Homework 004

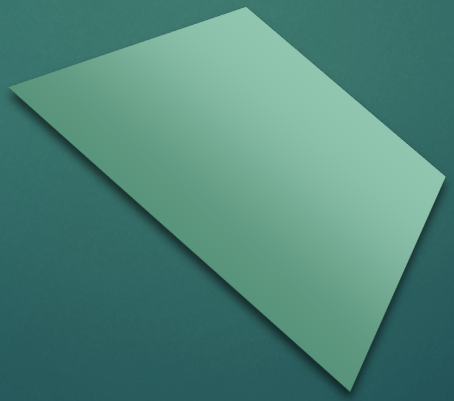
THE QUADRILATERALS

2021/4/24 SAT. 12:00 NOON DUE

2021/5/1 SAT. 23.59 為最後補交期限

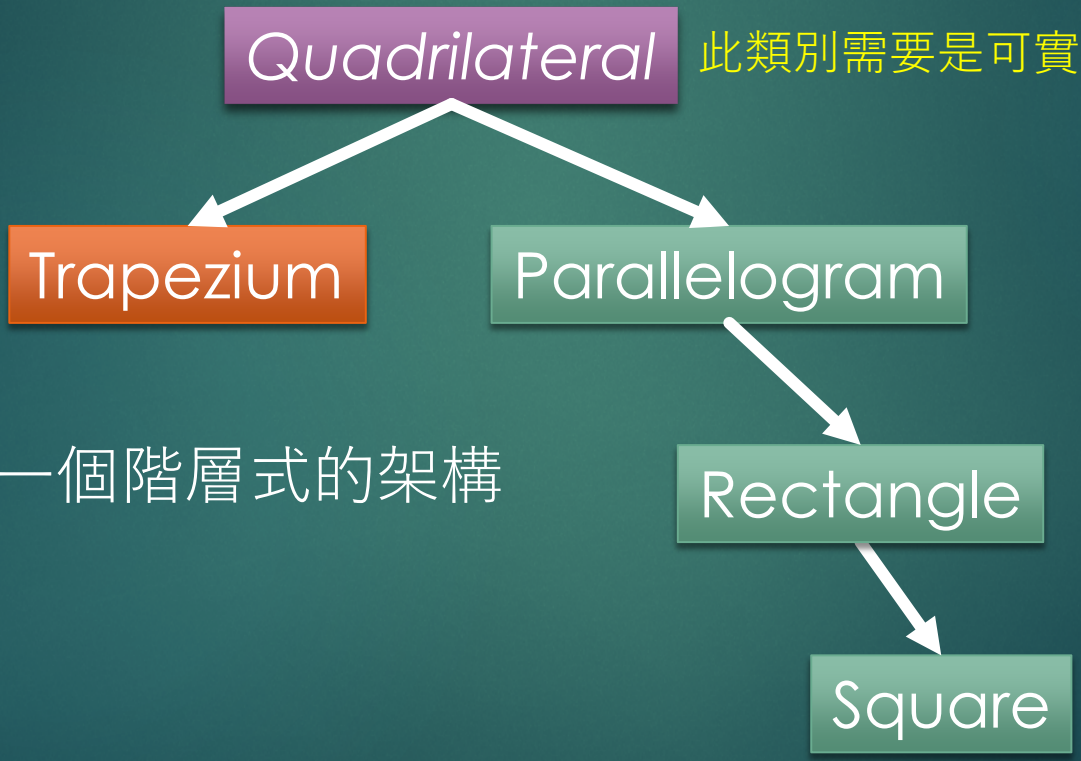
The Quadrilaterals

凹四邊形
也是四邊形



複雜多邊形不再本次作業範圍內

Build a family classes of Quadrilateral



建立一個階層式的架構

Provide following functions for these classes

R

提示：有些只要實作在Base class即可

- Provide these public functions
  - Print information of this shape: 印出資訊
    - void **print()**, **operator<<**
  - Return the TYPE NAME of this shape: 印出類型
    - string **getType()**
  - Calculate area of this shape: 計算面積
    - double **getArea()**
  - Validate its type:
    - bool **isValid()**

