

Vyplňování polygonů

Generováno programem Doxygen 1.7.1

út 27. čec 2010 12.26:53

Obsah

1	Rejstřík tříd	1
1.1	Hierarchie tříd	1
2	Rejstřík tříd	3
2.1	Seznam tříd	3
3	Dokumentace tříd	5
3.1	Dokumentace třídy <code>pogr.Algoritmus</code>	5
3.1.1	Detailní popis	6
3.1.2	Dokumentace konstruktoru a destruktoru	6
3.1.2.1	<code>Algoritmus</code>	6
3.1.3	Dokumentace k metodám	6
3.1.3.1	<code>pridejBod</code>	6
3.1.3.2	<code>setBarva</code>	6
3.1.3.3	<code>smaz</code>	6
3.1.3.4	<code>vykresliBod</code>	7
3.1.3.5	<code>vykresliPolygon</code>	7
3.1.3.6	<code>vykresliUsecku</code>	7
3.1.3.7	<code>vyplnPolygon</code>	7
3.1.3.8	<code>zprava</code>	7
3.1.4	Dokumentace k datovým členům	7
3.1.4.1	<code>barva</code>	7
3.1.4.2	<code>g</code>	8
3.1.4.3	<code>labZpravy</code>	8
3.1.4.4	<code>obrazek</code>	8
3.1.4.5	<code>platno</code>	8
3.1.4.6	<code>polygon</code>	8
3.2	Dokumentace třídy <code>pogr.Flood</code>	8
3.2.1	Detailní popis	9

3.2.2	Dokumentace konstruktoru a destruktoru	9
3.2.2.1	Flood	9
3.2.3	Dokumentace k metodám	9
3.2.3.1	platnoMouseClicked	9
3.2.3.2	vypln	9
3.2.3.3	vyplnPolygon	9
3.3	Dokumentace třídy pogr.FloodFIFO	10
3.3.1	Detailní popis	10
3.3.2	Dokumentace konstruktoru a destruktoru	10
3.3.2.1	FloodFIFO	10
3.3.3	Dokumentace k metodám	10
3.3.3.1	vypln	10
3.4	Dokumentace třídy pogr.FloodLIFO	11
3.4.1	Detailní popis	11
3.4.2	Dokumentace konstruktoru a destruktoru	11
3.4.2.1	FloodLIFO	11
3.4.3	Dokumentace k metodám	11
3.4.3.1	vypln	11
3.5	Dokumentace třídy pogr.MujCanvas	12
3.5.1	Detailní popis	12
3.5.2	Dokumentace k metodám	12
3.5.2.1	setObrazek	12
3.6	Dokumentace třídy pogr.PogrAboutBox	12
3.7	Dokumentace třídy pogr.PogrApp	12
3.7.1	Detailní popis	13
3.7.2	Dokumentace k metodám	13
3.7.2.1	configureWindow	13
3.7.2.2	getApplication	13
3.7.2.3	main	13
3.7.2.4	startup	13
3.8	Dokumentace třídy pogr.PogrView	13
3.8.1	Detailní popis	14
3.9	Dokumentace třídy pogr.Scanline	14
3.9.1	Detailní popis	14
3.9.2	Dokumentace konstruktoru a destruktoru	14
3.9.2.1	Scanline	14

3.9.3	Dokumentace k metodám	14
3.9.3.1	najdiPrusecky	14
3.9.3.2	vyplnPolygon	15
3.9.3.3	zkratUsecky	15
3.10	Dokumentace třídy pogr.Stencil	15
3.10.1	Detailní popis	15
3.10.2	Dokumentace konstruktoru a destrukturu	16
3.10.2.1	Stencil	16
3.10.3	Dokumentace k metodám	16
3.10.3.1	vyplnPolygon	16

