

Composició de classes

Tema 1: Programació orientada a objectes

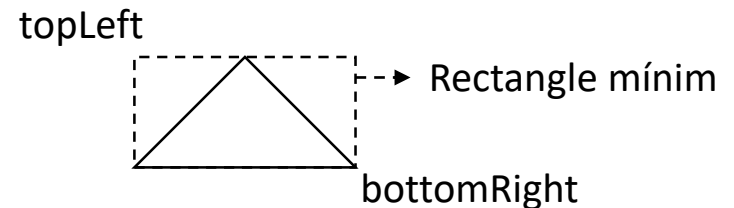
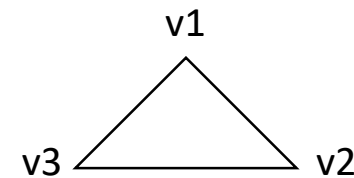
Exercici

A partir de la definició d'aquesta classe Punt volem crear una nova classe Poligon que permeti guardar tots els vèrtexs d'un polígon (màxim 30 vèrtexs) i també les coordenades de la cantonada superior esquerra i de la cantonada inferior dreta del rectangle mínim que engloba al polígon

```
class Punt
{
public:
    Punt();
    Punt(float x, float y);
    ~Punt();

    void setX(float x);
    void setY(float y);
    void llegeix();

    float getX() const;
    float getY() const;
    void mostra() const;
    float distancia(const Punt &p) const;
private:
    float m_x, m_y
};
```



Composició de classes

- Utilització d'una classe ja existent (Punt) com a atribut dins d'una altra classe (Poligon): *un polígon està compost per punts*

Quins atributs hem de posar a la part privada de la classe Poligon?

```
class Poligon
{
private:
    Punt m_vertexs[MAX_VERTEXES];
    int m_nVertexes;
    Punt m_topLeft;
    Punt m_bottomRight;
};
```

Poligon

m_vertexs
m_nVertexes
m_topLeft
m_bottomRight

Punt[]

int

Punt

Punt

Punt

llegeix()
mostra()
distancia(Punt &p)
...

m_x
m_y

Mètodes

Atributs



Relació de composició: Un polígon està compost per N punts

Composició de classes: accés als objectes de la classe

```
class Poligon
{
public:
    void afegeixVertex(const Punt& pt);
private:
    Punt m_vertexs[MAX_VERTEXS];
    int m_nVertexs;
    Punt m_topLeft;
    Punt m_bottomRight;
};
```

```
class Punt
{
public:
    ...
    void setX(const float x);
    void setY(const float y);
    float getX() const;
    float getY() const;
    ...
private:
    float m_x, m_y
};
```

- Dins del codi de la classe Poligon podem **accedir als membres públics** de la classe Punt, però **no podem accedir als membres privats**.

```
void Poligon::afegeixVertex(const Punt& pt)
{
    m_topLeft.m_x = pt.m_x; 
    m_topLeft.setX(pt.getX()); 
}
```

Composició de classes: accés als objectes de la classe

```
void Poligon::afegeixVertex(const Punt& pt)
{
    if (m_nVertexs < MAX_VERTEXES)
    {
        m_vertexs[m_nVertexs] = pt;
        m_nVertexs++;
        if (pt.getX() < m_topLeft.getX())
            m_topLeft.setX(pt.getX());
        ...
    }
}
```

Utilització dels mètodes
públics de la classe Punt

```
class Poligon
{
private:
    Punt m_vertexs[MAX_VERTEXES];
    int m_nVertexs;
    Punt m_topLeft;
    Punt m_bottomRight;
};
```

```
class Punt
{
public:
    ...
    void setX(const float x);
    void setY(const float y);
    float getX() const;
    float getY() const;
    ...
};
```

Composició de classes: constructors

```
int main()
{
    Poligon p;
    ...
}
```

- Inicialització de l'objecte p: crida al constructor de la classe Poligon
- Com queden inicialitzats els objectes de la classe Punt dins de Poligon?