檢查形狀是否正確

- 比方說矩形是否都是直角
- 梯形是否只有一邊平行

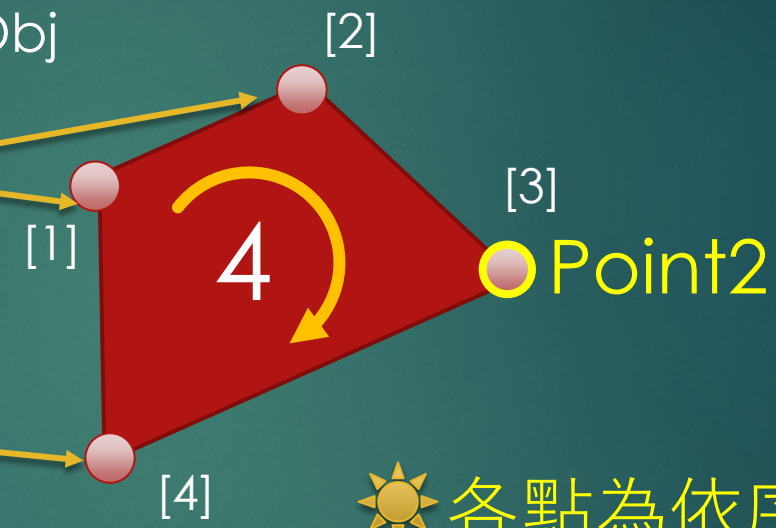
operator<<

R

- cout << TrapeziumObj

(0, 3)
(3, 4)
(6, 2)
(0, 0)

數字僅供參考



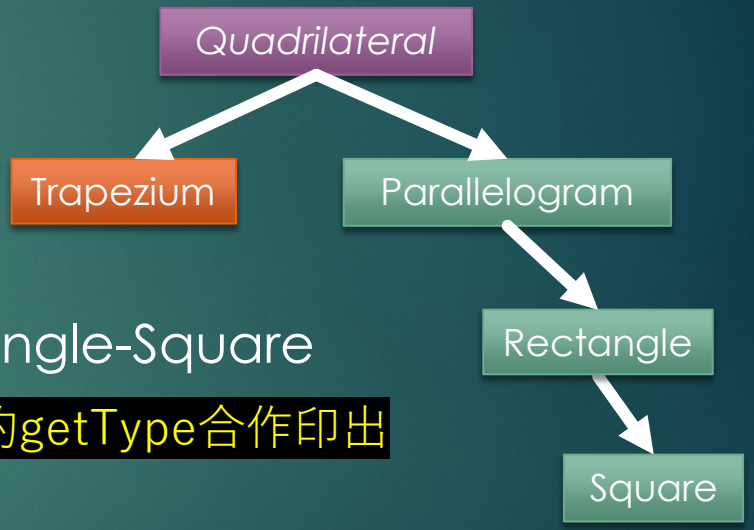
各點為依序輸入
(順時針方向)

getType()

R

- Quadrilateral.getType()

Quadrilateral



- Square.getType()

Quadrilateral-Parallelogram-Rectangle-Square

請注意！上面這個字串必須由所有父類別的getType合作印出

- Trapezium.getType()

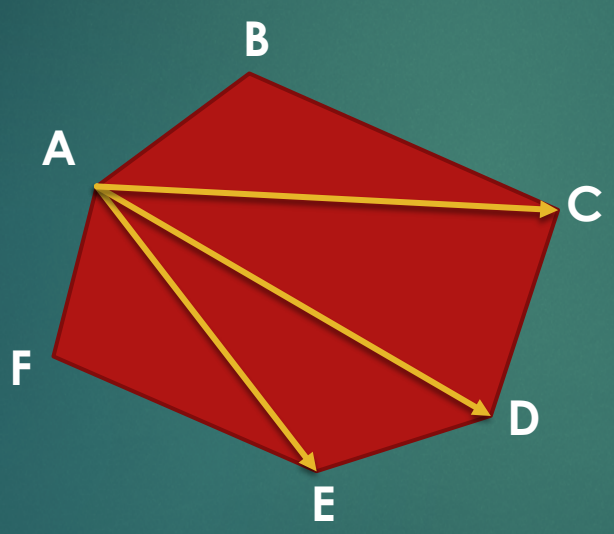
Quadrilateral-Trapezium

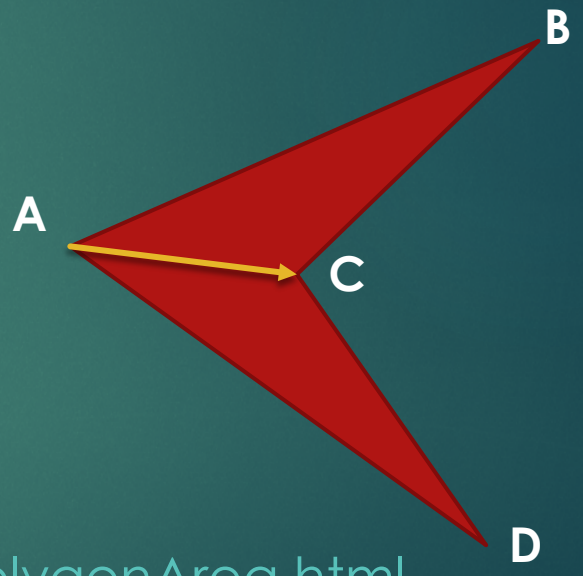
Reference: http://www.cplusplus.com/reference/string/

getArea()

