

RootFinding

Generated by Doxygen 1.5.6

Sat Apr 3 14:47:45 2010

Contents

1	RootFinding	1
2	Module Index	2
2.1	Modules	2
3	Data Structure Index	2
3.1	Data Structures	2
4	File Index	2
4.1	File List	2
5	Module Documentation	3
5.1	Mensagens de estados e erros	3
5.1.1	Define Documentation	4
5.2	Parte Generica	10
5.2.1	Typedef Documentation	12
5.2.2	Function Documentation	12
5.3	Metodo da Bissecao	15
5.3.1	Define Documentation	16
5.3.2	Typedef Documentation	16
5.3.3	Function Documentation	17
5.4	Metodo das Cordas	20
5.4.1	Define Documentation	21
5.4.2	Typedef Documentation	21
5.4.3	Function Documentation	21
5.5	Metodo de Newton-Rhapson	24
5.5.1	Define Documentation	25
5.5.2	Typedef Documentation	26
5.5.3	Function Documentation	26
5.6	Metodo de Pegaso	29
5.6.1	Define Documentation	30
5.6.2	Typedef Documentation	30
5.6.3	Function Documentation	30
6	Data Structure Documentation	33
6.1	RootFindingBase Struct Reference	33
6.1.1	Detailed Description	34

6.1.2	Member Enumeration Documentation	34
6.1.3	Field Documentation	35
6.2	RootFindingBissecas Struct Reference	36
6.2.1	Detailed Description	37
6.2.2	Member Enumeration Documentation	37
6.2.3	Field Documentation	37
6.3	RootFindingCordas Struct Reference	38
6.3.1	Detailed Description	39
6.3.2	Member Enumeration Documentation	39
6.3.3	Field Documentation	40
6.4	RootFindingNewtonRhapson Struct Reference	41
6.4.1	Detailed Description	42
6.4.2	Member Enumeration Documentation	42
6.4.3	Field Documentation	42
6.5	RootFindingPegaso Struct Reference	43
6.5.1	Detailed Description	44
6.5.2	Member Enumeration Documentation	44
6.5.3	Field Documentation	45
7	File Documentation	46
7.1	include/messages/RootFindingMessages.h File Reference	46
7.1.1	Detailed Description	46
7.2	include/messages/RootFindingMessages_PT-BR.h File Reference	46
7.2.1	Detailed Description	48
7.3	include/RootFinding.h File Reference	48
7.3.1	Detailed Description	48
7.4	include/RootFindingBase.h File Reference	48
7.4.1	Detailed Description	50
7.5	include/RootFindingBissecas.h File Reference	50
7.5.1	Detailed Description	51
7.6	include/RootFindingCommon.h File Reference	51
7.6.1	Detailed Description	52
7.6.2	Define Documentation	52
7.6.3	Typedef Documentation	53
7.6.4	Function Documentation	53
7.7	include/RootFindingCordas.h File Reference	54
7.7.1	Detailed Description	56

7.8	include/RootFindingNewtonRhapson.h File Reference	56
7.8.1	Detailed Description	57
7.9	include/RootFindingPegaso.h File Reference	57
7.9.1	Detailed Description	58
7.10	src/RootFindingBase.c File Reference	59
7.10.1	Detailed Description	60
7.10.2	Variable Documentation	60
7.11	src/RootFindingBissecas.c File Reference	60
7.11.1	Detailed Description	62
7.11.2	Variable Documentation	62
7.12	src/RootFindingCommon.c File Reference	62
7.12.1	Detailed Description	63
7.12.2	Function Documentation	63
7.13	src/RootFindingCordas.c File Reference	64
7.13.1	Detailed Description	66
7.13.2	Variable Documentation	66
7.14	src/RootFindingNewtonRhapson.c File Reference	66
7.14.1	Detailed Description	67
7.14.2	Variable Documentation	67
7.15	src/RootFindingPegaso.c File Reference	68
7.15.1	Detailed Description	69
7.15.2	Variable Documentation	69

1 RootFinding

Implementação de metodos de Calculo Numerico para encontrar raizes de funcoes matematicas.

Projeto hospedado no google code: <http://code.google.com/p/root-finding/>

Author:

Matheus Neder <matheusneder@gmail.com>

Metodos implementados:

- Bissecas
- Cordas (Secante)
- Newton-Rhapson
- Pegaso

Dependencias:

- [muParser](#) versao >= 1.32

2 Module Index

2.1 Modules

Here is a list of all modules:

Mensagens de estados e erros	3
Parte Generica	10
Metodo da Bissecacao	15
Metodo das Cordas	20
Metodo de Newton-Rhapson	24
Metodo de Pegaso	29

3 Data Structure Index

3.1 Data Structures

Here are the data structures with brief descriptions:

RootFindingBase (Estrutura de dados para RootFindingBase)	33
RootFindingBissecacao (Estrutura de dados para o Metodo da Bissecacao)	36
RootFindingCordas (Estrutura de dados para o Metodo das Cordas)	38
RootFindingNewtonRhapson (Estrutura de dados para o Metodo de Newton-Rhapson)	41
RootFindingPegaso (Estrutura de dados para o Metodo de Pegaso)	43

