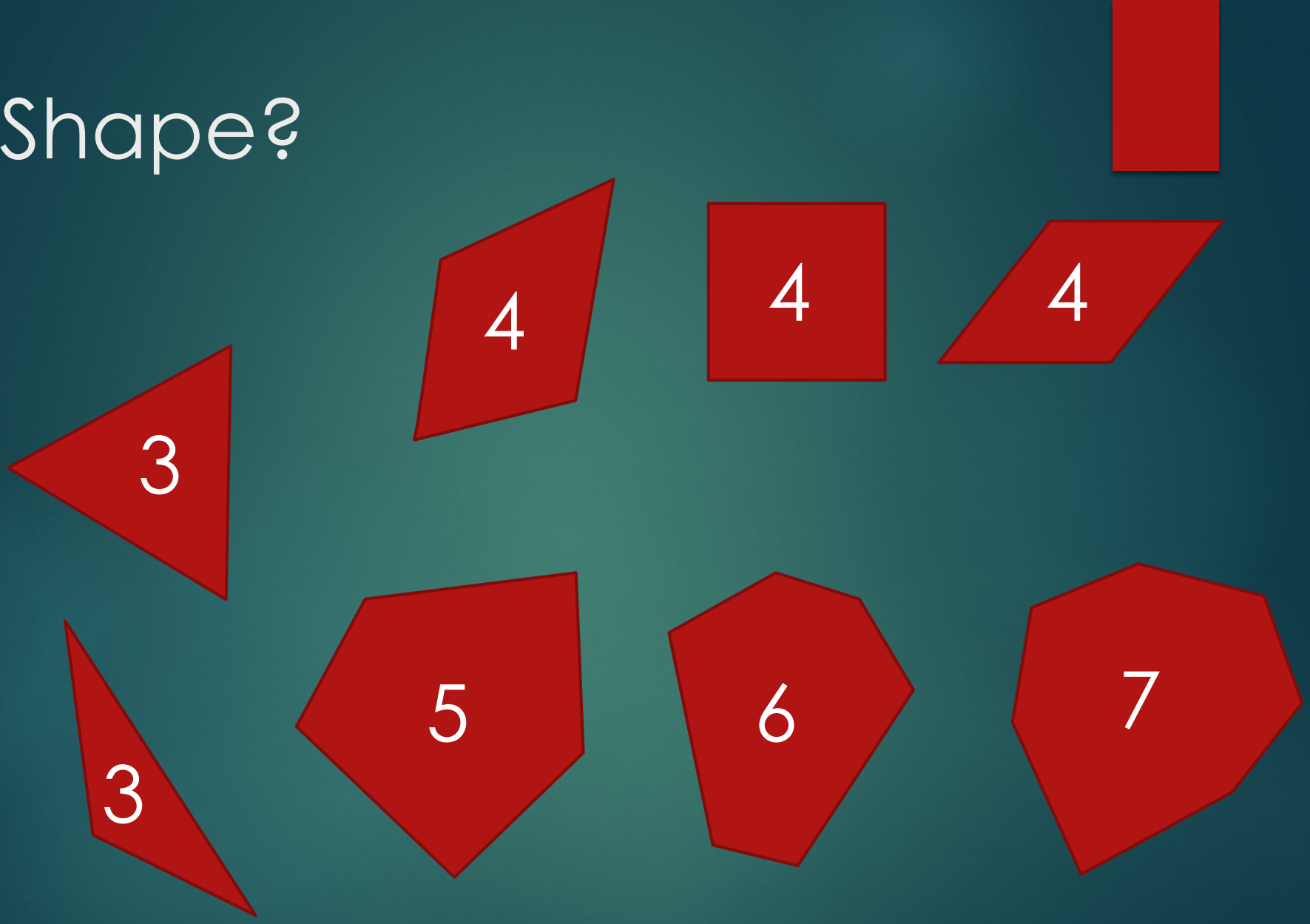
The Shapes

HOMEWORK 5

DUE DATE: 6/6

Shape?

