

Class Diagram

A position on the screen

Point
<ul style="list-style-type: none">- x : Float- y : Float
<ul style="list-style-type: none">+ initialize()+ getX() : Float+ getY() : Float+ equals(rhs : Point) : Boolean+ NotEquals(rhs : Point) : Boolean+ setX(x : Float)+ setY(y : Float)+ addX(x : Float)+ addY(y : Float)+ assign(rhs : Point) : Point

Set up OpenGL and get input (sort-a like `cin`):

Interface
<ul style="list-style-type: none">- <u>initialized</u> : Boolean- <u>timePeriod</u> : Float- <u>nextTick</u> : Integer- <u>isDownPress</u> : Integer- <u>isUpPress</u> : Integer- <u>isRightPress</u> : Integer- <u>isLeftPress</u> : Integer- <u>isSpace</u> : Bool- <u>p</u> : Void- <u>callback</u>
<ul style="list-style-type: none">+ initialize()+ run(callback, p)+ isTimeToDraw() : Boolean+ setNextDrawTime()+ getNextTick() : Integer+ setFramesPerSecond(Float)+ getFrameRate() : Float+ isDown() : Integer+ isUp() : Integer+ isLeft() : Integer+ isRight() : Integer+ isSpace() : Boolean

Send output (sort-a like `cout`):

ogstream
pt : Point
+ initialize() + flush() + setPosition() + assign() + drawLander() + drawLanderFlames() + drawStar() + drawRectangle() + drawLine() - rotate() - drawText()

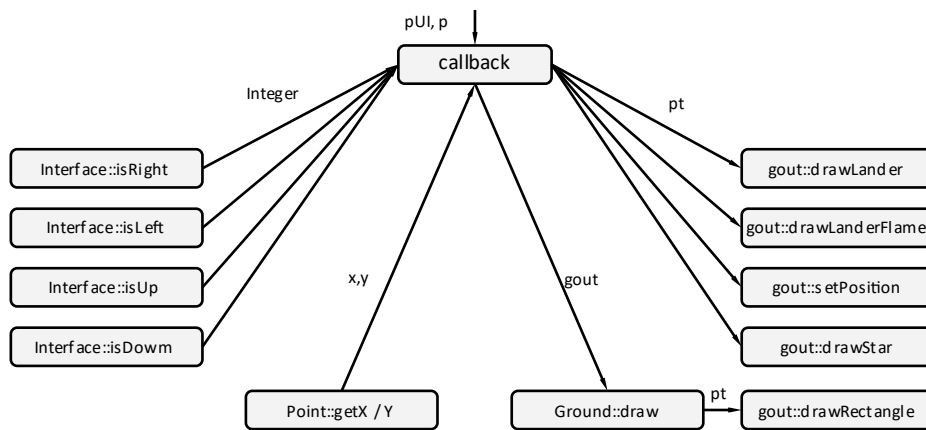
The class that contains the state of the simulator:

Demo
+ ptLM : Point + ptUpperRight : Point + angle : Float + phase : Character + ground : Ground + ptStar : Point
+ initialize()

The ground.

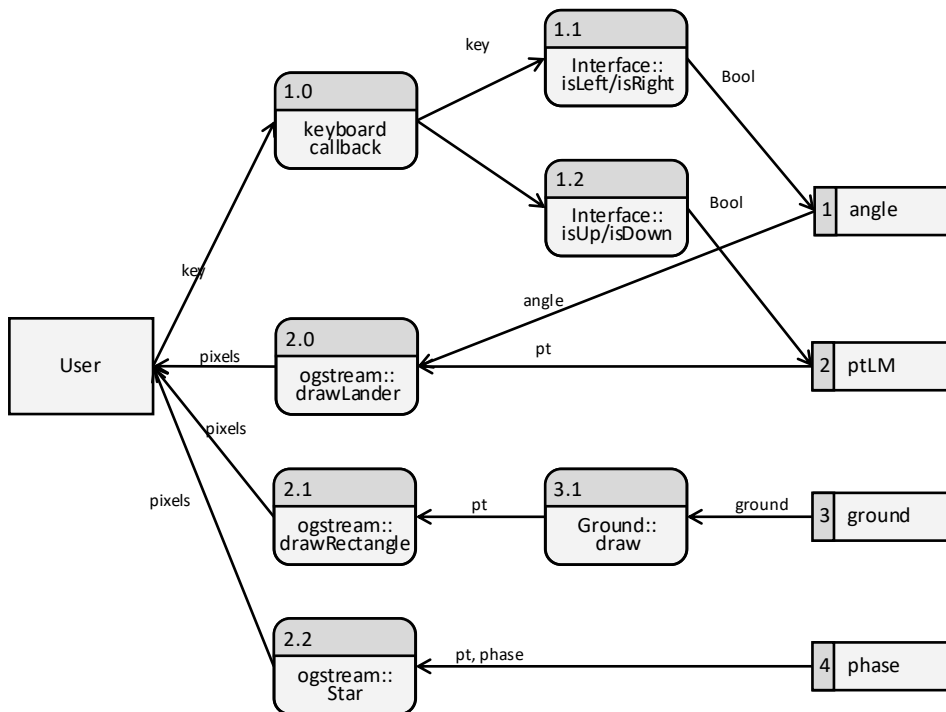
Ground
- ground : Float [*] - iLZ : Integer - ptUpperRight : Point
+ initialize() + reset() + draw(gout : ogstream) + getElevation(pt : Point) : Float + onPlatform(Point, Int) : Boolean + hitGround(Point, Int) : Boolean

Structure Chart



DFD

Input flows on the upper row, output flows on the next three rows



Pseudocode

The main function to handle all the input and update all the simulator state.

```
Callback(pUI, p)
    pDemo ← p

    if pUI.isRight()
        pDemo.angle -= .1
    if pUI.isLeft()
        pDemo.angle += .1
    if pUI.isUp()
        pDemo.y -= 1
    if pUI.isDown()
        pDemo.y += 1

    pDemo.ground.draw()

    drawLander(pDemo.ptLM, pDemo.angle)
    drawLanderFlame(pDemo.ptLM, pDemo.angle,
                    pUI.isDown(), pUI.isLeft(), pUI.isRight())
    DISPLAY pDemo.x pDemo.y
    drawStar(pDemo.ptStar, pDemo.phase++)
```

Test Case

These are for the `Ground::getElevation()` method.

<i>Name</i>	<i>Input</i>	<i>Output</i>
<i>On the ground</i>	ground=[11,12,13,14] pt=(0, 11)	0.0
<i>1 meter above</i>	ground=[11,12,13,14] pt=(0, 12)	1.0
<i>10 meters above</i>	ground=[11,12,13,14] pt=(3, 24)	10.0
<i>Off screen to left</i>	ground=[11,12,13,14] pt=(-1, 100)	0.0
<i>Off screen to right</i>	ground=[11,12,13,14] pt=(4, 100)	0.0
<i>Between points</i>	ground=[11,12,13,14] pt=(0.5, 12)	1.0