4 File Index

4.1 File List

Here is a list of all files with brief descriptions:

include/RootFinding.h (/root-finding/include/RootFinding.h)	48
include/RootFindingBase.h (/root-finding/include/RootFindingBase.h)	48
include/RootFindingBissecacao.h (/root-finding/include/RootFindingBissecacao.h)	50
include/RootFindingCommon.h (/root-finding/include/RootFindingCommon.h)	51
include/RootFindingCordas.h (/root-finding/include/RootFindingCordas.h)	54
include/RootFindingNewtonRhapson.h (/root-finding/include/RootFindingNewtonRhapson.h)	56

include/RootFindingPegaso.h (/root-finding/include/RootFindingPegaso.h)	57
include/messages/RootFindingMessages.h (/root-finding/include/messages/RootFindingMessages.h)	46
include/messages/RootFindingMessages_PT-BR.h (/root-finding/include/messages/RootFindingMessages_PT-BR.h)	46
src/RootFindingBase.c (/root-finding/src/RootFindingBase.c)	59
src/RootFindingBissecacao.c (/root-finding/src/RootFindingBissecacao.c)	60
src/RootFindingCommon.c (/root-finding/src/RootFindingCommon.c)	62
src/RootFindingCordas.c (/root-finding/src/RootFindingCordas.c)	64
src/RootFindingNewtonRhapson.c (/root-finding/src/RootFindingNewtonRhapson.c)	66
src/RootFindingPegaso.c (/root-finding/src/RootFindingPegaso.c)	68

5 Module Documentation

5.1 Mensagens de estados e erros

Defines

- `#define MSG_ROOTS_UNKNOW_ERROR` "Erro desconhecido."
- `#define MSG_ROOTS_UNKNOW_STATE` "Estado desconhecido."
- `#define MSG_GENERIC_APPROXIMATION_ROOT_FOUND`
- `#define MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED`
- `#define MSG_GENERIC_NO_ERROR` "Nao ocorreu erro."
- `#define MSG_GENERIC_ERROR_FOUND` "Ocorreu erro.";
- `#define MSG_GENERIC_NOT_INIT` "Nao inicializado."
- `#define MSG_GENERIC_INITIALIZED` "Inicializado."
- `#define MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR`
- `#define MSG_GENERIC_2NDIFF_TEST_ERROR`
- `#define MSG_ROOTS_NO_ERROR` `MSG_GENERIC_NO_ERROR`
- `#define MSG_ROOTS_MUP_EVAL_ERROR` "Erro c/ expressao do objeto muParser"
- `#define MSG_ROOTS_RANGE_ERROR_PROD_FA_FB_NOT_LT_0`
- `#define MSG_ROOTS_RANGE_ERROR_FA_OR_FB_ISINFINITY`
- `#define MSG_ROOTS_RANGE_NOT_SET` "Intervalo nao esta definido."
- `#define MSG_ROOTS_READY` "Objeto `RootFindingBase` esta pronto."
- `#define MSG_ROOTS_EXACT_ROOT_FOUND` "Encontrado a raiz exata: %lf."
- `#define MSG_BISSECAO_NO_ERROR` `MSG_GENERIC_NO_ERROR`
- `#define MSG_BISSECAO_X_ISINF_OR_ISNAN_ERROR` `MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR`
- `#define MSG_BISSECAO_NOT_INIT` `MSG_GENERIC_NOT_INIT`
- `#define MSG_BISSECAO_INITIALIZED` `MSG_GENERIC_INITIALIZED`
- `#define MSG_BISSECAO_MAX_ITERATIONS_LIMIT_REACHED` `MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED`
- `#define MSG_BISSECAO_APPROXIMATION_ROOT_FOUND` `MSG_GENERIC_APPROXIMATION_ROOT_FOUND`

- #define [MSG_BISSECAO_ERROR_FOUND](#) MSG_GENERIC_ERROR_FOUND
- #define [MSG_CORDAS_NO_ERROR](#) MSG_GENERIC_NO_ERROR
- #define [MSG_CORDAS_2NDIFF_TEST_ERROR](#) MSG_GENERIC_2NDIFF_TEST_ERROR
- #define [MSG_CORDAS_X_ISINF_OR_ISNAN_ERROR](#) MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR
- #define [MSG_CORDAS_NOT_INIT](#) MSG_GENERIC_NOT_INIT
- #define [MSG_CORDAS_INITIALIZED](#) MSG_GENERIC_INITIALIZED
- #define [MSG_CORDAS_MAX_ITERATIONS_LIMIT_REACHED](#) MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED
- #define [MSG_CORDAS_APPROXIMATION_ROOT_FOUND](#) MSG_GENERIC_APPROXIMATION_ROOT_FOUND
- #define [MSG_CORDAS_ERROR_FOUND](#) MSG_GENERIC_ERROR_FOUND
- #define [MSG_NEWTON_NO_ERROR](#) MSG_GENERIC_NO_ERROR
- #define [MSG_NEWTON_2NDIFF_TEST_ERROR](#) MSG_GENERIC_2NDIFF_TEST_ERROR
- #define [MSG_NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_EQUALS_ERROR](#)
- #define [MSG_NEWTON_X_ISINF_OR_ISNAN_ERROR](#)
- #define [MSG_NEWTON_NOT_INIT](#) MSG_GENERIC_NOT_INIT
- #define [MSG_NEWTON_INITIALIZED](#) MSG_GENERIC_INITIALIZED
- #define [MSG_NEWTON_MAX_ITERATIONS_LIMIT_REACHED](#) MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED
- #define [MSG_NEWTON_APPROXIMATION_ROOT_FOUND](#) MSG_GENERIC_APPROXIMATION_ROOT_FOUND
- #define [MSG_NEWTON_ERROR_FOUND](#) MSG_GENERIC_ERROR_FOUND
- #define [MSG_PEGASO_NO_ERROR](#) MSG_GENERIC_NO_ERROR
- #define [MSG_PEGASO_X_ISINF_OR_ISNAN_ERROR](#) MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR
- #define [MSG_PEGASO_NOT_INIT](#) MSG_GENERIC_NOT_INIT
- #define [MSG_PEGASO_INITIALIZED](#) MSG_GENERIC_INITIALIZED
- #define [MSG_PEGASO_MAX_ITERATIONS_LIMIT_REACHED](#) MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED
- #define [MSG_PEGASO_APPROXIMATION_ROOT_FOUND](#) MSG_GENERIC_APPROXIMATION_ROOT_FOUND
- #define [MSG_PEGASO_ERROR_FOUND](#) MSG_GENERIC_ERROR_FOUND

5.1.1 Define Documentation

5.1.1.1 #define MSG_BISSECAO_APPROXIMATION_ROOT_FOUND MSG_GENERIC_APPROXIMATION_ROOT_FOUND

Definition at line 83 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBissecaoGetStateMessage().

