << "Enter Model Number: ";

        cin >> modelNumber;

        if (countDigits(modelNumber) > 4) {

            throw invalid\_argument("Model number must not exceed 4 digits.");

        }

        cout << "Enter Screen Size (in inches): ";

        cin >> screenSize;

        if (screenSize < 12 || screenSize > 70) {

            throw invalid\_argument("Screen size must be between 12 and 70 inches.");

        }

        cout << "Enter Price (in Rs): ";

        cin >> price;

        if (price < 0 || price > 5000) {

            throw invalid\_argument("Price must be between 0 and 5000 Rs.");

        }

    } catch (const invalid\_argument &e) {

        cout << "Error: " << e.what() << endl;

        modelNumber = 0;

        screenSize = 0;

        price = 0.0;

        throw;

    }

}

*void* Television::displayDetails() const {

    cout << "\nTelevision Details:" << endl;

    cout << "Model Number: " << modelNumber << endl;

    cout << "Screen Size: " << screenSize << " inches" << endl;

    cout << "Price: Rs " << price << endl;

}

Main.cpp

#include "television.h"

*int* main() {

    Television tv;

    try {

        tv.inputDetails();

        tv.displayDetails();

    } catch (...) {

        cout << "\nException caught. Setting all values to zero." << endl;

        tv.displayDetails();

    }

    return 0;

}

Output:

PS D:\Fullstack-Java-FirstBit-Solutions\DSA\Assignments\Exception\Television> ./main

Enter Model Number: 345

Enter Screen Size (in inches): 23

Enter Price (in Rs): 5433

Error: Price must be between 0 and 5000 Rs.

Exception caught. Setting all values to zero.

Television Details:

Model Number: 0

Screen Size: 0 inches

Price: Rs 0

PS D:\Fullstack-Java-FirstBit-Solutions\DSA\Assignments\Exception\Television> ./main

Enter Model Number: 4567

Enter Screen Size (in inches): 34

Enter Price (in Rs): 2310

Television Details:

Model Number: 4567

Screen Size: 34 inches

Price: Rs 2310

PS D:\Fullstack-Java-FirstBit-Solutions\DSA\Assignments\Exception\Television>