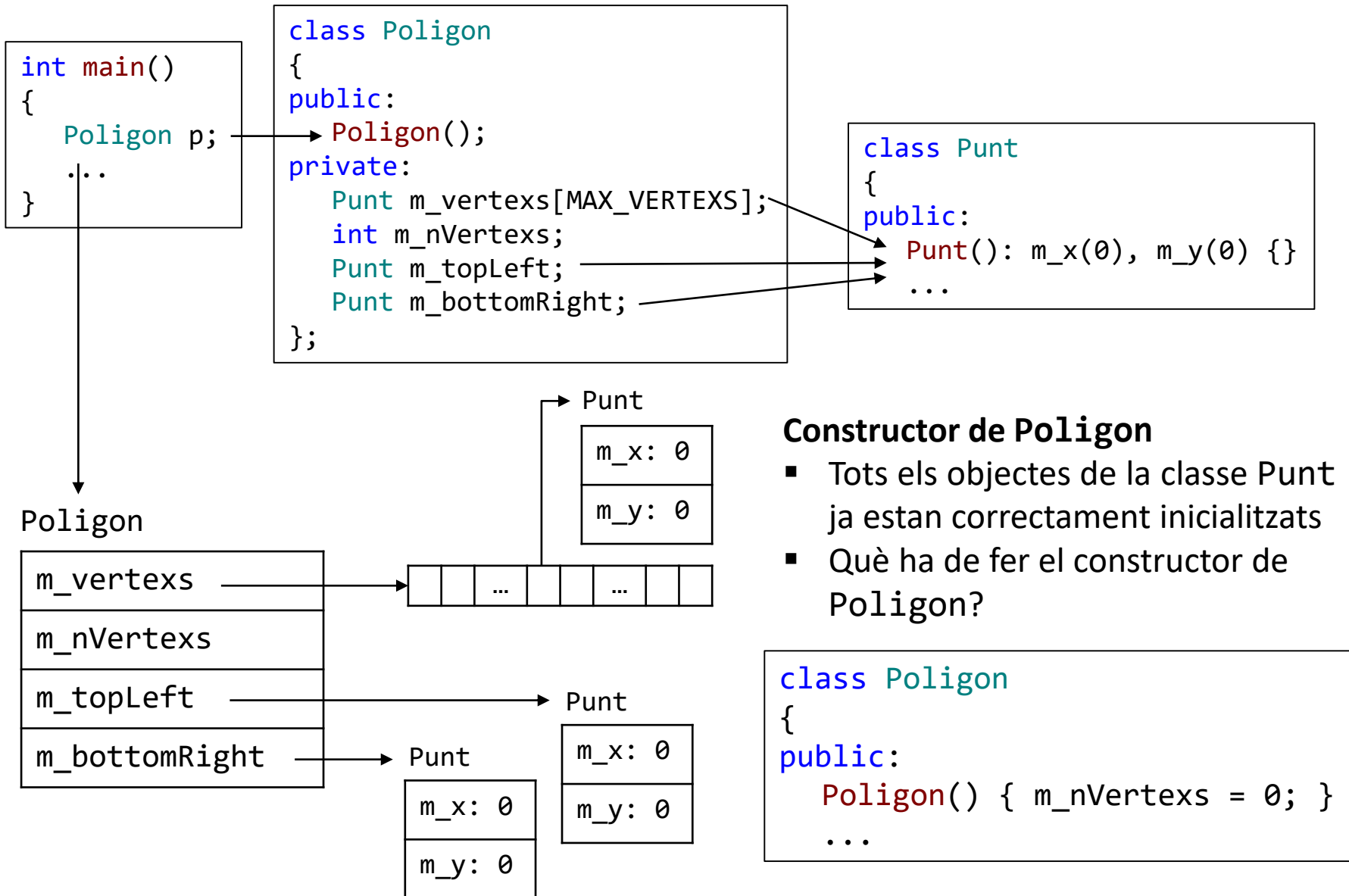
```
class Poligon
{
public:
    Poligon();
private:
    Punt m_vertices[MAX_VERTEXS];
    int m_nVertexs;
    Punt m_topLeft;
    Punt m_bottomRight;
};
```

Composició de classes i constructors:

Quan es crea un objecte d'una classe composta:

1. Es crida **primer** al **constructor** dels objectes que formen part de la classe (en aquest cas tots els objectes de la **classe Punt**) per inicialitzar-los correctament.
2. **Després** es crida al **constructor** de la classe composta (en aquest cas la **classe Poligon**) per acabar de completar la inicialització de la classe.

Composició de classes: constructors



Composició de classes: constructors

I si volem inicialitzar algun punt amb valors diferents als valors per defecte? ...

- Podem indicar que volem inicialitzar el punt cridant al constructor amb paràmetres enlloc del constructor per defecte

```
class Poligon
{
public:
    Poligon(): m_topLeft(1000, 1000), m_bottomRight(-1,-1) { m_nVertexs = 0; }
    ...
}
```

Poligon

m_vertexs
m_nVertexs
m_topLeft
m_bottomRight

Crida constructor defecte

class Punt

```
{
public:
    Punt(): m_x(0), m_y(0) {}
    Punt(float x, float y) : m_x(x), m_y(y) {}
    ...
}
```

Crida constructor paràmetres