5.1.1.2 #define MSG_BISSECAO_ERROR_FOUND MSG_GENERIC_ERROR_FOUND

Definition at line 84 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBissecaoGetStateMessage().

5.1.1.3 #define MSG_BISSECAO_INITIALIZED MSG_GENERIC_INITIALIZED

Definition at line 81 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBissecaoGetStateMessage().

5.1.1.4 #define MSG_BISSECAO_MAX_ITERATIONS_LIMIT_REACHED MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED

Definition at line 82 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBissecaoGetStateMessage().

5.1.1.5 #define MSG_BISSECAO_NO_ERROR MSG_GENERIC_NO_ERROR

Definition at line 76 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBissecaoGetErrorMessage().

5.1.1.6 #define MSG_BISSECAO_NOT_INIT MSG_GENERIC_NOT_INIT

Definition at line 80 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBissecaoGetStateMessage().

5.1.1.7 #define MSG_BISSECAO_X_ISINF_OR_ISNAN_ERROR MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR

Definition at line 77 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBissecaoGetErrorMessage().

5.1.1.8 #define MSG_CORDAS_2NDIFF_TEST_ERROR MSG_GENERIC_2NDIFF_TEST_ERROR

Definition at line 89 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetErrorMessage().

5.1.1.9 #define MSG_CORDAS_APPROXIMATION_ROOT_FOUND MSG_GENERIC_APPROXIMATION_ROOT_FOUND

Definition at line 97 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetStateMessage().

5.1.1.10 #define MSG_CORDAS_ERROR_FOUND MSG_GENERIC_ERROR_FOUND

Definition at line 98 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetStateMessage().

5.1.1.11 #define MSG_CORDAS_INITIALIZED MSG_GENERIC_INITIALIZED

Definition at line 95 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetStateMessage().

5.1.1.12 #define MSG_CORDAS_MAX_ITERATIONS_LIMIT_REACHED MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED

Definition at line 96 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetStateMessage().

5.1.1.13 #define MSG_CORDAS_NO_ERROR MSG_GENERIC_NO_ERROR

Definition at line 87 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetErrorMessage().

5.1.1.14 #define MSG_CORDAS_NOT_INIT MSG_GENERIC_NOT_INIT

Definition at line 94 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetStateMessage().

5.1.1.15 #define MSG_CORDAS_X_ISINF_OR_ISNAN_ERROR MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR

Definition at line 91 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingCordasGetErrorMessage().

5.1.1.16 #define MSG_GENERIC_2NDIFF_TEST_ERROR

Value:

```
"O teste da segunda derivada "\
                                     "falhou para ambos os extremos"\
                                     "do intervalo [%lf, %lf]."
```

Definition at line 56 of file RootFindingMessages_PT-BR.h.

5.1.1.17 #define MSG_GENERIC_APPROXIMATION_ROOT_FOUND

Value:

```
"Encontrado uma aproximacao da raiz: %lf "\
                                     "(com e = %lg)."
```

Definition at line 38 of file RootFindingMessages_PT-BR.h.

5.1.1.18 #define MSG_GENERIC_ERROR_FOUND "Ocorreu erro.";

Definition at line 49 of file RootFindingMessages_PT-BR.h.

5.1.1.19 #define MSG_GENERIC_INITIALIZED "Inicializado."

Definition at line 51 of file RootFindingMessages_PT-BR.h.

5.1.1.20 #define MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED

Value:

```
"Nao pode ser encontrado "\
                                     "uma aproximacao "\
                                     "(com t <= %lg) para o limite "\
                                     "de iteracoes igual a %u.\n"\
                                     "\n"\
                                     "O valor obtido foi: %lf (com e = %lg)."
```

Definition at line 41 of file RootFindingMessages_PT-BR.h.

5.1.1.21 #define MSG_GENERIC_NO_ERROR "Nao ocorreu erro."

Definition at line 48 of file RootFindingMessages_PT-BR.h.

5.1.1.22 #define MSG_GENERIC_NOT_INIT "Nao inicializado."

Definition at line 50 of file RootFindingMessages_PT-BR.h.

5.1.1.23 #define MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR

Value:

```
"Ocorreu uma indeterminacao do "\
                                     "tipo (0/0, inf/inf ...) durante o processo."
```

Definition at line 53 of file RootFindingMessages_PT-BR.h.

5.1.1.24 #define MSG_NEWTON_2NDIFF_TEST_ERROR MSG_GENERIC_2NDIFF_TEST_ERROR

Definition at line 103 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetErrorMessage().

5.1.1.25 #define MSG_NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_EQUALS_ERROR

Value:

```
"O sinal da segunda derivada nos extremos "\
                                     "do intervalo [%lf, %lf] nao se conserva."
```

Definition at line 105 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetErrorMessage().

5.1.1.26 #define MSG_NEWTON_APPROXIMATION_ROOT_FOUND MSG_GENERIC_APPROXIMATION_ROOT_FOUND

Definition at line 124 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetStateMessage().