Kapitola 1

Rejstřík tříd

1.1 Hierarchie tříd

Zde naleznete seznam, vyjadřující vztah dědičnosti tříd. Je seřazen přibližně (ale ne úplně) podle abecedy:

pogr.Algoritmus	5
pogr.Flood	8
pogr.FloodFIFO	10
pogr.FloodLIFO	11
pogr.Scanline	14
pogr.Stencil	15
pogr.MujCanvas	12
pogr.PogrAboutBox	12
pogr.PogrApp	12
pogr.PogrView	13

Kapitola 2

Rejstřík tříd

2.1 Seznam tříd

Následující seznam obsahuje především identifikace tříd, ale nacházejí se zde i další netriviální prvky, jako jsou struktury (struct), unie (union) a rozhraní (interface). V seznamu jsou uvedeny jejich stručné popisy:

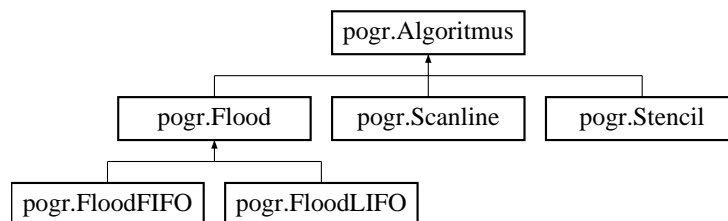
pogr.Algoritmus	5
pogr.Flood	8
pogr.FloodFIFO	10
pogr.FloodLIFO	11
pogr.MujCanvas	12
pogr.PogrAboutBox	12
pogr.PogrApp	12
pogr.PogrView	13
pogr.Scanline	14
pogr.Stencil	15

Kapitola 3

Dokumentace tříd

3.1 Dokumentace třídy `pogr.Algoritmus`

Diagram dědičnosti pro třídu `pogr.Algoritmus`



Veřejné metody

- `Algoritmus` (`MujCanvas _platno`, `JLabel _labZpravy`)
- void `pridejBod` (`Point b`)
- boolean `vykresliPolygon` ()
- void `smaz` ()
- void `obnov` ()
- abstract void `vyplnPolygon` (boolean postup)

Chráněné metody

- final void `zprava` (`String s`)
- void `setBarva` (`int _barva`)
- void `vykresliBod` (`Point b`)
- void `vykresliUsecku` (`int x1`, `int y1`, `int x2`, `int y2`)

Chráněné atributy

- `ArrayList< Point > polygon`
- `MujCanvas platno`
- `JLabel labZpravy`

- Graphics [g](#)
- int [barva](#)
- BufferedImage [obrazek](#)

3.1.1 Detailní popis

Abstraktní třída, která je předkem všech vyplňovacích algoritmů. Obsahuje některé společné funkce pro všechny algoritmy

Autor

Kamil Ondrák

3.1.2 Dokumentace konstruktoru a destruktoru

3.1.2.1 `pogr.Algoritmus.Algoritmus (MujCanvas _platno, JLabel _labZpravy)`

Konstruktory třídy [Algoritmus](#)

Parametry

_platno Plátno (Canvas) do kterého se bude kreslit

_labZpravy Label, do kterého budou zobrazovány zprávy pro uživatele

3.1.3 Dokumentace k metodám

3.1.3.1 `void pogr.Algoritmus.pridejBod (Point b)`

Přidá bod k polygonu a vykreslí ho

Parametry

b Bod

3.1.3.2 `void pogr.Algoritmus.setBarva (int _barva) [protected]`

Nastavuje barvu, kterou bude dále prováděno kreslení

Parametry

_barva barva

3.1.3.3 `void pogr.Algoritmus.smaz ()`

Smaže aktuální polygon a plátno

3.1.3.4 void `pogr.Algoritmus.vykresliBod (Point b)` [protected]

Vykresluje křížek v zadaném bodě a zobrazí aktualizovaný obraz

Parametry

b Bod

3.1.3.5 boolean `pogr.Algoritmus.vykresliPolygon ()`

Vykreslí polygon dle dosud zadaných vrcholů. Pokud jsou zadány méně než tři vrcholy, vypíše zoznámení a polygon nevykresluje