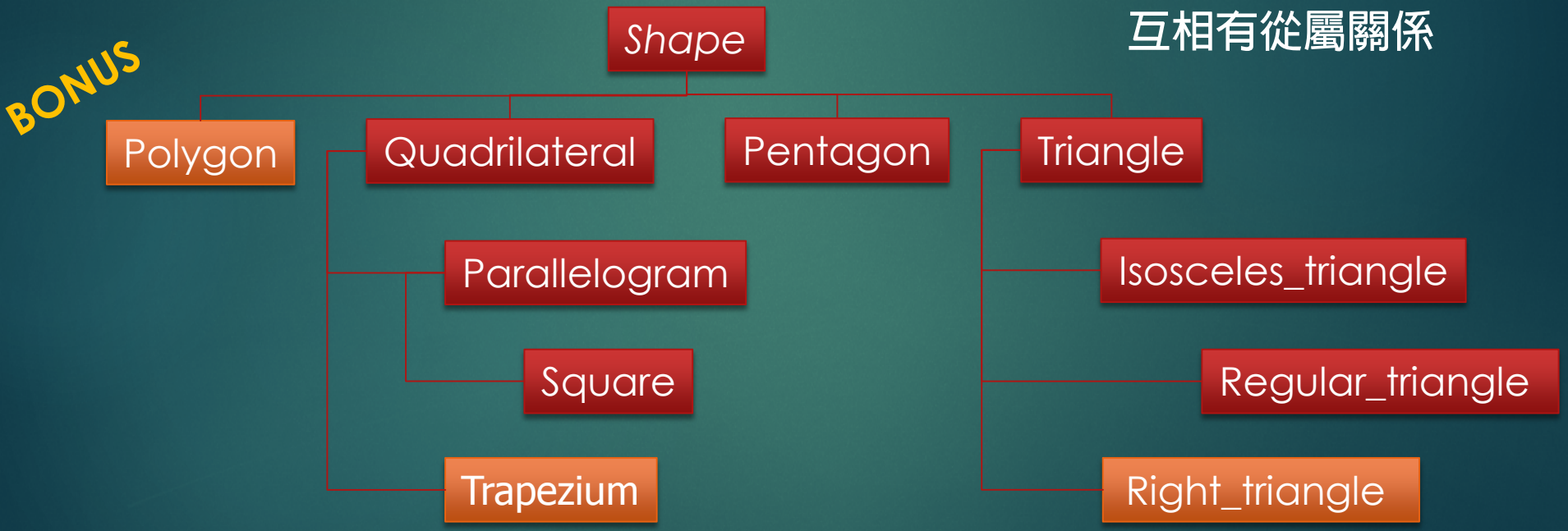


Write a base class for shape…

- Extend this shape class into several specific shapes

實作多種不同的形狀
互相有從屬關係

BONUS



Write a base class for these shapes…

提供以下這五種不同的函示

- Provide these public functions
  - Print information of this shape: print()　印出資訊
    - Print vertices one-by-one
  - Return the TYPE NAME of this shape: getType()　印出類型
    - Triangle, Pentagon, Parallelogram …
  - Calculate area of this shape: getArea()　計算面積
  - Validate its type by: isValid()　檢查形狀是否正確
    - Check the type of shape
    - 比方說等腰三角形是否等腰
    - 直角三角形是否有直角
  - Convex of Concave: isConvex(): BONUS
    - Talk to u later　檢查形狀是凸型還是凹型

Examples

OUTPUT AND SOLUTION

print()

- Pentagon.print()

TYPE: Pentagon

(2.01, 4.56)

(7.45, 5.65)

(6.55, 2.44)

(3.05, 0.0)

(0.0, 3.15)

getType()

- Pentagon.getType()

Pentagon

- Quadrilateral.getType()

Quadrilateral

- Square.getType()

Quadrilateral-Parallelogram-Square

- Right_triangle.getType()

Triangle-Right_triangle

getArea()

- Triangle:

vector absolute

$$\text{AREA} = (1/2) \; | \; \overrightarrow{AB} \times \overrightarrow{AC} \; |$$

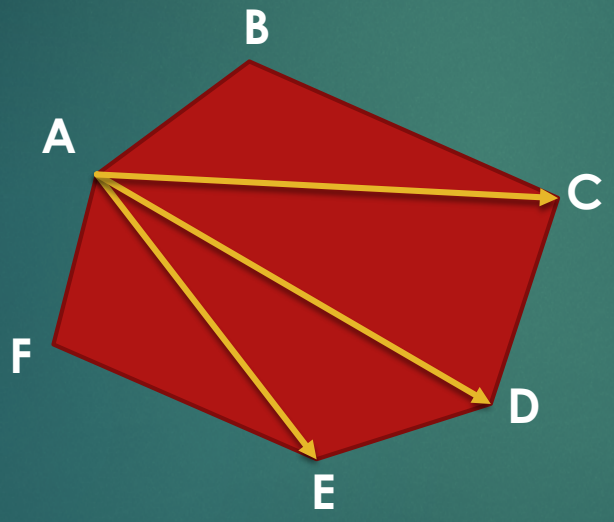
(2D)

Cross product

- http://mathworld.wolfram.com/TriangleArea.html

getArea()

- General solution – dived the shape into multiple triangles



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

isValid()

檢查形狀是否正確

- 比方說是否為矩形?
- 是否回平行四邊形?

