

Package ‘proftools’

February 20, 2015

Title Profile Output Processing Tools for R

Version 0.1-0

Author Luke Tierney and Riad Jarjour

Description Tools for examining Rprof profile output.

Maintainer Luke Tierney <luke-tierney@uiowa.edu>

Suggests graph,Rgraphviz

License GPL

ByteCompile yes

NeedsCompilation no

Repository CRAN

Date/Publication 2013-10-03 19:34:34

R topics documented:

flatProfile	1
plotProfileCallGraph	2
printProfileCallGraph	4
profileCallGraph2Dot	6
readProfileData	8
styles	9

Index	11
--------------	-----------

flatProfile	<i>Flat Profile for Rprof Profile Data</i>
-------------	--

Description

Computes a flat profile reflecting time spent in functions themselves (self) and functions plus callees (total).

Usage

```
flatProfile(pd, byTotal = TRUE)
```

Arguments

pd	profile data as returned by readProfileData.
byTotal	logical; sort by total time if true, self time if not.

Details

If `byTotal` is true then the result is analogous to the `by.total` component of the result returned by `summaryRprof`. Otherwise, the result is analogous to the `by.self` component returned by `summaryRprof` but with an additional cumulative self times column. The result returned when `byTotal` is not true is analogous to the flat profile produced by `gprof`.

Value

A matrix with one row per function recorded in the profile data.

Author(s)

Luke Tierney

References

User manual for `gprof`, the GNU profiler.

See Also

[Rprof](#), [summaryRprof](#), [readProfileData](#), [plotProfileCallGraph](#), [printProfileCallGraph](#), [profileCallGraph2Dot](#)

Examples

```
pd <- readProfileData(system.file("samples", "glmEx.out", package="proftools"))
flatProfile(pd)
flatProfile(pd, FALSE)
```

`plotProfileCallGraph` *Plot Call Graph for Rprof Profile Data*

Description

Uses the `graph` and `Rgraphviz` packages to plot a call graph for profile data produced by `Rprof`.

Usage

```
plotProfileCallGraph(pd, layout = "dot",
                    score = c("none", "total", "self"),
                    transfer = function(x) x, nodeColorMap = NULL,
                    edgeColorMap = NULL, mergeCycles = FALSE,
                    edgesColored = FALSE, rankDir = c("TB", "LR"),
                    nodeDetails = FALSE, edgeDetails = FALSE,
                    nodeSizeScore = c("none", "total", "self"),
                    edgeSizeScore = c("none", "total"),
                    shape = "ellipse", style, ...)
```

Arguments

pd	profile data as returned by readProfileData.
layout	The layout method to use: One of "dot", "neato", and "twopi".
score	character string specifying whether to use total time or self time for coloring nodes/edges; no color used if missing.
transfer	function; maps score values in unit interval to unit interval
nodeColorMap, edgeColorMap	character vectors of color specifications as produced by rainbow; transfer of score is mapped to color
mergeCycles	logical; whether to merge each cycle of recursion into a single node
edgesColored	logical; whether to color edges
rankDir	The direction that the plot is laid out in, one of either "TB" for Top-to-Bottom or "LR" for Left-to-Right. The default value is "LR". This argument is only useful for dot layouts.
nodeDetails, edgeDetails	logical; whether count information should be shown.
nodeSizeScore	character; value to encode in the size of the nodes.
edgeSizeScore	character; value to encode in the width of the edges.
shape	character; node shape.
style	named list of values for arguments score through layout to use if not explicitly supplied.
...	additional arguments for the graphNEL plot method.

Details

Color is used to encode the fraction of total or self time spent in each function or call. The scores used correspond to the values in the printed representation produced by printProfileCallGraph. For now, see the gprof manual for further details. The color encoding for a score s and a color map m is $m[\text{ceiling}(\text{length}(m) * \text{transfer}(s))]$

A style can be specified to set options to a set of cvalues that work well together.

Value

Used for side effect.

Note

Because of lazy evaluation, nested calls like $f(g(x))$ appear in the profile graph as f or one of its callees calling g .

Author(s)

Luke Tierney

References

User manual for gprof, the GNU profiler.

Graphviz: <http://www.research.att.com/sw/tools/graphviz/>

See Also

[Rprof](#), [summaryRprof](#), [readProfileData](#), [flatProfile](#), [profileCallGraph2Dot](#) [printProfileCallGraph](#)
[plain.style](#) [google.style](#)

Examples

```
pd <- readProfileData(system.file("samples", "glmEx.out", package="proftools"))
plotProfileCallGraph(pd)
plotProfileCallGraph(pd, score = "total")
plotProfileCallGraph(pd, style = google.style, score = "total")
```

`printProfileCallGraph` *Print Call Graph for Rprof Profile Data*

Description

Prints a representation of the call graph for profile data produced by Rprof. Output can be directed to a connection or a file.

