Model::Main

TaxiGUI
(from JavaReverse)

+LoadMap(map: int[*,*], size: int): void
+SetTaxiStatus(index: int, point: Point, status: int): void
+SetTaxiType(index: int, type: int): void
+RequestTaxi(src: Point, dst: Point): void
+SetLightStatus(p: Point, status: int): void
+SetRoadStatus(p1: Point, p2: Point, status: int): void
«   *TaxiGUI()

queue1
(from JavaReverse)

~taxinum: int
~sum: int = 0
~sx: int
~sy: int
~tx: int
~ty: int
~way: int
~wayans: int

+repOK(): boolean

redgreeninit
(from JavaReverse)

~strs: String
~sum_of_red: int = 0
~trafficmap: boolean
~waittime: int
~lasttime: long

+repOK(): boolean
+readstrs(filePath: String): boolean
+pre_solve(): void
~solve(i: int, j: int): void
+run(): void

reqinput
(from JavaReverse)

~sos: Scanner
~sum: int = 0
~taxisum: int
~nummax: int

+repOK(): boolean
«   *reqinput(ts: int, ns: int, pw1: pubwork, txx: taxis, gg: TaxiGUI)
~input(): String
~match(input1: String): int
+run(): void

for_tester
(from JavaReverse)

«   *for_tester(taxs1: taxis, rqs1: reqinput)
+repOK(): boolean
~printtaxi(pos: int): void
~printreq(pos: int): void
+run(): void

pubwork
(from JavaReverse)

-full: boolean = true
-mapsig: boolean = true
~nummax: int
~nummax: int
~x: int
~y: int
+map: boolean
+wayans1: int
+wayans2: int
+findway: boolean

«   *pubwork(nm: int, ts: taxis)
+repOK(): boolean
+find(xin: int, yin: int): void
~modifymap(x: int, y: int, type: int): void
~addflow(x: int, y: int, pos: int, flow: int): void
~alloc(q1: queue1): boolean

mainpart
(from JavaReverse)

~nummax: int = 80
~taximax: int = 100

+repOK(): boolean
~set_taxi(gui: TaxiGUI): void
+main(args: String[*]): void

init
(from JavaReverse)

+map: boolean
~strs: String
+wayans: int
~nummax: int
+findway: boolean
+way: int
+mmm: int
+flow: int
~x: int
~y: int

+repOK(): boolean
«   *init(nm: int)
+readstrs(filePath: String): boolean
~build_map(): void
+find(xin: int, yin: int): void
~initial(): boolean

taxi
(from JavaReverse)

~lasttime: long
~lastguest: long
~basetime: long
~x: int
~y: int
+type1: boolean
~summ: int = 0
~lastx: int = 80
~lasty: int = 80
~lasttype: int = 4
~lltype: int = 4
~llx: int = 80
~lly: int = 80
~tx: int
~ty: int
~gx: int
~gy: int
~credit: int = 0
~state: int
~free: boolean = true
~newguest: boolean = false
~self: int
~map: boolean
~way1: boolean
~way: int
~wayans: int
~flowans: int
~father: int
~wayqx: int
~wayqy: int
~inqueue: boolean
~helpmodify: boolean = false

+repOK(): boolean
«   *taxi(xy: int, yy: int, gg: TaxiGUI, tt1: txtoutput, ini1: init)
~pre_work(): void
~traffic_light(): void
~bfs(fromx: int, fromy: int): void
~service(): void
~getguest(): void
~twait(): void
~solve(): void
+run(): void

information
(from JavaReverse)

+getax: Integer[*] {collection="ArrayList"}
+getay: Integer[*] {collection="ArrayList"}
+surax: Integer[*] {collection="ArrayList"}
+suray: Integer[*] {collection="ArrayList"}

+repOK(): boolean

taxis
(from JavaReverse)

+repOK(): boolean

req
(from JavaReverse)

~lastime: long = 0
~basetime: long
~taxisum: int
~sx: int
~sy: int
~tx: int
~ty: int
~live: boolean

+repOK(): boolean
«   *req(txs: taxis, gg: TaxiGUI)
«   *req(time: long, bstime: long)
~addtaxiqueue(i: int): void
~findtaxiqueue(i: int): boolean
+run(): void

taxi_new
(from JavaReverse)

+type1: boolean
+map_ini: boolean
~sum_of_information: int = 0
~nowpo: int

+repOK(): boolean
«   *taxi_new(xy: int, yy: int, gg: TaxiGUI, tt1: txtoutput, ini1: init, type11: boolean)
~next(): information
~previous(): information
~hasnext(): boolean
~hasprevious(): boolean
~update(): void
~bfs(fromx: int, fromy: int): void
~new_service(): void
~new_getguest(): void
~new_solve(): void
+run(): void

txtoutput
(from JavaReverse)

~oo1: FileOutputStream
~oo2: FileOutputStream
~cnt: boolean = true

+repOK(): boolean
«   *txtoutput()
~outputt(str: String, bb: boolean): void

~gui
~gui
{static=true}
~gui
~gui~gui
~gui
~q
~rg
~rgs
{static=true}
~ri
~ini
{static=true}
~ri
{static=true}
~init1
~ini
~ini
~ini
~pw
~pw
~pw
{static=true}
~pw
~rrs
+rr
~info
~taxs
{static=true}
~txs
~txs
~tt
~taxx
~taxs
~txs
{static=true}
~tt
{static=true}
~tt

JavaReverse::Package Structure

JavaReverse::Type Hierarchy

for_tester (from JavaReverse)	
redgreeninit (from JavaReverse)	
req (from JavaReverse)	
reqinput (from JavaReverse)	

Thread
(from JavaReverse)

taxi
(from JavaReverse)

taxi_new
(from JavaReverse)

JPanel
(from JavaReverse)

DrawBoard
(from JavaReverse)

myform
(from JavaReverse)

JFrame
(from JavaReverse)

processform
(from JavaReverse)

debugform
(from JavaReverse)

JavaReverse::Overview

gv | nodedge# | **TaxiGUI** | guitaxi | brush | ~ini | **init** | **Thread** | guiInfo | **JPanel** | **JFrame** | **IOException** | **Graphics** | **Graphics2D** | **Point** | **CopyOnWriteArrayList** | **HashMap** | **JProgressBar** | **JLabel** | **JTextArea** | **Scanner** | **FileOutputStream**

~gui
{static=true}

~ini
~ini

{static=true}

pubwork
{static=true}

~ini {static=true}
redgreeninit

{static=true}
guigv | **DrawBoard**

+drawboard

myform | **processform** | **debugform**

~pw | ~pw
~rg

taxi
{static=true}

txtoutput
{static=true}

queue1 | **taxis**
{static=true}
~taxs
~taxx

~tt | ~txs
~taxs

req

~rr
~rs

information

{static=true}
reqinput

{collection=ArrayList}
*
taxi_new
~res
for_tester

{static=true}
mainpart