R

- General solution – dived the shape into multiple triangles





Trick:

http://mathworld.wolfram.com/PolygonArea.html

(BONUS) 不同類型的四邊型，採用不同算法來加快計算面積的速度可獲得加分

isValid()

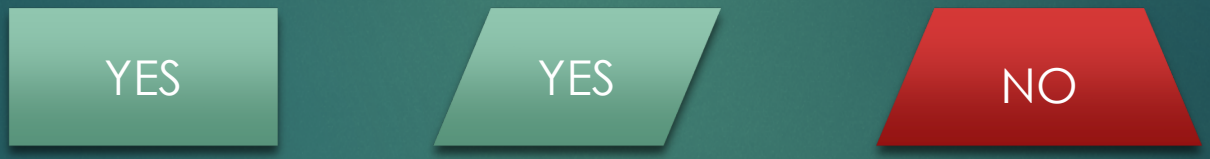
檢查形狀是否正確

- 比方說是否為矩形?
- 是否為平行四邊形?

R

- Is it a square?

- Is it a parallelogram?



請注意設計！必須使用父類別協助判斷（評分項目）

- 比方說如果要判斷一個形狀是不是矩形，將由：
- Quadrilaterals 判斷是不是四邊形 （四個不重疊、不共線的點）
- Parallelogram 判斷對邊是否平行
- 最後才由 Rectangle判斷是否相鄰邊為垂直

Build a helper class Point2 and helper functions

R

- Use this class as private data member of *Quadrilaterals*
- Has 2 private data member (at least)
  - for example int x, y;

    測試資料採用整數

    注意做除法運算前先轉浮點數

Build a helper class Point2 and helper functions

R

Provide following functions

- Constructor of a point, for example:
  - POINT2(int, int)
- Calculate distance between two Point2, for example:
  - static float getDistance(const Point2&, const Point2&);
- Is $\overline{AB}$ // $\overline{CD}$?
  - static bool isParallel(const Point2 &A, const &Point2 &B, const Point2 &C, const Point2 &D)

```cpp
int main(int argc, const char * argv[]) {
    int x1, y1, x2, y2, x3, y3, x4, y4, choice;
    myQuadrilateral* current = 0;

    while (true) {
        cout << "(1) Quadrilateral (default)" << endl;
        cout << "(2) Rectangle" << endl;
        cout << "(3) Parallelogram" << endl;
        cout << "(4) Square" << endl;
        cout << "(5) Trapezium" << endl;
        cout << "Choose one kind of shape(1~5)" << endl;
        cin >> choice;
        cout << "Input 4 point [x1] [y1] ... [x4] [y4]" << endl;
        cin >> x1 >> y1 >> x2 >> y2 >> x3 >> y3 >> x4 >> y4;

        switch (choice) {
            case 1:
                current = new myQuadrilateral(x1, y1, x2, y2, x3, y3, x4, y4);
                break;
            case 2:
                current = new myRectangle(x1, y1, x2, y2, x3, y3, x4, y4);
                break;
            case 3:
                current = new myParallelogram(x1, y1, x2, y2, x3, y3, x4, y4);
                break;
            case 4:
                current = new mySquare(x1, y1, x2, y2, x3, y3, x4, y4);
                break;
            case 5:
                current = new myTrapezium(x1, y1, x2, y2, x3, y3, x4, y4);
                break;

            default:
                current = new myQuadrilateral(x1, y1, x2, y2, x3, y3, x4, y4);
                break;
        }

        if(current->isValid()){
            cout << "the input shape is valid and the type is : " << endl;
            cout << *current << endl;
            cout << "Area of this shape is = " << current->getArea() << endl;
        }
        else {
            cout << "Fail to pass the validation test of : " << endl;
            cout << current->getType() << endl;
        }

        delete current;
        system("pause");
    }
```

