Assignment – 30 A Job Ready Bootcamp in C++, DSA and IOT MySirG

Exception Handling

1. Write a C++ program to demonstrate the use of try, catch block with the argument as an

integer and string using multiple catch blocks.

#include <iostream>

using namespace std;

void test\_try(int num)

{

try

{

if (num >= 0 && num <= 9)

throw num;

else

cout << "\nIt is not a Single Number\n";

throw;

}

catch (int a)

{

cout << "\nIt is a Single Number\n";

}

catch (char b[100])

{

cout << b << endl;

}

}

int main()

{

int num;

cout << "Enter a number: ";

cin >> num;

test\_try(num);

return 0;

}

2. Write a C++ program to demonstrate try, throw and catch statements.

#include <iostream>

using namespace std;

int main()

{

try

{

throw 'E';

}

catch (char e)

{

cout << "\nException Caught " << e << " " << endl;

}

return 0;

}

3. Write a C++ program to perform arithmetic operations on two numbers and throw an

exception if the dividend is zero or does not contain an operator.

#include <iostream>

using namespace std;

int main()

{

char Operator;

float num1, num2, ans;

cout << "\nPerform Arithmetic Operations on Two Numbers";

cout << "\n---------------------------------------------";

try

{

cout << "\nEnter First Number: ";

cin >> num1;

cout << "\nEnter operator: ";

cin >> Operator;

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

throw Operator;

cout << "Enter the second number: ";

cin >> num2;

if (num1 == 0)

throw 0;

cout << "\n-------------------------------------------------------\n";

switch (Operator)

{

case '+':

ans = num1 + num2;

break;

case '-':

ans = num1 - num2;

break;

case '\*':

ans = num1 \* num2;

break;

case '/':

ans = num1 / num2;

break;

default:

break;

}

cout << "\nAnswer : " << num1 << " " << Operator << " " << num2 << " = " << ans << endl;

}

catch (const char c)

{

cout << "\nExcepiton Caught...\nWrong Operator : " << c << " is not a valid operator.." << endl;

}

catch (const int x)

{

cout << "\nError!, Wrong Operator.." << endl;

}

return 0;

}

4. Write a C++ program to accept an email address and throw an exception if it does not

contain @ symbol.

#include <iostream>

using namespace std;

bool isEmailValid(string email)

{

int atFlag = -1, dotFlag = -1;

int length = 0;

for (int i = 0; email[i] != '\0'; i++)

{

if (email[i] == '@')

atFlag = i;

if (email[i] == '.')

dotFlag = i;

length++;

}

if (atFlag == -1 && dotFlag == -1)

return 0;

else if (atFlag > dotFlag)

return 0;

else

return 1;

}

int main()

{

string email;

cout << "Enter Email Id: ";

cin >> email;

try

{

if (isEmailValid(email))

cout << "Email is valid..";

else

throw 'c';

}

catch (const char e)

{

cout << "\nException caught...!, Invalid email id..!";

}

return 0;

}

5. Write a C++ program to accept a mobile number and throw an exception if it does not

contain 10 digits.

#include <iostream>

#include <string.h>

using namespace std;

bool isValidNumber(string m)

{

int i;

for (i = 0; m[i] != '\0'; i++)

;

if (i == 10)

return true;

return false;

}

int main()

{

string mobileNum;

cout<<"Enter Mobile Number: ";

cin >> mobileNum;

try

{

if (isValidNumber(mobileNum))

cout << "\nValid Mobile Number..." << endl;

else

throw 1;

}

catch (int n)

{

cout << "\nException Caught...\n Invalid Mobile Number" << endl;

}

return 0;

}

6. Write a C++ program to accept area pin code and throw an exception if it does not

contain 6 digits.

