Estructuras (struct)

Tipos basados en struct

- Paso previo a la **Programación Orientada a Objetos**
- Nos permiten usar abstracción a un nivel superior para crear entidades de mayor complejidad que short, long, int, float, double, char, bool
- Podemos crear nuestros propios tipos para representar entidades como por ejemplo estudiantes, clases, grados, Universidades, etc.

Sintaxis

```
// placed in a header file
typedef struct
{
    member_type1 member_name1;
    member_type2 member_name2;
            .
            .
            .
    member_typeN member_nameN;
} type_name;
```

Ejemplo de declaración

```
// placed in a header file
typedef struct
{
    member_type1 member_name1;
    member_type2 member_name2;
            .
            .
            .
    member_typeN member_nameN;
} type_name;
```

```
// point.h
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;
```

Ejemplo de uso

```
int main()
{
    point p1, p2;  // 2 points with 2 floats in each
    p1.m_Xcoord = 4;
    p1.m_Ycoord = 6;
    cout << "enter p2's x: ";
    cin >> p2.m_Xcoord;
    cout << " enter p2's y: ";
    cin >> p2.m_Ycoord;
    cout << "the x coordinate of p1 is " << p1.m_Xcoord;
    ...
```

Ejemplo de uso

```
int main()
{
    point p1, p2;  // 2 points with 2 floats in each
    p1.m_Xcoord = 4;
    p1.m_Ycoord = 6;
    cout << "enter p2's x: ";
    cin >> p2.m_Xcoord;
    cout << " enter p2's y: ";
    cin >> p2.m_Ycoord;
    cout << "the x coordinate of p1 is " << p1.m_Xcoord;
    ...
```

Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

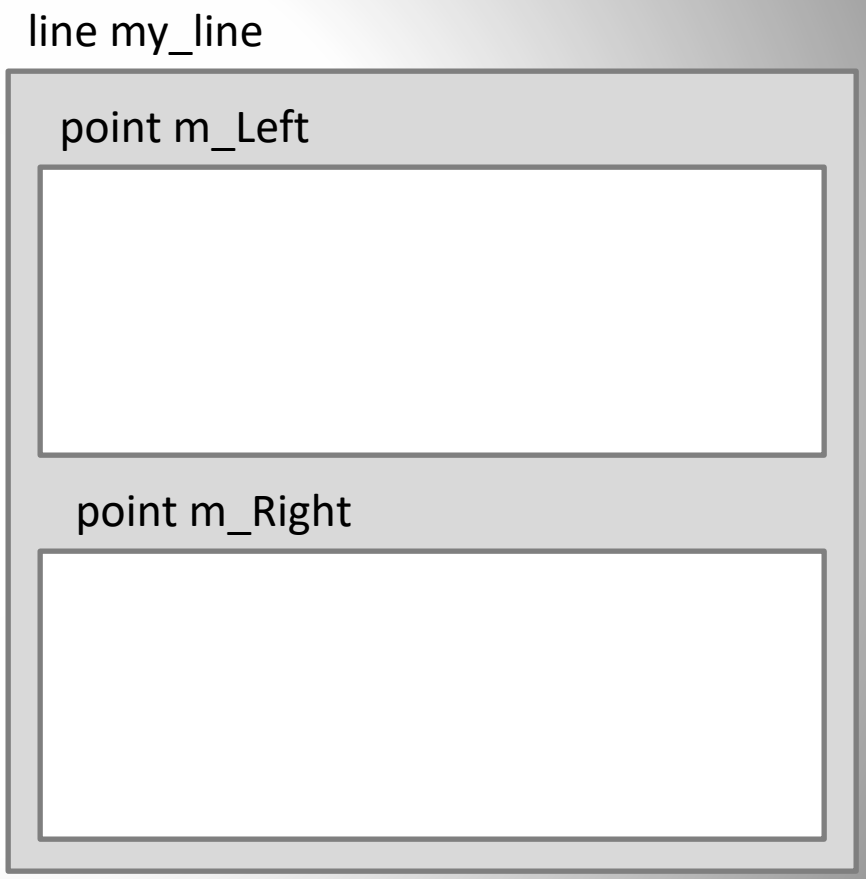
line my_line

Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

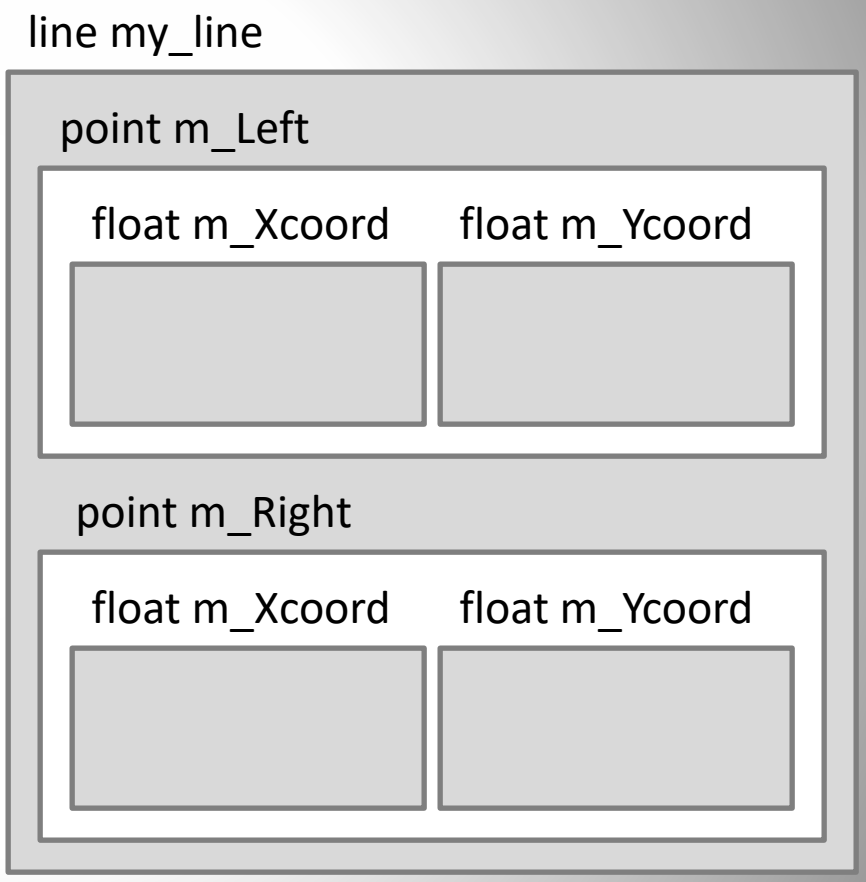


Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