Návratová hodnota

true pokud byl polygon vykreslen, false pokud ne

3.1.3.6 void `pogr.Algoritmus.vykresliUsecku (int x1, int y1, int x2, int y2)` [protected]

Vykreslí úsečku DDA algoritmem. Používá aktuální hodnotu proměnné barva

Parametry

x1 x-ová souřadnice prvního vrcholu

y1 y-ová souřadnice prvního vrcholu

x2 x-ová souřadnice druhého vrcholu

y2 y-ová souřadnice druhého vrcholu

3.1.3.7 abstract void `pogr.Algoritmus.vyplnPolygon (boolean postup)` [pure virtual]

Abstraktní třída pro samotné vyplnění polygonu

Parametry

postup Pokud je true, bude průběžně vykreslován postup vyplňování na obrazovku

Implementováno v [pogr.Flood](#), [pogr.Scanline](#) a [pogr.Stencil](#).

3.1.3.8 final void `pogr.Algoritmus.zprava (String s)` [protected]

Zobrazuje zprávu uživateli

Parametry

s zpráva

3.1.4 Dokumentace k datovým členům**3.1.4.1 int `pogr.Algoritmus.barva` [protected]**

barva, kterou probíhá kreslení

3.1.4.2 Graphics `pogr.Algoritmus.g` [protected]

Graphics pro kreslení vygenerovaného obrázku

3.1.4.3 JLabel `pogr.Algoritmus.labZpravy` [protected]

Label pro zprávy pro uživatele

3.1.4.4 BufferedImage `pogr.Algoritmus.obrazek` [protected]

Obrázek, do kterého se kreslí

3.1.4.5 MujCanvas `pogr.Algoritmus.platno` [protected]

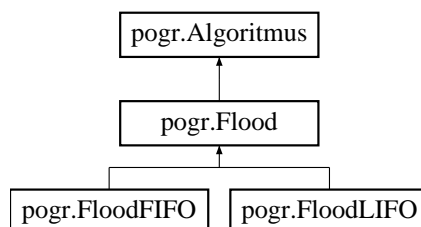
Plátno, do kterého probíhá vykreslování vygenerovaného obrazu

3.1.4.6 ArrayList<Point> `pogr.Algoritmus.polygon` [protected]

Seznam všech vrcholů v zadaném polygonu

3.2 Dokumentace třídy `pogr.Flood`

Diagram dědičnosti pro třídu `pogr.Flood`



Veřejné metody

- `Flood` (`MujCanvas platno`, `JLabel label`)
- `void vyplnPolygon` (`boolean _postup`)

Chráněné metody

- `void platnoMouseClicked` (`MouseEvent evt`)
- `abstract void vypln` (`Point b`)

Chráněné atributy

- `boolean postup`

3.2.1 Detailní popis

Abstraktní třída, která obsahuje základní funkce pro Floodfill

Autor

Kamil Ondrák

3.2.2 Dokumentace konstruktoru a destruktoru

3.2.2.1 `pogr.Flood.Flood (MujCanvas platno, JLabel label)`

Konstruktor třídy

Parametry

platno Plátno pro kreslení

label Label pro vypisování zpráv

3.2.3 Dokumentace k metodám

3.2.3.1 `void pogr.Flood.platnoMouseClicked (MouseEvent evt) [protected]`

Funkce, která obstarává umístění prvotního semínka do plátna. Především volá funkci vyplň. Dále pak dává uživateli najevo, že se něco děje - změni kurzor myši a vypíše zprávu.

Parametry

evt Událost - kliknutí myši

3.2.3.2 `abstract void pogr.Flood.vypln (Point b) [protected, pure virtual]`

Abstraktní třída pro vyplnění oblasti

Parametry

b Bod, kde je umístěno počáteční semínko

Implementováno v [pogr.FloodFIFO](#) a [pogr.FloodLIFO](#).