5.1.1.27 #define MSG_NEWTON_ERROR_FOUND MSG_GENERIC_ERROR_FOUND

Definition at line 125 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetStateMessage().

5.1.1.28 #define MSG_NEWTON_INITIALIZED MSG_GENERIC_INITIALIZED

Definition at line 122 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetStateMessage().

5.1.1.29 #define MSG_NEWTON_MAX_ITERATIONS_LIMIT_REACHED MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED

Definition at line 123 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetStateMessage().

5.1.1.30 #define MSG_NEWTON_NO_ERROR MSG_GENERIC_NO_ERROR

Definition at line 101 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetErrorMessage().

5.1.1.31 #define MSG_NEWTON_NOT_INIT MSG_GENERIC_NOT_INIT

Definition at line 121 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetStateMessage().

5.1.1.32 #define MSG_NEWTON_X_ISINF_OR_ISNAN_ERROR

Value:

```
"Nao pode ser encontrada uma aproximacao "\n\n"para a raiz. A causa mais provavel "\n"eh de que a derivada da funcao "\n"no ponto da iteracao imediatamente posterior "\n"a do ponto '%lf' "\n"seja igual a 0.\n"\n\n"O valor encontrado ateh a "\n"iteracao onde ocorreu o problema foi:"\n"%lf (com e = %lg)."
```

Definition at line 109 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingNewtonRhapsonGetErrorMessage().

5.1.1.33 #define MSG_PEGASO_APPROXIMATION_ROOT_FOUND MSG_GENERIC_APPROXIMATION_ROOT_FOUND

Definition at line 135 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingPegasoGetStateMessage().

5.1.1.34 #define MSG_PEGASO_ERROR_FOUND MSG_GENERIC_ERROR_FOUND

Definition at line 136 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingPegasoGetStateMessage().

5.1.1.35 #define MSG_PEGASO_INITIALIZED MSG_GENERIC_INITIALIZED

Definition at line 133 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingPegasoGetStateMessage().

5.1.1.36 #define MSG_PEGASO_MAX_ITERATIONS_LIMIT_REACHED MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED

Definition at line 134 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingPegasoGetStateMessage().

5.1.1.37 #define MSG_PEGASO_NO_ERROR MSG_GENERIC_NO_ERROR

Definition at line 128 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingPegasoGetErrorMessage().

5.1.1.38 #define MSG_PEGASO_NOT_INIT MSG_GENERIC_NOT_INIT

Definition at line 132 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingPegasoGetStateMessage().

5.1.1.39 #define MSG_PEGASO_X_ISINF_OR_ISNAN_ERROR MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR

Definition at line 129 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingPegasoGetErrorMessage().

5.1.1.40 #define MSG_ROOTS_EXACT_ROOT_FOUND "Encontrado a raiz exata: %lf."

Definition at line 73 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetStateMessage().

5.1.1.41 #define MSG_ROOTS_MUP_EVAL_ERROR "Erro c/ expressao do objeto muParser"

Definition at line 62 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetErrorMessage().

5.1.1.42 #define MSG_ROOTS_NO_ERROR MSG_GENERIC_NO_ERROR

Definition at line 61 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetErrorMessage().

5.1.1.43 #define MSG_ROOTS_RANGE_ERROR_FA_OR_FB_ISINFINITY

Value:

```
"A funcao nao esta definida para um ou ambos "\
                                         "dos extremos de [%lf, %lf]."
```

Definition at line 67 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetErrorMessage().

5.1.1.44 #define MSG_ROOTS_RANGE_ERROR_PROD_FA_FB_NOT_LT_0

Value:

```
"f(%lf) * f(%lf) >= 0, portanto nao eh possivel "\
                                         "procurar raizes no intervalo."
```

Definition at line 64 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetErrorMessage().

5.1.1.45 #define MSG_ROOTS_RANGE_NOT_SET "Intervalo nao esta definido."

Definition at line 71 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetStateMessage().

5.1.1.46 #define MSG_ROOTS_READY "Objeto RootFindingBase esta pronto."

Definition at line 72 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetStateMessage().

5.1.1.47 #define MSG_ROOTS_UNKNOW_ERROR "Erro desconhecido."

Definition at line 34 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetErrorMessage(), RootFindingBissecaoGetErrorMessage(), RootFindingCordasGetErrorMessage(), RootFindingNewtonRhapsonGetErrorMessage(), and RootFindingPegasoGetErrorMessage().

5.1.1.48 #define MSG_ROOTS_UNKNOW_STATE "Estado desconhecido."

Definition at line 35 of file RootFindingMessages_PT-BR.h.

Referenced by RootFindingBaseGetStateMessage(), RootFindingBissecaoGetStateMessage(), RootFindingCordasGetStateMessage(), RootFindingNewtonRhapsonGetStateMessage(), and RootFindingPegasoGetStateMessage().

5.2 Parte Generica

Data Structures

- struct [RootFindingBase](#)
Estrutura de dados para [RootFindingBase](#).

Typedefs

- typedef struct [RootFindingBase](#) [RootFindingBaseT](#)
Apelido para struct [RootFindingBase](#).

