LAB 12

| Question | Task | TIME ALLOCATION | REMARKs |
| --- | --- | --- | --- |
| 1, 2 | Single inheritance | 70 minutes |  |
| 3 | Single inheritance | 50 minutes |  |

**Question 1**

Create a complete program based on the following criterias:

A) Create class **Triangle**.

1. protected data members : a(int), b(int), c(int)
2. public member functions

* **Default constructor**

To output *“-------- Triangle Class -------“.*

* **void setData(………)**To set the data member **a and b** with the value of its arguments.

B) Create class **PythogorasTriangle** which inherits **publicly** from class **Triangle.**

1. private data members : none
2. public member functions

* **Default constructor**

To output *“-------Pythogoras Triangle -------“.*

* **void calcHypotenuse()**

To calculate the hypotenuse which is ***c*** using the formula given below

(Hint: Use the functions in the math library (#include<cmath>)

Square root function : sqrt(<a number>)

Power function : pow(<a number>, 2) )

* **void displaySides()**

Display values of a, b and c.

C) In **main()**

* Create an object of class PythogorasTriangle using the new operator.
* Get user input in the main function for 2 sides(side a and side b) of a Pythogoras Triangle and pass it to the ***setData(..)*** function as arguments. Invoke the setData(..) function through the dynamic object .
* Using the same object , call ***calcHypotenuse()*** and ***displaySides().***
* Deallocate the memory for the object.

**[Note: refer to sample output screens below]**

|  |
| --- |
| **Sample Output Screen** |
| ----- Triangle class -------  -----Pythogoras Triangle------  Enter a : ***5***  Enter b : ***12***  ::The sides of a pythogoras triangle::  a :5  b :12  c :13(also known as the Hypotenuse) |

**Question 2**

Modify main function of question 1 solution so that the program will repeat as long as user wants to. Use a *do-while* loop.

**[Note: refer to sample output screens below]**

|  |
| --- |
| **Sample Output Screen** |
| ----- Triangle class -------  -----Pythogoras Triangle------  Enter a : ***5***  Enter b : ***12***  ::The sides of a pythogoras triangle::  a :5  b :12  c :13(also known as the Hypotenuse)  Do you want to continue [Y/N] : ***Y***  ----- Triangle class -------  -----Pythogoras Triangle------  Enter a : ***3***  Enter b : ***4***  ::The sides of a pythogoras triangle::  a :3  b :4  c :5(also known as the Hypotenuse)  Do you want to continue [Y/N]: ***N*** |

**Question 3**

Create a complete program based on the following criterias :

A) Create class **Product**

1. private data member : prodID (int)
2. protected data member: total\_price(float)
3. public member functions :

* **constructor** to initialize total\_price to ***0.0***
* **int getProd\_ID()**

Returns prodID

* **void input\_ProdID()**

Get user input for prodID (representing product ID)

B) Create class **CanFood** which inherits **protectedly** from **class Product.**

(i) private data members : unit\_price(float), order\_unit(int)

(ii) public member functions

* **void get\_Product()**

Call ***input\_prodID( ).*** Get user input for unit\_price and order\_unit .

* **void calculate\_Total()**   
  Calculate total\_price
* **void display\_product()**Display prodID by calling accessor function. Display total\_price.

C) In main()

* Prompt the user to enter the number of types of canned food.
* Create pointer ***tp*** of class *canFood*
* Use the pointer ***tp*** to create a dynamic array of *canFood* using new operator. (the size of the array will be the number of types of canned food entered by the user earlier)
* In a for-loop that loop through the array, using pointer *tp*:
  + Call *get\_Product( )*
  + Call *calculate\_Total( )*
* In another for-loop that loop through the array, using pointer *tp*:
  + Call *display\_product( )*
* Deallocate the memory of the dynamic array.

|  |
| --- |
| **Sample Output Screen** |
| How many types of canned food? ***2***  Enter product ID : ***1001***  Enter price : RM***30***  Enter order unit : ***5***  Enter product ID : ***1002***  Enter price : RM***45***  Enter order unit : ***3***  Product ID : 1001  Total price : 150  Product ID : 1002  Total price : 135 |