3.2.3.3 `void pogr.Flood.vyplnPolygon (boolean _postup) [virtual]`

Funkce dovolí uživateli zvolit počáteční semínko, proto přidá k plátnu listener, který čeká na událost kliknutí.

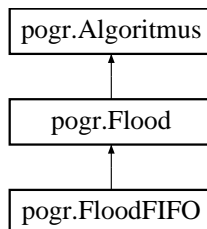
Parametry

_postup

Implementuje [pogr.Algoritmus](#).

3.3 Dokumentace třídy `pogr.FloodFIFO`

Diagram dědičnosti pro třídu `pogr.FloodFIFO`



Veřejné metody

- `FloodFIFO` ([MujCanvas platno](#), `JLabel label`)

Chráněné metody

- `void vypln` (`Point b`)

3.3.1 Detailní popis

Třída, která umožňuje vyplnění oblasti metodou Floodfill za použití fronty

Autor

Kamil Ondrák

3.3.2 Dokumentace konstruktoru a destruktoru

3.3.2.1 `pogr.FloodFIFO.FloodFIFO (MujCanvas platno, JLabel label)`

Konstruktör

Parametry

platno Plátno pro kreslení

label Label pro zprávy

3.3.3 Dokumentace k metodám

3.3.3.1 `void pogr.FloodFIFO.vypln (Point b) [protected, virtual]`

Funkce pro samotné vyplnění oblasti

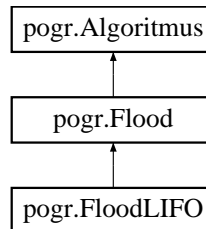
Parametry

b Umístění počátečního semínka

Implementuje [pogr.Flood](#).

3.4 Dokumentace třídy `pogr.FloodLIFO`

Diagram dědičnosti pro třídu `pogr.FloodLIFO`



Veřejné metody

- `FloodLIFO` (`MujCanvas platno`, `JLabel label`)

Chráněné metody

- `void vypln` (`Point b`)

3.4.1 Detailní popis

Třída, která umožňuje vyplnění oblasti metodou Floodfill za použití zásobníku

Autor

Kamil Ondrák

3.4.2 Dokumentace konstruktoru a destruktoru

3.4.2.1 `pogr.FloodLIFO.FloodLIFO (MujCanvas platno, JLabel label)`

Konstruktör

Parametry

platno Plátno pro kreslení

label Label pro zprávy

3.4.3 Dokumentace k metodám

3.4.3.1 `void pogr.FloodLIFO.vypln (Point b) [protected, virtual]`

Funkce pro samotné vyplnění oblasti

Parametry

b Umístění počátečního semínka

Implementuje `pogr.Flood`.

3.5 Dokumentace třídy `pogr.MujCanvas`

Dědí z báze třídy `java.awt.Canvas`.

Veřejné metody

- void `setObrazek` (`BufferedImage _obrazek`)
- void `paint` (`Graphics g`)

Chráněné atributy

- `BufferedImage obrazek`

3.5.1 Detailní popis

Plátno, které při vykreslení zobrazí aktuální `BufferedImage` uložený v proměnné `obrazek`

Autor

Kamil Ondrák

3.5.2 Dokumentace k metodám

3.5.2.1 void `pogr.MujCanvas.setObrazek` (`BufferedImage _obrazek`)

Setter pro `obrazek`

Parametry

`_obrazek` nový obrázek

3.6 Dokumentace třídy `pogr.PogrAboutBox`

Veřejné metody

- `PogrAboutBox` (`java.awt.Frame parent`)
- void `closeAboutBox` ()

3.7 Dokumentace třídy `pogr.PogrApp`

Dědí z báze třídy `org.jdesktop:application::SingleFrameApplication`.