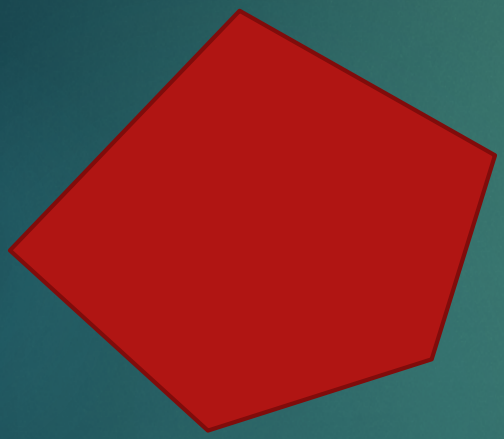
- Is it a square?

- Is it a parallelogram?

- Is it a Isosceles triangle?

isConvex() - BONUS

- http://mathworld.wolfram.com/ConvexPolygon.html



Convex

Concave

Complex

NOTE: You may assume all polygons in the basic test are CONVEX.

Input file

[ShapeType] [x1] [y1] [x2] [y2] [x3] [y3] …

3 2.01 4.56 7.45 5.65 6.55 2.44 3.05 0.0 0.0 3.15

1 2.01 4.56 7.45 5.65 6.55 2.44

7 -2.0 0.0 0.0 6.0 2.0 0.0 (EOF, end of file)

```
static enum ShapeType{
    Shape = 0, Triangle, Quadrilateral, Pentagon, Parallelogram, Square,
    Trapezium, Isosceles_triangle, Regular_triangle, Right_triangle, Polygon
};
```

Scores

- 4 basic shape classes 12pts (可讀取/生成物件)
  - *Shape, Triangle, Quadrilateral, Pentagon*
- 4 specified shape classes
  - *Parallelogram, Square, Isosceles_triangle, Regular_triangle*
- *DEMO 15pts*
- ~~*Shape.print() 7pts*~~
- *getType() for each class: 1pts x 8 = 8pts*
- *getArea() for each basic class: 5pts x 3 = 15pts*
- *getArea() for each specified class: 5pts x 4 = 20pts*
- *isValid() for each basic class: 5pts x 3 = 15pts* (只要算點數即可)
- *isValid() for each basic class: 5pts x 3 = 15pts* (驗證各種形狀)

需針對不同形狀的特性實作
比方說 底*高 或 垂直邊相乘..

BONUS Score

- Complete Trapezium or Right_triangle
  - +5pts for each class
- Complete Polygon +10pts
- NOTE: "Complete" means implement all 4 basic functions correctly
  - print(), getType(), getArea(), isValid()

- Complete isConvex() +5pts for each class
  - *Only for Triangle, Quadrilateral, Pentagon*
  - *Or implement Shape.isConvex() directly (+15)*

(二選一)

```
int main()
{
int NUMofSHAPE = loadShapes("test_shape.txt");
std::cout << "# of shapes = " << NUMofSHAPE << std::endl;
for each (MIME::Shape* var in data) {
    var->print();
    std::cout << "AREA: " << var->getArea() << std::endl;
    std::cout << "Valid?: " << (var->isValid() ? "YES":"NO") <<
    std::endl;
    std::cout << "Convex?: " << (var->isConvex() ? "YES":"NO") <<
    std::endl << std::endl;
}
for each (MIME::Shape* var in data)
    delete var;
system("PAUSE");
return 0;
}
```

Sample:

需針對不同形狀，輸出不同名稱

TYPE: Isosceles_triangle

(-2, 0)

(0, 6)　依序輸出所有點

(2, 0)

AREA: 12　計算面積

Valid?: YES　判定形狀是否正確

Convex?: YES　判定形狀是否為凸型

DEMO