#include <iostream>

#include <string.h>

using namespace std;

bool isValidPinCode(string m)

{

int i;

for (i = 0; m[i] != '\0'; i++)

{

if (!(m[i] >= '0' && m[i] <= '9'))

return false;

}

if (i == 6)

return true;

return false;

}

int main()

{

string pinCode;

cout << "Enter Area Pincode: ";

cin >> pinCode;

try

{

if (isValidPinCode(pinCode))

cout << "\nValid area pincode..." << endl;

else

throw 1;

}

catch (int n)

{

cout << "\nException Caught...\n Invalid area pincode" << endl;

}

return 0;

}

7. Write a C++ program to accept a username if the username has less than 6 characters

or does contain any digit or special symbol.

#include <iostream>

using namespace std;

bool ifUsernameValid(string userName)

{

int digit = -1, specialSymbol = -1, character = -1;

int length = 0;

for (int i = 0; userName[i] != '\0'; i++)

{

if (userName[i] >= '0' && userName[i] <= '9')

digit = 1;

else if ((userName[i] >= 'a' && userName[i] <= 'z') || (userName[i] >= 'A' && userName[i] <= 'Z'))

character = 1;

else

specialSymbol = 1;

length++;

}

if (length < 6 || digit == -1 || character == -1 || specialSymbol == -1)

return false;

return true;

}

int main()

{

string userName;

cout << "Enter user name: ";

cin >> userName;

try

{

if (ifUsernameValid(userName))

cout << "\nUserName is valid..";

else

throw 'e';

}

catch (char e)

{

cout << "\nExeception occurr. " << e << ", Invalid username";

}

return 0;

}

8. Write a C++ program to accept a password and throw an exception if the password has

less than 6 characters or does not contain a digit or does not contain any special

character or does not contain any capital letter.

#include <iostream>

using namespace std;

bool ifUsernameValid(string userName)

{

int digit = -1, specialSymbol = -1, character = -1;

int length = 0;

for (int i = 0; userName[i] != '\0'; i++)

{

if (userName[i] >= '0' && userName[i] <= '9')

digit = 1;

else if ((userName[i] >= 'a' && userName[i] <= 'z') || (userName[i] >= 'A' && userName[i] <= 'Z'))

character = 1;

else

specialSymbol = 1;

length++;

}

if (length < 6 || digit == -1 || character == -1 || specialSymbol == -1)

return false;

return true;

}

int main()

{

string userName;

cout << "Enter user name: ";

cin >> userName;

try

{

if (ifUsernameValid(userName))

cout << "\nUserName is valid..";

else

throw 'e';

}

catch (char e)

{

cout << "\nExeception occurr. " << e << ", Invalid username";

}

return 0;

}

9. Write a C++ program to accept Gmail id only and throw an exception if the id does not

contain @ and gmail.com.

#include <iostream>

using namespace std;

bool isValidGamilId(string gmailId)

{

int atContain = -1, gmailContain = -1;

int i;

if (gmailId.find("@gmail.com") != -1) // this funtion find substring in gmaild

return true;

return false;

}

int main()

{

string gmailId;

cout << "Enter Gemail Id - ";

cin >> gmailId;

try

{

if (isValidGamilId(gmailId))

cout << "\nValid gmail id..";

else

throw 'e';

}

catch (char e)

{

cout << "\nException occurr... " << e << " \nInvalid gmaild id";

}

return 0;

}

10. Write a C++ program to accept Nickname and throw an exception if it has greater than 8

characters or does contain a digit or special symbol or space.

#include <iostream>

using namespace std;

bool isValidNickName(string nickName)

{

int i;

for (i = 0; nickName[i] != '\0'; i++)

{

if (nickName[i] >= '0' && nickName[i] <= '9')

return false;

else if (nickName[i] == ' ')

return false;

else if (!((nickName[i] >= 'a' && nickName[i] <= 'z') || (nickName[i] >= 'A' && nickName[i] <= 'Z')))

return false; // this is for special symbol

}

if (i > 8)

return false;

return true;

}

int main()

{

string nickName;

cout << "Enter NickName: ";

cin >> nickName;

try

{

if (isValidNickName(nickName))

cout << "\nValid nickname.." << endl;

else

throw 'e';

}

catch (char e)

{

cout << "\nException occur..." << e << "\nInvalid nickName" << endl;

}

return 0;

}