Usage

```
printProfileCallGraph(pd, file = stdout(), percent = TRUE)
```

Arguments

pd	profile data as returned by readProfileData.
file	a connection or the name of the file where the profile graph will be written.
percent	logical; if true use percent of total time; otherwise use time in seconds

Details

printProfileCallGraph produces a printed representation of the call graph for profile data produced by Rprof. The representation is analogous to the call graph produced by gprof with a few minor changes. eventually more complete documentation of the format will be provided here; for now, reading the gprof manual section on the call graph should help understanding this output. The output is similar enough to gprof output for the cgprof script to be able to produce a visual representation of the call graph via Graphviz.

Value

Used for side effect.

Note

Because of lazy evaluation, nested calls like $f(g(x))$ appear in the profile graph as f or one of its callees calling g .

Author(s)

Luke Tierney

References

User manual for gprof, the GNU profiler.

cgprof: <http://mvertes.free.fr/>

Graphviz: <http://www.research.att.com/sw/tools/graphviz/>

See Also

[Rprof](#), [summaryRprof](#), [flatProfile](#), [readProfileData](#), [plotProfileCallGraph](#), [profileCallGraph2Dot](#)

Examples

```
pd <- readProfileData(system.file("samples", "glmEx.out", package="proftools"))
printProfileCallGraph(pd)
## Not run:
## If you have graphviz and cgprof installed on a UNIX-like system
## then in R do:

pd <- readProfileData(system.file("samples", "glmEx.out", package="proftools"))
printProfileCallGraph(pd, "foo.graph")

## and then in a shell do (to use the interactive dotty):

cgprof -TX foo.graph

## or (to create a postscript version and view with gv):

cgprof -Tps foo.graph > foo.ps
gv foo.ps
```

```
## End(Not run)
```

```
profileCallGraph2Dot Write Call Graph for Rprof Profile Data to Graphviz Dot File
```

Description

Prints a Graphviz .dot file representation of the call graph for profile data produced by Rprof.

Usage

```
profileCallGraph2Dot(pd, score = c("none", "total", "self"),
  transfer = function(x) x, nodeColorMap = NULL,
  edgeColorMap = NULL, filename = "Rprof.dot",
  landscape = FALSE, mergeCycles = FALSE,
  edgesColored = FALSE,
  rankDir = c("TB", "LR"),
  nodeDetails = FALSE, edgeDetails = FALSE,
  nodeSizeScore = c("none", "total", "self"),
  edgeSizeScore = c("none", "total"),
  center = FALSE, size, shape = "ellipse",
  layout = "dot", style)
```

Arguments

pd	profile data as returned by readProfileData.
score	character string specifying whether to use total time or self time for coloring nodes/edges; no color used if missing.
transfer	function; maps score values in unit interval to unit interval
nodeColorMap, edgeColorMap	character vectors of color specifications as produced by rainbow; transfer of score is mapped to color
filename	name of .dot file
landscape	logical; whether to add the rotate=90 option to the .dot file
mergeCycles	logical; whether to merge each cycle of recursion into a single node
edgesColored	logical; whether to color edges
rankDir	character; value to use for the rankdir= option to specify the direction that the plot is laid out using the dot layout; must be either "TB" for Top-to-Bottom or "LR" for Left-to-Right. The default value is "LR".
nodeDetails, edgeDetails	logical; whether count information should be shown.
nodeSizeScore	character; value to encode in the size of the nodes.
edgeSizeScore	character; value to encode in the width of the edges.

center	logical; whether to add the center=1 option to the .dot file.
size	character; string to add as size= option in the .dot file.
shape	character; node shape.
layout	character; layout method to use.
style	named list of values for arguments score through layout to use if not explicitly supplied.

Details

Writes the call graph as a Graphviz .dot file. Color is used to encode the fraction total or self time spent in each function or call. The scores used correspond to the values in the printed representation produced by `printProfileCallGraph`. For now, see the `gprof` manual for further details. The color encoding for a score `s` and a color map `m` is `ceiling(length(m) * transfer(s))`

A style can be specified to set options to a set of `cvalues` that work well together.

Value

Used for side effect.

Note

Because of lazy evaluation, nested calls like `f(g(x))` appear in the profile graph as `f` or one of its callees calling `g`.

Author(s)

Luke Tierney

References

User manual for `gprof`, the GNU profiler.