Statické veřejné metody

- static `PogrApp` `getApplication` ()
- static void `main` (`String[] args`)

Chráněné metody

- void `startup` ()
- void `configureWindow` (`java.awt.Window root`)

3.7.1 Detailní popis

Hlavní třída aplikace

3.7.2 Dokumentace k metodám

3.7.2.1 void `pogr.PogrApp.configureWindow` (`java.awt.Window root`) [`protected`]

Metoda pro inicializaci okna. Zobrazené okno už je připraveno díky kodu vygenerovanému jinde, proto je metoda prázdná.

3.7.2.2 static `PogrApp pogr.PogrApp.getApplication` () [`static`]

Spolehlivý statický getter pro instanci aplikace.

Návratová hodnota

instance `PogrApp`

3.7.2.3 static void `pogr.PogrApp.main` (`String[] args`) [`static`]

Hlavní metoda spouštějící aplikaci.

Parametry

args argumenty zadané při spouštění

3.7.2.4 void `pogr.PogrApp.startup` () [`protected`]

Na začátku je vytvořeno a zobrazeno hlavní okno aplikace

3.8 Dokumentace třídy `pogr.PogrView`

Dědí z báze třídy `org.jdesktop::application::FrameView`.

Veřejné metody

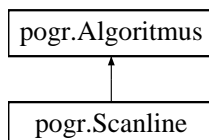
- `PogrView` (`SingleFrameApplication app`)
- void `showAboutBox` ()

3.8.1 Detailní popis

Hlavní okno aplikace

3.9 Dokumentace třídy `pogr.Scanline`

Diagram dědičnosti pro třídu `pogr.Scanline`



Veřejné metody

- `Scanline` (`MujCanvas _platno`, `JLabel label`)
- `void vyplnPolygon` (`boolean postup`)

Chráněné metody

- `void zkratUsecky` ()
- `ArrayList< Point > najdiPruseciky` (`int radek`, `int zac`, `int kon`)

3.9.1 Detailní popis

Třída pro vyplňování geometricky zadaného polygonu pomocí rozkladových řádků

Autor

Kamil

3.9.2 Dokumentace konstruktoru a destruktoru

3.9.2.1 `pogr.Scanline.Scanline (MujCanvas _platno, JLabel label)`

Konstruktork

Parametry

_platno Plátno pro kreslení

label Label pro zobrazování zpráv pro uživatele

3.9.3 Dokumentace k metodám

3.9.3.1 `ArrayList<Point> pogr.Scanline.najdiPruseciky (int radek, int zac, int kon)` [protected]

Funkce hledá vechyn průsečíky na zadaném řádku

Parametry

radek Řádek ,ve kterém se průsečíky hledají

zac x-ová souřadnice začátku řádku

kon x-ová souřadnice konce řádku

Návratová hodnota

seznam nalezených průsečíků

3.9.3.2 void `pogr.Scanline.vyplnPolygon (boolean postup) [virtual]`

Funkce vyplňuje polygon pomocí rozkladových řádků

Parametry

postup Pokud je true, bude průběžně vykreslován postup vyplňování na obrazovku

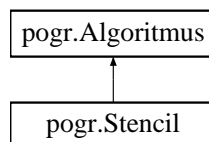
Implementuje [pogr.Algoritmus](#).

3.9.3.3 void `pogr.Scanline.zkratUsecky () [protected]`

Funkce generuje ArrayList usecky, obsahující všechny hrany polygonu. Hrany jsou na spodním okraji zkráceny o jeden pixel, abychom se vyhnuli lichému počtu průsečíků s rozkladovou řádkou

3.10 Dokumentace třídy `pogr.Stencil`

Diagram dědičnosti pro třídu `pogr.Stencil`

**Veřejné metody**

- [Stencil](#) ([MujCanvas](#) _platno, JLabel label)
- void [vyplnPolygon](#) (boolean postup)

3.10.1 Detailní popis

Třída, která zastupuje algoritmus pro inverzní vyplňování geometricky zadaného polygonu pomocí šablony.