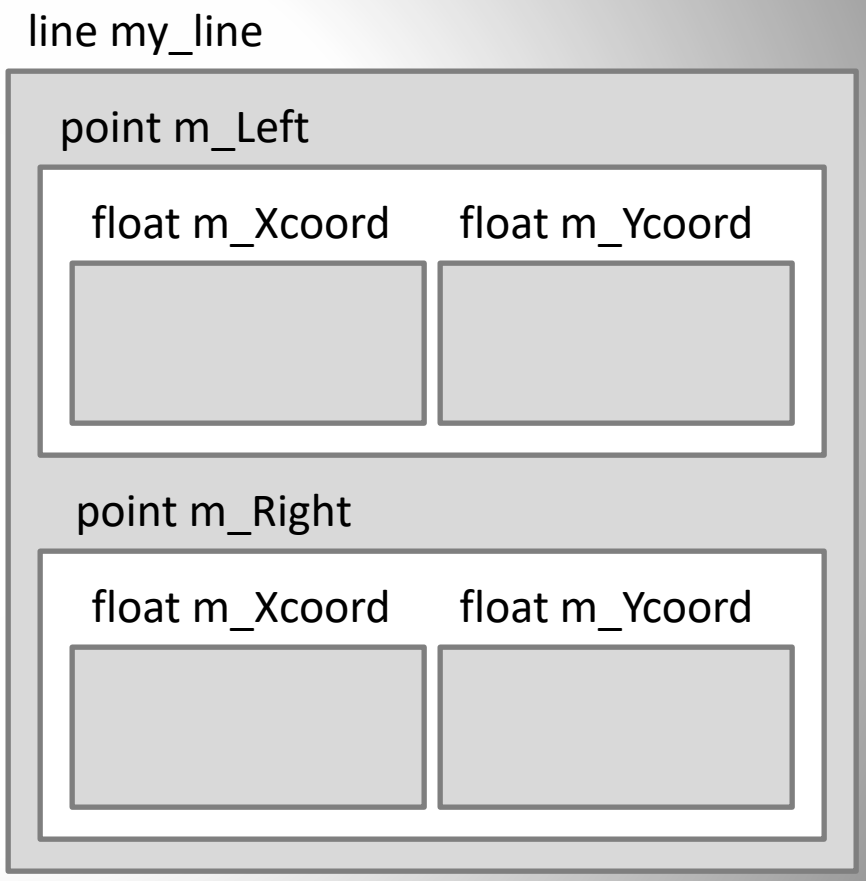


Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

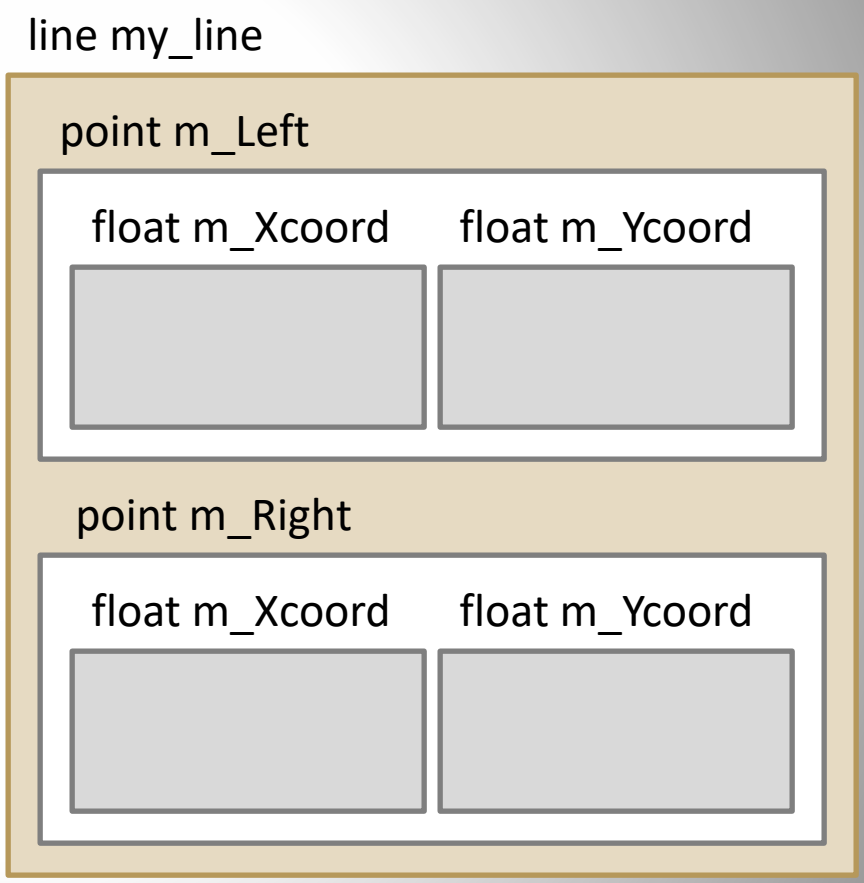


Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

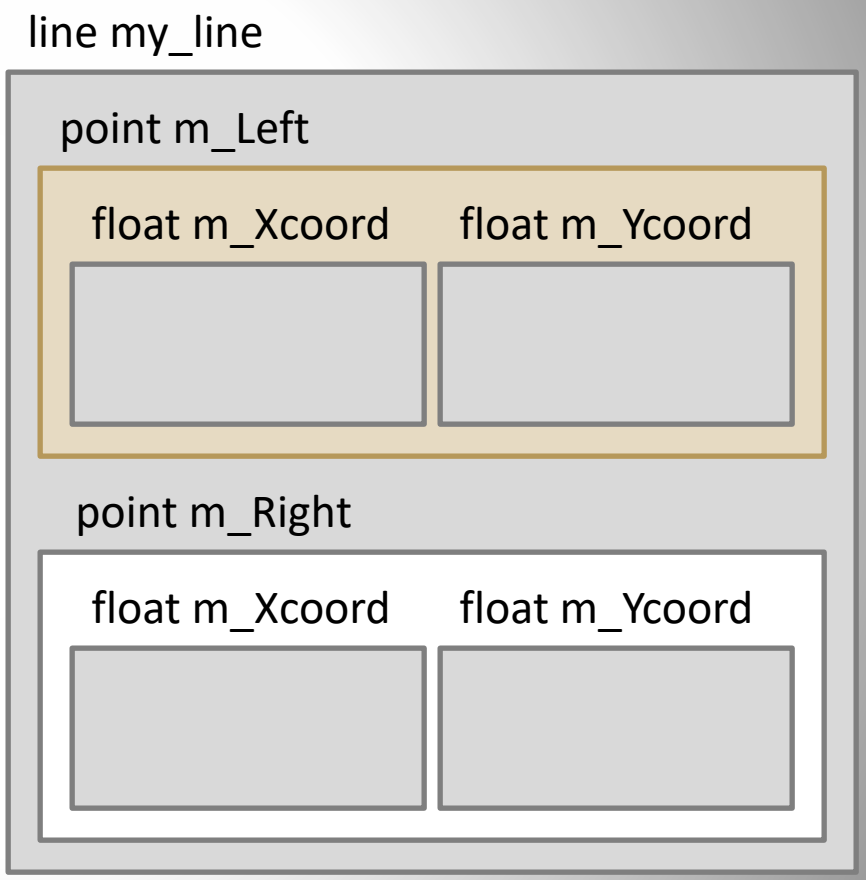


Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

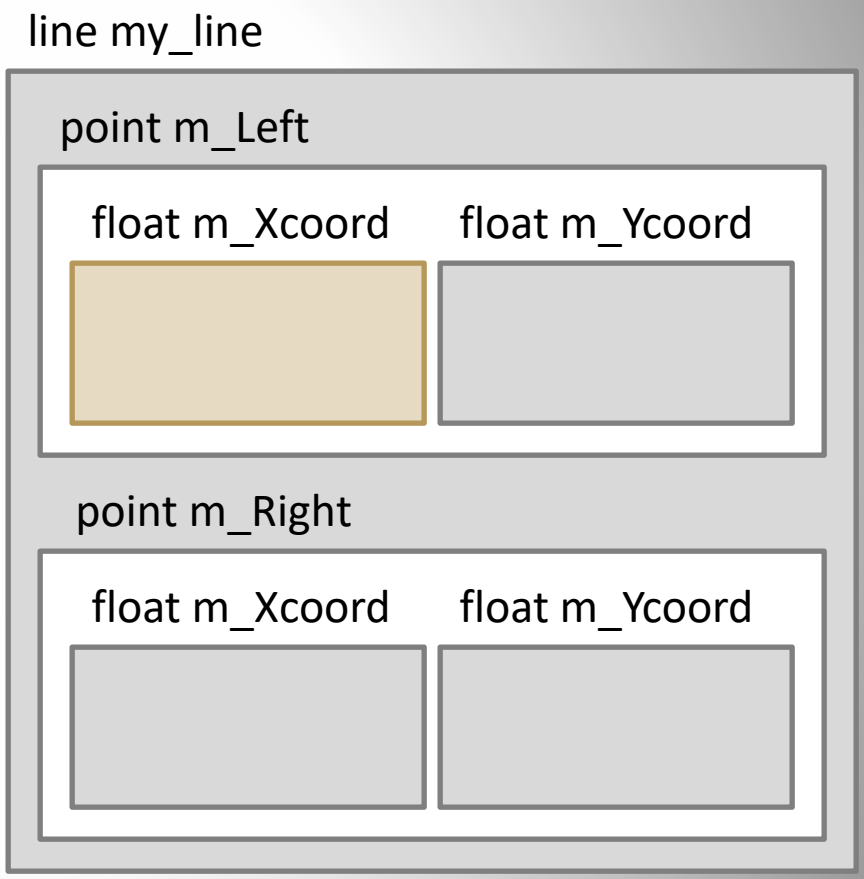


Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```

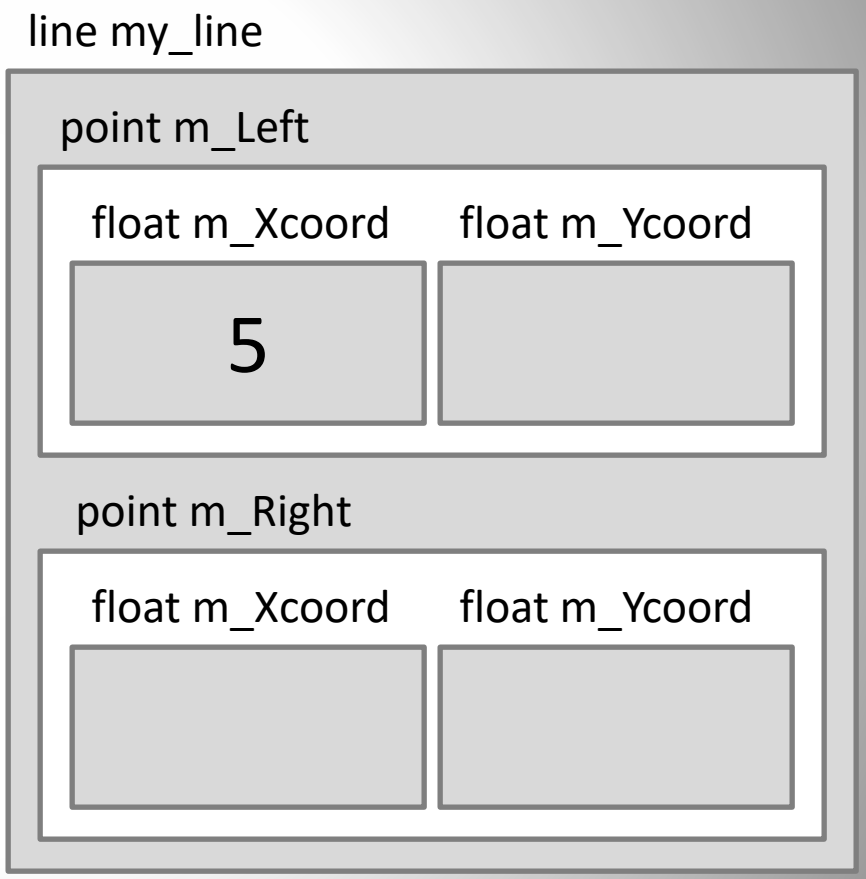


Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```



Estructuras anidadas

```
typedef struct
{
    float m_Xcoord;
    float m_Ycoord;
} point;

typedef struct
{
    point m_Left;
    point m_Right;
} line;

int main()
{
    line my_line;
    my_line.m_Left.m_Xcoord = 5;
    my_line.m_Left.m_Ycoord = 8;
    ...
```