Functions

- `RootFindingBaseT * RootFindingBaseCreate` (`muParserHandle_t mupObj`, `RootFindingDoubleT *varPtr`)
Cria o objeto `RootFindingBase`.
- `void RootFindingBaseDelete` (`RootFindingBaseT *obj`)
Apaga o objeto `RootFindingBase`.
- `RootFindingBoolT RootFindingBaseSetRange` (`RootFindingBaseT *rootsObj`, `RootFindingDoubleT a`, `RootFindingDoubleT b`)
Define o intervalo para procura da raiz.
- `RootFindingDoubleT RootFindingBaseEval` (`RootFindingBaseT *rootsObj`, `RootFindingDoubleT value`)
Avalia a funcao em um ponto.
- `RootFindingDoubleT RootFindingBase2nDiffEval` (`RootFindingBaseT *rootsObj`, `RootFindingDoubleT value`)
Avalia a 2a. diferencial da funcao em um ponto.
- `RootFindingDoubleT RootFindingBaseDiffEval` (`RootFindingBaseT *rootsObj`, `RootFindingDoubleT value`)
Avalia a diferencial da funcao em um ponto.
- `void RootFindingBaseReset` (`RootFindingBaseT *rootsObj`)
Reinicializa o objeto struct `RootFindingBase`.
- `int RootFindingBaseGetErrorCode` (`RootFindingBaseT *rootsObj`)
Obtem o codigo de erro.
- `int RootFindingBaseGetStateCode` (`RootFindingBaseT *rootsObj`)
Obtem o codigo referente ao estado do objeto.
- `const char * RootFindingBaseGetErrorMessage` (`RootFindingBaseT *rootsObj`)
Obtem a mensagem de erro.
- `const char * RootFindingBaseGetStateMessage` (`RootFindingBaseT *rootsObj`)
Obtem a mensagem referente ao estado do objeto.
- `RootFindingBoolT RootFindingBaseHasError` (`RootFindingBaseT *rootsObj`)
Verifica se ha erros.
- `static RootFindingBoolT isRangeError` (`RootFindingBaseT *rootsObj`)
Verify if `errorCode` is a `RangeError`.

5.2.1 Typedef Documentation

5.2.1.1 typedef struct RootFindingBase RootFindingBaseT

Apelido para struct [RootFindingBase](#).

Definition at line 83 of file RootFindingBase.h.

5.2.2 Function Documentation

5.2.2.1 static RootFindingBoolT isRangeError (RootFindingBaseT * *rootsObj*) [static, private]

Verify if errorCode is a RangeError.

Definition at line 71 of file RootFindingBase.c.

References RootFindingBase::errorCode.

Referenced by RootFindingBaseSetRange().

5.2.2.2 RootFindingDoubleT RootFindingBase2ndDiffEval (RootFindingBaseT * *rootsObj*, RootFindingDoubleT *value*)

Avalia a 2a. diferencial da funcao em um ponto.

Parameters:

rootsObj Ponteiro para o objeto struct [RootFindingBase](#)
value Valor do ponto a ser avaliado

Returns:

Avaliacao da 2a. dif da funcao no ponto

Definition at line 117 of file RootFindingBase.c.

References Mup2ndDiff(), RootFindingBase::mupObj, and RootFindingBase::varPtr.

Referenced by RootFindingCordasInit(), and RootFindingNewtonRhapsonInit().

5.2.2.3 RootFindingBaseT* RootFindingBaseCreate (muParserHandle_t *mupObj*, RootFindingDoubleT * *varPtr*)

Cria o objeto [RootFindingBase](#).

Parameters:

mupObj Ponteiro para o objeto muParser contendo a expressao
varPtr Pontero para a variavel relacionada a expressao no objeto muParser referente a qual eixo deve se procurar a raiz

Returns:

Ponteiro para o objeto criado

Definition at line 32 of file RootFindingBase.c.

References RootFindingBase::mupObj, RootFindingBaseReset(), RootFindingPegaso::rootsObj, and RootFindingBase::varPtr.

5.2.2.4 void RootFindingBaseDelete (RootFindingBaseT * *obj*)

Apaga o objeto [RootFindingBase](#).

Parameters:

obj Ponteiro para o objeto [RootFindingBase](#)

Definition at line 61 of file RootFindingBase.c.

5.2.2.5 RootFindingDoubleT RootFindingBaseDiffEval (RootFindingBaseT * *rootsObj*, RootFindingDoubleT *value*)

Avalia a diferencial da funcao em um ponto.

Parameters:

rootsObj Ponteiro para o objeto struct [RootFindingBase](#)

value Valor do ponto a ser avaliado

Returns:

Avaliacao da dif da funcao no ponto

Definition at line 124 of file RootFindingBase.c.

References MupDiff(), RootFindingBase::mupObj, and RootFindingBase::varPtr.

Referenced by getNextX().

5.2.2.6 RootFindingDoubleT RootFindingBaseEval (RootFindingBaseT * *rootsObj*, RootFindingDoubleT *value*)

Avalia a funcao em um ponto.

Parameters:

rootsObj Ponteiro para o objeto struct [RootFindingBase](#)

value Valor do ponto a ser avaliado

Returns:

Avaliacao da funcao no ponto

Definition at line 110 of file RootFindingBase.c.

References RootFindingBase::mupObj, and RootFindingBase::varPtr.

Referenced by RootFindingBaseReset(), RootFindingBaseSetRange(), RootFindingBissecacaoFindNewRange(), RootFindingBissecacaoInit(), RootFindingBissecacaoPerformIteration(), RootFindingCordasInit(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonInit(), RootFindingNewtonRhapsonPerformIteration(), RootFindingPegasoInit(), and RootFindingPegasoPerformIteration().

5.2.2.7 int RootFindingBaseGetErrorCode (RootFindingBaseT * *rootsObj*)

Obtem o codigo de erro.

Definition at line 131 of file RootFindingBase.c.

References RootFindingBase::errorCode.

5.2.2.8 const char* RootFindingBaseGetErrorMessage (RootFindingBaseT * rootsObj)

Obtem a mensagem de erro.

Definition at line 143 of file RootFindingBase.c.