R

測試環境為
一個交互式的系統

系統首先詢問使用者要輸入的形狀類型

接著讓使用者輸入四個點的X與Y

最後輸出形狀驗證結果
(1) 形狀正確：計算面積
(2) 形狀不正確：回報錯誤

```
8  int main(int argc, const char * argv[]) {
9      int x1, y1, x2, y2, x3, y3, x4, y4, choice;
10     myQuadrilateral* current = 0;
11
12     while (true) {
13         cout << "(1) Quadrilateral (default)" << endl;
14         cout << "(2) Rectangle" << endl;
15         cout << "(3) Parallelogram" << endl;
16         cout << "(4) Square" << endl;
17         cout << "(5) Trapezium" << endl;
18         cout << "Choose one kind of shape(1~5)" << endl;
19         cin >> choice;
20         cout << "Input 4 point [x1] [y1] ... [x4] [y4]" << endl;
21         cin >> x1 >> y1 >> x2 >> y2 >> x3 >> y3 >> x4 >> y4;
22
23         switch (choice) {
24             case 1:
25                 current = new myQuadrilateral(x1, y1, x2, y2, x3, y3, x4, y4);
26                 break;
27             case 2:
28                 current = new myRectangle(x1, y1, x2, y2, x3, y3, x4, y4);
29                 break;
30             case 3:
31                 current = new myParallelogram(x1, y1, x2, y2, x3, y3, x4, y4);
32                 break;
33             case 4:
34                 current = new mySquare(x1, y1, x2, y2, x3, y3, x4, y4);
35                 break;
36             case 5:
37                 current = new myTrapezium(x1, y1, x2, y2, x3, y3, x4, y4);
```

父類別指標

動態生成物件

```
28              current = new myRectangle(x1, y1, x2, y2, x3, y3, x4, y4);
29              break;
30          case 3:
31              current = new myParallelogram(x1, y1, x2, y2, x3, y3, x4, y4);
32              break;
33          case 4:
34              current = new mySquare(x1, y1, x2, y2, x3, y3, x4, y4);
35              break;
36          case 5:
37              current = new myTrapezium(x1, y1, x2, y2, x3, y3, x4, y4);
38              break;
39
40          default:
41              current = new myQuadrilateral(x1, y1, x2, y2, x3, y3, x4, y4);
42              break;
43      }
44
45      if(current->isValid()){
46          cout << "the input shape is valid and the type is : " << endl;
47          cout << *current << endl;
48          cout << "Area of this shape is = " << current->getArea() << endl;
49      }
50      else {
51          cout << "Fail to pass the validation test of : " << endl;
52          cout << current->getType() << endl;
53      }
54
55      delete current;
56      system("pause");
57  }
```

請注意這邊的使用方式

REQUIREMENTs

R

- Classes & their member functions:
  - myQuadrilateral, myTrapezium, myParallelogram, myRectangle, mySquare
    - Use class **Point2** to storage points
    - Provide: operator<<, getType(), getArea(), isValid()
    - Polymorphism! (記得要支援多形操作！請多參考各頁面的黃字說明)
  - Point2
    - Provide: Point2(int, int),
      - static functions
      - getDistance(const Point2&, const Point2&);
      - isParallel(const &Point2 A, const &Point2 B, const &Point2 C, const &Point2 D)

其他注意事項...

- 架構需要符合作業要求
  - 形狀辨識, getType ...
- 請注意各種特例 – 四邊形變成一個點/一條線或是三角形

複雜多邊形不再本次作業範圍內

Submit your codes to TA

S

- Please complete the implementation of the classes to fulfill all challenge in the driver program (109hw4_main.cpp).
  - 請多多嘗試各種可能!

- 請再次確定你的程式碼可以編譯了
  - 無法編譯此次將無法批改（0分）
  - 助教會嘗試通知你，並請你補交
    - (沒有加入LINE群的請記得加入~
      助教找不到人就請自行負責)

4/26開始會排統一DEMO！

Submit your codes to Portal

S

- Please use s1234567_myQuad.h & .cpp, s1234567_myPoint2.h & .cpp as your file names.
  - Replace s1234567 by your own student ID.
  - And upload **ONLY** these codes.
    - Attach a s1234567_hw4.txt if u want to add some additional information to TA
    - Please ZIP them with your student ID, s1234567_109hw4.zip
    - If you try to upload another files (for example *.sln or others), you get **0** point.
  - Make sure you have complete the assignment and Do it ON YOUR OWN and ON TIME.