Topic: **In-Lab Practice Session 13**
Course: **CSE1002 – Problem Solving & Object-Oriented Programming**
Date: **21st April, 2021**

IPS_ExceptionHandling-1

A police camp is recruiting the trainees by measuring the parameters such as height, age, and chest. The minimum eligibility for selection is based on the following conditions:

-Height should be between 5.2 to 5.6 inches

-Age should be greater than 18 and less than 25

-Chest should be greater than 45 cm

Write an inline function to find whether the person is fit for the police or not. Print “Selected“ when criteria is satisfied and print “Not selected” when criteria are not satisfied. Raise an exception when age is entered negative. Print “Negative age” when the exception raises.

Input Format:

Read height

Read age

Read chest

Output Format:

If the person is fit then print 'Selected'

Otherwise, print 'Not selected'

Boundary Conditions:

All input >0

Topic: **In-Lab Practice Session 13**
Course: **CSE1002 – Problem Solving & Object-Oriented Programming**
Date: **21st April, 2021**

IPS_ExceptionHandling-2

A sample of nitrogen gas occupies a volume of 'v1' cubic units at 'p1' mmHg (unit of pressure is mmHg) and 't1' °C. The volume increases by 'v2' cubic units and the temperature decreases to 't2' °C. Write a C++ program to compute the final pressure exerted on the gas. Use the equation p1v1/t1 = p2v2/t2, for your computation. If any of the value in the denominator of the equation is zero then throw an exception that prints 'zero not allowed' and do not perform the computation.

Note: Syntax to print 'x' decimal places of variable 'a'

include <iomanip>

use

cout<<fixed<<setprecision(x)<<a;

Input format

Volume v1 in cubic units

Pressure p1 in mmHg

Initial temperature t1 in ° C

Final volume v2 in L

Final temperature t2 in ° C

Output Format

Represent the output in two decimal points

Final pressure p2 in mmHg

Topic: **In-Lab Practice Session 13**
Course: **CSE1002 – Problem Solving & Object-Oriented Programming**
Date: **21st April, 2021**

KVP_ExceptionHandlers-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.

Exception handling means handling of abnormal or unexpected events.

Exception handling is done by 'try', 'catch', 'throw' keywords.

It transfers the control to special functions called Handlers and makes it easy to separate the error handling code.

Input :

First line : Read a Number

Output :

if Number possitive and a Single Digit, throw an exception to print "Single Digit Number", else through an Exception "Not a Single Digit Number" . if Number is negative, then throw an exception "Negative Number"

Topic: **In-Lab Practice Session 13**
Course: **CSE1002 – Problem Solving & Object-Oriented Programming**
Date: **21st April, 2021**

KVP_ExceptionHandlers-2

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.

case - 1 : if first number is 0, then throw an exception "Bad Operation"

case - 2 : if operator is other than '+', '-', '*', '/' then throw an exception "Not a Valid Operator"

case - 3 : if second number is 0, then throw an exception "Divide by Zero"

case - 4 : if operator is a number, then throw an exception "Exception Caught"

Input :

Read First Number (Num1)

Read an Operator (Op)

Read Second Number(Num2)

Output :

Result (Num1 Op Num2 : - 23 + 20 = 43 or 23 / 0 then throw exception)