References RootFindingBase::a, RootFindingBase::b, RootFindingBase::errorCode, msg, MSG_ROOTS_MUP_EVAL_ERROR, MSG_ROOTS_NO_ERROR, MSG_ROOTS_RANGE_ERROR_FA_OR_FB_ISINFINITY, MSG_ROOTS_RANGE_ERROR_PROD_FA_FB_NOT_LT_0, and MSG_ROOTS_UNKNOW_ERROR.

5.2.2.9 int RootFindingBaseGetStateCode (RootFindingBaseT * rootsObj)

Obtem o codigo referente ao estado do objeto.

Definition at line 136 of file RootFindingBase.c.

References RootFindingBase::state.

5.2.2.10 const char* RootFindingBaseGetStateMessage (RootFindingBaseT * rootsObj)

Obtem a mensagem referente ao estado do objeto.

Definition at line 164 of file RootFindingBase.c.

References msg, MSG_ROOTS_EXACT_ROOT_FOUND, MSG_ROOTS_RANGE_NOT_SET, MSG_ROOTS_READY, MSG_ROOTS_UNKNOW_STATE, RootFindingBase::state, and RootFindingBase::x.

5.2.2.11 RootFindingBoolT RootFindingBaseHasError (RootFindingBaseT * rootsObj)

Verifica se ha erros.

Returns:

TRUE caso haja erro

Definition at line 180 of file RootFindingBase.c.

References RootFindingBase::errorCode.

Referenced by RootFindingBaseSetRange().

5.2.2.12 void RootFindingBaseReset (RootFindingBaseT * rootsObj)

Reinicializa o objeto struct [RootFindingBase](#).

Parameters:

rootsObj Ponteiro para o objeto struct [RootFindingBase](#)

Definition at line 47 of file RootFindingBase.c.

References RootFindingBase::e, RootFindingBase::errorCode, infinity(), RootFindingBase::mupObj, RootFindingBaseEval(), and RootFindingBase::state.

Referenced by RootFindingBaseCreate().

5.2.2.13 RootFindingBoolT RootFindingBaseSetRange (RootFindingBaseT * rootsObj, RootFindingDoubleT a, RootFindingDoubleT b)

Define o intervalo para procura da raiz.

Parameters:

rootsObj Ponteiro para o objeto [RootFindingBase](#)

a

b

Returns:

TRUE em caso de sucesso, quando $f(a) * f(b) < 0$ e $f(a)$, $f(b)$ sejam ambos diferentes de infinito

Definition at line 77 of file RootFindingBase.c.

References [RootFindingBase::a](#), [RootFindingBase::b](#), [RootFindingBase::errorCode](#), [FALSE](#), [isRangeError\(\)](#), [RootFindingBaseEval\(\)](#), [RootFindingBaseHasError\(\)](#), [RootFindingBase::state](#), and [TRUE](#).

5.3 Metodo da Bissecao

Data Structures

- struct [RootFindingBissecao](#)
Estrutura de dados para o Metodo da Bissecao.

Defines

- #define [BISSECAO_DEFAULT_MAX_ITERATIONS](#) 100
- #define [BISSECAO_DEFAULT_TOLERANCE](#) 1e-7

Typedefs

- typedef struct [RootFindingBissecao](#) [RootFindingBissecaoT](#)
Apelido para struct [RootFindingBissecao](#).

Functions

- [RootFindingBissecaoT * RootFindingBissecaoCreate](#) ([RootFindingBaseT * rootsObj](#))
Cria um objeto do tipo struct [RootFindingBissecao](#).
- [RootFindingBoolT RootFindingBissecaoInit](#) ([RootFindingBissecaoT * bissecaoObj](#))
Inicializa o objeto [RootFindingBissecao](#).
- void [RootFindingBissecaoDelete](#) ([RootFindingBissecaoT * obj](#))
Apaga a instancia do objeto [RootFindingBissecao](#).
- [RootFindingBoolT RootFindingBissecaoPerformIteration](#) ([RootFindingBissecaoT * bissecaoObj](#))

Realiza a iteracao.

- void [RootFindingBissecaoFindNewRange](#) ([RootFindingBaseT](#) *rootsObj)
Encontra um novo intervalo $[A, B]$ e os altera no objeto [RootFindingBaseT](#) baseado nos $[A, B]$ e x existentes. Utilizado em [RootFindingBissecaoPerformIteration](#) porem principalmente util para alterar o intervalo quando intercambiando entre metodos diferentes.
- int [RootFindingBissecaoGetErrorCode](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem o codigo de erro.
- int [RootFindingBissecaoGetStateCode](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingBissecaoGetErrorMessage](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem a mensagem de erro.
- const char * [RootFindingBissecaoGetStateMessage](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingBissecaoHasError](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Verifica se ha erros.
- static void [setError](#) ([RootFindingBissecaoT](#) *bissecaoObj, int errorCode)
Set error code and change state to `BISSECAO_ERROR_FOUND`.
- static void [resetError](#) ([RootFindingBissecaoT](#) *bissecaoObj)
- static [RootFindingDoubleT](#) [computeX](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Calcula o X baseado no intervalo $[a, b]$.

5.3.1 Define Documentation

5.3.1.1 #define BISSECAO_DEFAULT_MAX_ITERATIONS 100

Definition at line 34 of file [RootFindingBissecao.h](#).

Referenced by [RootFindingBissecaoCreate\(\)](#).

5.3.1.2 #define BISSECAO_DEFAULT_TOLERANCE 1e-7

Definition at line 35 of file [RootFindingBissecao.h](#).

Referenced by [RootFindingBissecaoCreate\(\)](#).

5.3.2 Typedef Documentation

5.3.2.1 typedef struct RootFindingBissecao RootFindingBissecaoT

Apelido para struct [RootFindingBissecao](#).