Graphviz: <http://www.research.att.com/sw/tools/graphviz/>

See Also

[Rprof](#), [summaryRprof](#), [readProfileData](#), [flatProfile](#), [plotProfileCallGraph](#), [printProfileCallGraph](#)
[plain.style](#) [google.style](#)

Examples

```
pd <- readProfileData(system.file("samples", "glmEx.out", package="proftools"))
tmp <- tempfile()
profileCallGraph2Dot(pd, filename = tmp)
file.show(tmp)
unlink(tmp)
```

```
## Not run:
## If you have graphviz installed on a UNIX-like system
## then in R do:
```

```
tmp.dot <- tempfile()
tmp.pdf <- tempfile()

profileCallGraph2Dot(pd, filename = tmp.dot)
system(sprintf("dot -Tpdf -o
browseURL(sprintf("file://

profileCallGraph2Dot(pd, score = "total", filename = tmp.dot)
system(sprintf("dot -Tpdf -o
browseURL(sprintf("file://

unlink(tmp.dot)
unlink(tmp.pdf)

## End(Not run)
```

readProfileData	<i>Read Rprof Profile Data</i>
-----------------	--------------------------------

Description

Reads in Rprof profile data for further processing.

Usage

```
readProfileData(filename = "Rprof.out")
```

Arguments

filename Name of a file produced by Rprof()

Details

readProfileData reads the data in the file produced by Rprof into a data structure for processing by other functions. The details of the structure are subject to change.

Value

R representation of Rprof data,

Author(s)

Luke Tierney

See Also

[Rprof](#), [summaryRprof](#), [flatProfile](#), [plotProfileCallGraph](#), [printProfileCallGraph](#), [profileCallGraph2Dot](#)

Examples

```
pd <- readProfileData(system.file("samples", "glmEx.out", package="proftools"))
flatProfile(pd)
flatProfile(pd, FALSE)
```

styles

Style Specifications for Call Graphs

Description

Styles providing coordinated settings of display parameters for the call graph display functions `plotProfileCallGraph` and `profileCallGraph2Dot`.

Usage

```
plain.style
google.style
```

Details

The `plain.style` style corresponds to the default parameter settings in the display functions. It can be used as the basis for creating a new custom style.

The `google.style` style is based on the display style used in the `pprof` tool from the Google Performance Tools suite.

Value

A list containing the following components:

<code>layout</code>	The layout method to use: One of "dot", "neato", and "twopi".
<code>score</code>	character string specifying whether to use total time or self time for coloring nodes/edges; no color used if missing.
<code>transfer</code>	function; maps score values in unit interval to unit interval
<code>nodeColorMap, edgeColorMap</code>	character vectors of color specifications as produced by <code>rainbow</code> ; transfer of score is mapped to color
<code>mergeCycles</code>	logical; whether to merge each cycle of recursion into a single node
<code>edgesColored</code>	logical; whether to color edges
<code>rankDir</code>	The direction that the plot is laid out in, one of either "TB" for Top-to-Bottom or "LR" for Left-to-Right. The default value is "LR". This argument is only useful for dot layouts.
<code>nodeDetails, edgeDetails</code>	logical; whether count information should be shown.
<code>nodeSizeScore</code>	character; value to encode in the size of the nodes.
<code>edgeSizeScore</code>	character; value to encode in the width of the edges.
<code>shape</code>	character; node shape.

Author(s)

Luke Tierney

References

<http://google-perftools.googlecode.com/svn/trunk/doc/cpuprofile.html>.

See Also

[Rprof](#), [flatProfile](#), [summaryRprof](#), [readProfileData](#), [plotProfileCallGraph](#), [printProfileCallGraph](#), [profileCallGraph2Dot](#)

Examples

```
pd <- readProfileData(system.file("samples", "glmEx.out", package="proftools"))
plotProfileCallGraph(pd, style = plain.style)
plotProfileCallGraph(pd, style = google.style)
```

Index

*Topic **programming**

- flatProfile, 1
- plotProfileCallGraph, 2
- printProfileCallGraph, 4
- profileCallGraph2Dot, 6
- readProfileData, 8
- styles, 9

*Topic **utilities**

- flatProfile, 1
- plotProfileCallGraph, 2
- printProfileCallGraph, 4
- profileCallGraph2Dot, 6
- readProfileData, 8
- styles, 9

flatProfile, 1, 4, 5, 7, 8, 10

google.style, 4, 7

google.style (styles), 9

plain.style, 4, 7

plain.style (styles), 9

plotProfileCallGraph, 2, 2, 5, 7, 8, 10

printProfileCallGraph, 2, 4, 4, 7, 8, 10

profileCallGraph2Dot, 2, 4, 5, 6, 8, 10

readProfileData, 2, 4, 5, 7, 8, 10

Rprof, 2, 4, 5, 7, 8, 10

styles, 9

summaryRprof, 2, 4, 5, 7, 8, 10