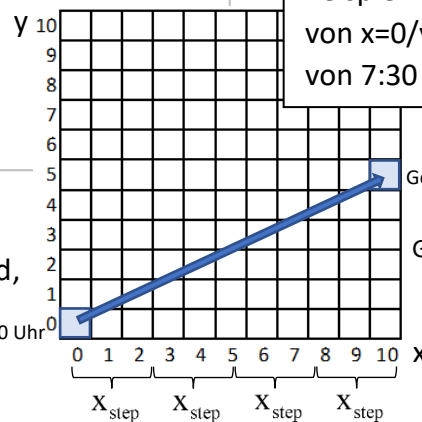


Flugplan

Die Textdatei *Flugplan.csv* enthält geplante Linienflüge der Luftlinie OS.

```
Startzeit,Flugdauer (Min),Name,xstart,ystart,xlandung,ylandung
7:30,60,OS1,0,0,10,5
9:00,90,OS2,0,0,0,10
9:30,120,OS3,10,10,0,0
10:00,90,OS4,10,0,0,5
10:30,60,OS5,0,5,10,5
11:00,75,OS6,2,0,5,10
11:30,90,OS7,5,10,0,0
```



Schreibe eine Simulation in der die Position jedes Flugzeugs vom Start bis zur Landung ausgegeben wird, wenn sich dieses gleichförmig bewegt:

- Simulation von 7:30 Uhr bis 13:00 Uhr
- Simulationsintervall von 15 Minuten

Jedes Flugzeug soll in einem eigenen Thread laufen bzw. fliegen!

Main:

```
static void Main(string[] args)
{
    Flugsicherung pm = new Flugsicherung("Flugplan.csv");
    pm.Simulation();

    Console.WriteLine("Simulation beendet.");
    Console.ReadKey();
}
```

Flugstrecke in x- und y-Richtung eines Flugzeugs während eines Simulationsintervalls von 15 Minuten:

$$x_{\text{step}} = \frac{15}{\text{Flugdauer}} \cdot (x_{\text{Landung}} - x_{\text{Start}}) = 2,5 \text{ für OS1}$$

$$y_{\text{step}} = \frac{15}{\text{Flugdauer}} \cdot (y_{\text{Landung}} - y_{\text{Start}}) = 1,25 \text{ für OS1}$$

1. Definiere eine Klasse für ein Flugzeug die alle Daten speichern kann.

- Die x- und y-Position des Flugzeugs soll jeweils als `double` gespeichert werden.
- Erstelle eine Methode `Fly` die in einem Thread gestartet wird und die Zeitschleife enthält.

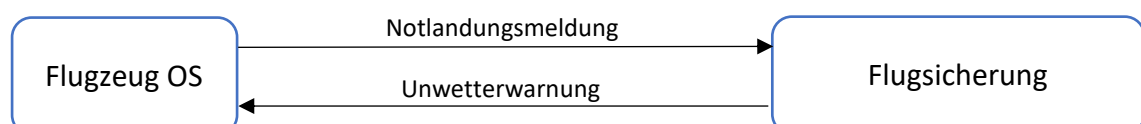
2. Klasse Flugsicherung:

- Lade alle Flüge im Konstruktor in einen Container.
- Erstelle eine Methode `Simulation` mit einer ähnlichen Zeitschleife von 7:30 Uhr bis 13:00 Uhr und demselben Wert für `Sleep` wie in `Flugzeug.Fly`
- Starte in der Zeitschleife jedes Flugzeug bei seinem Startzeitpunkt in einem Thread (egal mit welcher Klasse)

```
public void Fly()
{
    for (DateTime time = /* Startzeitpunkt */
        time < /* Zeitpunkt der Landung */
        time = time.AddMinutes( /* Simulationsintervall */
    {
        Console.WriteLine( /* Name und Position des Flugzeugs */
            System.Threading.Thread.Sleep(500);
        // Flugzeug um xstep und ystep weiterzubewegen
    }
    ...
}
```

3. Erweiterung um eine Unwetterwarnung:

- Alle 15 Minuten besteht eine 25%ige Wahrscheinlichkeit für ein Unwetter, das auf einer zufälligen Stelle des 10x10- Quadrats auftritt.
- Die Unwetterwarnung wird von der Flugsicherung ausgeschildet.
- Falls ein Flugzeug näher als 4 Längeneinheiten beim Unwetter ist, muss es notlanden und meldet seine momentane Position als ein Event an die Flugsicherung.



Beispielausgabe:

07:30 Uhr

OS1 auf 0/0

07:45 Uhr

OS1 auf 2,5/1,2

UNWETTER auf 10/7

08:00 Uhr

OS1 auf 5/2,5

08:15 Uhr

OS1 auf 7,5/3,8

08:30 Uhr

OS1 gelandet auf 10/5

08:45 Uhr

UNWETTER auf 2/10

09:00 Uhr

OS2 auf 0/0

OS2 auf 0/1,7

09:15 Uhr

OS2 auf 0/3,3

UNWETTER auf 4/0

OS3 auf 10/10

09:30 Uhr

OS2 auf 0/5

OS3 auf 8,8/8,8

UNWETTER auf 8/10

OS3 notgelandet auf 9/9

09:45 Uhr

OS2 auf 0/6,7

UNWETTER auf 10/6

10:00 Uhr

OS4 auf 10/0

OS2 auf 0/8,3

10:15 Uhr

OS4 auf 8,3/0,8

OS2 gelandet auf 0/10

UNWETTER auf 7/6

10:30 Uhr

OS5 auf 0/5

OS4 auf 6,7/1,7

10:45 Uhr

OS5 auf 2,5/5

OS4 auf 5/2,5

UNWETTER auf 3/6

OS5 notgelandet auf 2/5

OS6 auf 2/0

11:00 Uhr

OS4 auf 3,3/3,3

OS6 auf 2,6/2

UNWETTER auf 7/7

11:15 Uhr

OS4 auf 1,7/4,2

OS6 auf 3,2/4

11:30 Uhr

OS7 auf 5/10

OS4 gelandet auf 0/5

OS6 auf 3,8/6

11:45 Uhr

OS7 auf 4,2/8,3

OS6 auf 4,4/8

UNWETTER auf 9/10

12:00 Uhr

OS7 auf 3,3/6,7

OS6 gelandet auf 5/10

UNWETTER auf 0/5

OS7 notgelandet auf 3/7

12:15 Uhr

12:30 Uhr

UNWETTER auf 9/6

12:45 Uhr

13:00 Uhr

Simulation beendet.

Beispiele für Unwetterwarnungen die keine Notlandungen zur Folge hatten, weil kein Flugzeug in der Nähe war.

Beispiele für Unwetterwarnungen die zu Notlandungen der Flugzeugen führten, weil sie in der Nähe (<4) waren.