Definition at line 77 of file [RootFindingBissecao.h](#).

5.3.3 Function Documentation

5.3.3.1 static RootFindingDoubleT computeX (RootFindingBissecacaoT * *bissecacaoObj*) [static, private]

Calcula o X baseado no intervalo [a, b].

Parameters:

bissecacaoObj Ponteiro para o objeto [RootFindingBissecacao](#)

Returns:

O X calculado

Definition at line 70 of file RootFindingBissecacao.c.

References RootFindingBase::a, RootFindingBase::b, and RootFindingBissecacao::rootsObj.

Referenced by RootFindingBissecacaoInit(), and RootFindingBissecacaoPerformIteration().

5.3.3.2 static void resetError (RootFindingBissecacaoT * *bissecacaoObj*) [static, private]

Definition at line 164 of file RootFindingBissecacao.c.

References RootFindingBissecacao::errorCode, and RootFindingBissecacao::state.

Referenced by RootFindingBissecacaoCreate(), RootFindingBissecacaoInit(), RootFindingCordasCreate(), RootFindingCordasInit(), RootFindingNewtonRhapsonCreate(), RootFindingPegasoCreate(), and RootFindingPegasoInit().

5.3.3.3 RootFindingBissecacaoT* RootFindingBissecacaoCreate (RootFindingBaseT * *rootsObj*)

Cria um objeto do tipo struct [RootFindingBissecacao](#).

Parameters:

rootsObj Ponteiro para o objeto do tipo struct [RootFindingBase](#)

Returns:

Ponteiro para o objeto criado

Definition at line 47 of file RootFindingBissecacao.c.

References BISSECAO_DEFAULT_MAX_ITERATIONS, BISSECAO_DEFAULT_TOLERANCE, RootFindingBissecacao::maxIterations, resetError(), RootFindingBissecacao::rootsObj, RootFindingBissecacao::state, and RootFindingBissecacao::tolerance.

5.3.3.4 void RootFindingBissecacaoDelete (RootFindingBissecacaoT * *obj*)

Apaga a instancia do objeto [RootFindingBissecacao](#).

Parameters:

obj Ponteiro para o obj [RootFindingBissecacao](#)

Definition at line 90 of file RootFindingBissecacao.c.

5.3.3.5 void RootFindingBissecaoFindNewRange (RootFindingBaseT * rootsObj)

Encontra um novo intervalo [A, B] e os altera no objeto RootFindingBaseT baseado nos [A, B] e x existentes. Utilizado em [RootFindingBissecaoPerformIteration](#) porem principalmente util para alterar o intervalo quando intercambiando entre metodos diferentes.

Parameters:

rootsObj Ponteiro para o objeto [RootFindingBase](#)

Definition at line 139 of file RootFindingBissecao.c.

References RootFindingBase::a, RootFindingBase::b, RootFindingBase::fX, RootFindingBaseEval(), and RootFindingBase::x.

Referenced by RootFindingBissecaoPerformIteration().

5.3.3.6 int RootFindingBissecaoGetErrorCode (RootFindingBissecaoT * bissecaoObj)

Obtem o codigo de erro.

Definition at line 159 of file RootFindingBissecao.c.

References RootFindingBissecao::errorCode.

5.3.3.7 const char* RootFindingBissecaoGetErrorMessage (RootFindingBissecaoT * bissecaoObj)

Obtem a mensagem de erro.

Definition at line 180 of file RootFindingBissecao.c.

References RootFindingBissecao::errorCode, MSG_BISSECAO_NO_ERROR, MSG_BISSECAO_X_ISINF_OR_ISNAN_ERROR, and MSG_ROOTS_UNKNOW_ERROR.

5.3.3.8 int RootFindingBissecaoGetStateCode (RootFindingBissecaoT * bissecaoObj)

Obtem o codigo referente ao estado do objeto.

Definition at line 173 of file RootFindingBissecao.c.

References RootFindingBissecao::state.

5.3.3.9 const char* RootFindingBissecaoGetStateMessage (RootFindingBissecaoT * bissecaoObj)

Obtem a mensagem referente ao estado do objeto.

Definition at line 194 of file RootFindingBissecao.c.

References RootFindingBase::e, RootFindingBissecao::maxIterations, msg, MSG_BISSECAO_APPROXIMATION_ROOT_FOUND, MSG_BISSECAO_ERROR_FOUND, MSG_BISSECAO_INITIALIZED, MSG_BISSECAO_MAX_ITERATIONS_LIMIT_REACHED, MSG_BISSECAO_NOT_INIT, MSG_ROOTS_UNKNOW_STATE, RootFindingBissecao::rootsObj, RootFindingBissecao::state, RootFindingBissecao::tolerance, and RootFindingBase::x.

5.3.3.10 RootFindingBoolT RootFindingBissecaoHasError (RootFindingBissecaoT * bissecaoObj)

Verifica se ha erros.

Returns:

TRUE caso haja erro

Definition at line 220 of file RootFindingBissecao.c.

References RootFindingBissecao::errorCode.

5.3.3.11 RootFindingBoolT RootFindingBissecaoInit (RootFindingBissecaoT * bissecaoObj)

Inicializa o objeto [RootFindingBissecao](#).

Parameters:

bissecaoObj Ponteiro para o objeto a ser inicializado

Returns:

TRUE em caso de sucesso

Definition at line 77 of file RootFindingBissecao.c.

References computeX(), RootFindingBase::e, RootFindingBase::fX, RootFindingBissecao::i, infinity(), resetError(), RootFindingBaseEval(), RootFindingBissecao::rootsObj, RootFindingBissecao::state, TRUE, and RootFindingBase::x.

