建立兩個類別square\_2D與rect\_3D，分別為基本類別與衍生類別(也就是rect\_3D繼承於square\_2D)，輸入長、寬、高，分別印出四方形面積與長方體的體積與表面積。

(60 points)

|  |
| --- |
| Square\_2D |
| double length,  width  compute\_area() |

|  |
| --- |
| Square\_2D |
| double length,  width  compute\_area() |

|  |
| --- |
| Is-a |

|  |
| --- |
| Rect\_3D |
| double height  compute\_volume()  compute\_surface() |

(90 max)

20210616建立兩個類別square\_2D與rect\_3D，分別為基本類別與衍生類別(也就是rect\_3D繼承於square\_2D)，輸入長、寬、高，分別印出四方形面積與長方體的體積與表面積。

|  |
| --- |
| Square\_2D |
| double length,  width  compute\_area() |

|  |
| --- |
| Rect\_3D |
| double height  compute\_volume()  compute\_surface() |

※square\_2D.h檔

/\*這次程式的重點就是在運用類別的「繼承(inherit)」關係\*/

/\*「繼承」的定義->先定義基礎類別(base class)，可以依據定義的基礎類別，再定義一個包含基礎類別內的運算子.函數等，且可以再賦予新函數與運算子的衍生類別(derived class)\*/

/\*例如本題就是利用square\_2D當作基礎類別，再自行定義一個rect\_3D的衍生類別\*/

#ifndef square2d\_H

#define square2d\_H

using namespace std;

//square\_2d的類別(包含6個成員函數與2個資料成員)

class square\_2d{

public:

square\_2d(double=0,double=0); //建構子

void setlength(const double); //設定長度

void setwidth(const double); //設定寬度

double getlength(); //取得長度

double getwidth(); //取得寬度

double compute\_area() const; //計算面積

double length; //長

double width; //寬

};

#endif

※square\_2D.cpp檔(類別實作檔)

#include<iostream>

#include<stdexcept>

#include"square\_2D.h"

using namespace std;

square\_2d::square\_2d(const double l, const double w)

{

setlength(l);

setwidth(w);

}

void square\_2d::setlength(const double l){

length = l;

}

void square\_2d::setwidth(const double w){

width = w;

}

double square\_2d::getlength(){

return length;

}

double square\_2d::getwidth(){

return width;

}

//面積計算：長\*寬

double square\_2d::compute\_area() const{

return length\*width;

}

※rect\_3D.h檔

#ifndef square3d\_H

#define square3d\_H

#include"square\_2D.h"

using namespace std;

/\*繼承類別的宣告方法為「衍生類別：基礎類別」\*/

//繼承square\_2d類別的rect\_3d類別(包含5個成員函數與1個資料成員 + square\_2d內的函數與資料成員)

class rect\_3d:public square\_2d

{

public:

rect\_3d(double = 0, double = 0, double = 0); //建構子

void setheight(const double); //設定高度

double getheight(); //取得高度

double compute\_volume() const; //計算體積

double compute\_surface() const; //計算表面積

double height; //高

};

#endif

※rect\_3D.cpp檔(類別實作檔)

#include<iostream>

#include<stdexcept>

#include"rect\_3D.h"

using namespace std;

rect\_3d::rect\_3d(double l, double w, double h)

{

setlength(l);

setwidth(w);

setheight(h);

}

void rect\_3d::setheight(const double h){

height = h;

}

double rect\_3d::getheight(){

return height;

}

//體積計算：長\*寬\*高

double rect\_3d::compute\_volume() const{

return length \* width \* height;

}

//表面積計算：(長\*寬+寬\*高+長\*高)\*2

double rect\_3d::compute\_surface() const{

return (length \* width +width\*height+length\*height)\*2;

}

※square\_rect(main).cpp檔(主程式檔)

#include<iostream>

#include<stdexcept>

#include"square\_2D.h"

#include"rect\_3D.h"

using namespace std;

int main(void){

double length;

double width;

double height;

//視窗輸出配置

cout <<"Enter the length and width for square："<< endl;

cout << " Length=";

cin >> length;

cout << " Width =";

cin >> width;

square\_2d userenter(length, width);

cout <<"-------------------------------------------------"<< endl;

cout << "Enter the height for rect:"<< endl;

cout << " Height=";

cin >> height;

rect\_3d userenter2(length, width, height);

cout << "-------------------------------------------------" << endl;

cout << "The result is:\n\n";

cout << " The result of square:\n";

cout << " Length：" << userenter.getlength() << " cm" << " | Width："

<< userenter.getwidth() << " cm"<< "\n The area is "

<< userenter.compute\_area() <<" cm2\n"<< endl;

cout << " The result of rect:\n";

cout << " Length：" << userenter2.getlength() << " cm" << " | Width："

<< userenter2.getwidth() << " cm" << " | Height：" << userenter2.getheight()

<< " cm" << "\n The volume is "<< userenter2.compute\_volume() << " cm3 | The surface is " << userenter2.compute\_surface()<<" cm2\n"<< endl;

system("pause");

}