Autor

Kamil Ondrák

3.10.2 Dokumentace konstruktoru a destruktoru

3.10.2.1 `pogr.Stencil.Stencil (MujCanvas _platno, JLabel label)`

Konstruktor

Parametry

_platno plátno pro kreslení

label label pro zobrazování zpráv uživateli

3.10.3 Dokumentace k metodám

3.10.3.1 `void pogr.Stencil.vyplnPolygon (boolean postup) [virtual]`

Metoda provádějící vyplnění polygonu.

Nejprve se sestaví 2D pole *sb*, zastupující stencil buffer, a inicializuje všechno jeho prvky na -1. Poté metoda projde všechny hrany polygonu a oblast vpravo od nich ve stencil buffer vynásobí -1. Na závěr jsou pak vybareveny body, které mají v *sb* hodnotu 1.

Parametry

postup Pokud je true, bude průběžně vykreslován postup vyplňování na obrazovku

Implementuje [pogr.Algoritmus](#).

Rejstřík

Algoritmus
pogr::Algoritmus, 6

barva
pogr::Algoritmus, 7

configureWindow
pogr::PogrApp, 13

Flood
pogr::Flood, 9

FloodFIFO
pogr::FloodFIFO, 10

FloodLIFO
pogr::FloodLIFO, 11

g
pogr::Algoritmus, 7

getApplication
pogr::PogrApp, 13

labZpravy
pogr::Algoritmus, 8

main
pogr::PogrApp, 13

najdiPruseciky
pogr::Scanline, 14

obrazek
pogr::Algoritmus, 8

platno
pogr::Algoritmus, 8

platnoMouseClicked
pogr::Flood, 9

pogr::Algoritmus, 5
Algoritmus, 6

barva, 7

g, 7

labZpravy, 8

obrazek, 8

platno, 8

polygon, 8

pridejBod, 6

setBarva, 6

smaz, 6

vykresliBod, 6

vykresliPolygon, 7

vykresliUsecku, 7

vyplnPolygon, 7

zprava, 7

pogr::Flood, 8

Flood, 9

platnoMouseClicked, 9

vypln, 9

vyplnPolygon, 9

pogr::FloodFIFO, 10

FloodFIFO, 10

vypln, 10

pogr::FloodLIFO, 11

FloodLIFO, 11

vypln, 11

pogr::MujCanvas, 12

setObrazek, 12

pogr::PogrAboutBox, 12

pogr::PogrApp, 12

configureWindow, 13

getApplication, 13

main, 13

startup, 13

pogr::PogrView, 13

pogr::Scanline, 14

najdiPruseciky, 14

Scanline, 14

vyplnPolygon, 15

zkratUsecky, 15

pogr::Stencil, 15

Stencil, 16

vyplnPolygon, 16

polygon

pogr::Algoritmus, 8

pridejBod

pogr::Algoritmus, 6

Scanline

pogr::Scanline, 14

setBarva

pogr::Algoritmus, 6

setObrazek
 pogr::MujCanvas, [12](#)
smaz
 pogr::Algoritmus, [6](#)
startup
 pogr::PogrApp, [13](#)
Stencil
 pogr::Stencil, [16](#)

vykresliBod
 pogr::Algoritmus, [6](#)
vykresliPolygon
 pogr::Algoritmus, [7](#)
vykresliUsecku
 pogr::Algoritmus, [7](#)
vypln
 pogr::Flood, [9](#)
 pogr::FloodFIFO, [10](#)
 pogr::FloodLIFO, [11](#)
vyplnPolygon
 pogr::Algoritmus, [7](#)
 pogr::Flood, [9](#)
 pogr::Scanline, [15](#)
 pogr::Stencil, [16](#)

zkratUsecky
 pogr::Scanline, [15](#)
zprava
 pogr::Algoritmus, [7](#)