5.3.3.12 RootFindingBoolT RootFindingBissecaoPerformIteration (RootFindingBissecaoT * bissecaoObj)

Realiza a iteracao.

Parameters:

bissecaoObj Ponteiro para o objeto

Returns:

TRUE caso haja mais iteracoes a serem realizadas

Definition at line 95 of file RootFindingBissecao.c.

References computeX(), RootFindingBase::e, FALSE, RootFindingBase::fX, RootFindingBissecao::i, isInfOrNan(), RootFindingBissecao::maxIterations, RootFindingBaseEval(), RootFindingBissecaoFindNewRange(), RootFindingBissecao::rootsObj, setError(), RootFindingBissecao::state, RootFindingBase::state, RootFindingBissecao::tolerance, TRUE, and RootFindingBase::x.

5.3.3.13 static void setError (RootFindingBissecaoT * bissecaoObj, int errorCode) [static, private]

Set error code and change state to BISSECAO_ERROR_FOUND.

Parameters:

bissecaoObj Ponteiro para objeto RootFindingBissecaoT

errorCode Código de erro

Definition at line 153 of file RootFindingBissecao.c.

References RootFindingBissecao::errorCode, and RootFindingBissecao::state.

Referenced by RootFindingBissecaoPerformIteration(), RootFindingCordasInit(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonInit(), RootFindingNewtonRhapsonPerformIteration(), and RootFindingPegasoPerformIteration().

5.4 Metodo das Cordas

Data Structures

- struct [RootFindingCordas](#)

Estrutura de dados para o Metodo das Cordas.

Defines

- #define [CORDAS_DEFAULT_MAX_ITERATIONS](#) 100
- #define [CORDAS_DEFAULT_TOLERANCE](#) 1e-7

Typedefs

- typedef struct [RootFindingCordas](#) [RootFindingCordasT](#)

Apelido para struct [RootFindingCordas](#).

Functions

- [RootFindingCordasT * RootFindingCordasCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingCordas](#).
- [RootFindingBoolT RootFindingCordasInit](#) ([RootFindingCordasT](#) *cordasObj)
Inicializa o objeto [RootFindingCordas](#).
- void [RootFindingCordasDelete](#) ([RootFindingCordasT](#) *obj)
Apaga a instancia do objeto [RootFindingCordas](#).
- [RootFindingBoolT RootFindingCordasPerformIteration](#) ([RootFindingCordasT](#) *cordasObj)
Realiza a iteracao.
- int [RootFindingCordasGetErrorCode](#) ([RootFindingCordasT](#) *cordasObj)
Obtem o codigo de erro.
- int [RootFindingCordasGetStateCode](#) ([RootFindingCordasT](#) *cordasObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingCordasGetErrorMessage](#) ([RootFindingCordasT](#) *cordasObj)

Obtem a mensagem de erro.

- `const char * RootFindingCordasGetStateMessage (RootFindingCordasT *cordasObj)`
Obtem a mensagem referente ao estado do objeto.
- `RootFindingBoolT RootFindingCordasHasError (RootFindingCordasT *cordasObj)`
Verifica se ha erros.
- `static void setError (RootFindingCordasT *cordasObj, int errorCode)`
Set error code and change state to CORDAS_ERROR_FOUND.
- `static void resetError (RootFindingCordasT *cordasObj)`
- `static RootFindingDoubleT getNextX (RootFindingCordasT *cordasObj)`
Obtem o valor para o proximo x.

5.4.1 Define Documentation

5.4.1.1 #define CORDAS_DEFAULT_MAX_ITERATIONS 100

Definition at line 38 of file RootFindingCordas.h.

Referenced by RootFindingCordasCreate().

5.4.1.2 #define CORDAS_DEFAULT_TOLERANCE 1e-7

Definition at line 39 of file RootFindingCordas.h.

Referenced by RootFindingCordasCreate().

5.4.2 Typedef Documentation

5.4.2.1 typedef struct RootFindingCordas RootFindingCordasT

Apelido para struct [RootFindingCordas](#).

Definition at line 84 of file RootFindingCordas.h.

5.4.3 Function Documentation

5.4.3.1 static RootFindingDoubleT getNextX (RootFindingCordasT * cordasObj) [static, private]

Obtem o valor para o proximo x.

Parameters:

cordasObj Ponteiro para objeto struct [RootFindingCordas](#)

Returns:

Valor do proximo x

Definition at line 118 of file RootFindingCordas.c.

References RootFindingCordas::c, RootFindingCordas::fC, RootFindingBase::fX, RootFindingCordas::rootsObj, and RootFindingBase::x.

Referenced by RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonPerformIteration(), and RootFindingPegasoPerformIteration().

5.4.3.2 static void resetError (RootFindingCordasT * *cordasObj*) [static, private]

Definition at line 186 of file RootFindingCordas.c.

References RootFindingCordas::errorCode, and RootFindingCordas::state.

5.4.3.3 RootFindingCordasT* RootFindingCordasCreate (RootFindingBaseT * *rootsObj*)

Cria um objeto do tipo struct [RootFindingCordas](#).

Parameters:

rootsObj Ponteiro para o objeto do tipo struct [RootFindingBase](#)

Returns:

Ponteiro para o objeto criado

Definition at line 48 of file RootFindingCordas.c.

References CORDAS_DEFAULT_MAX_ITERATIONS, CORDAS_DEFAULT_TOLERANCE, RootFindingCordas::maxIterations, resetError(), RootFindingCordas::rootsObj, RootFindingCordas::state, and RootFindingCordas::tolerance.

5.4.3.4 void RootFindingCordasDelete (RootFindingCordasT * *obj*)

Apaga a instancia do objeto [RootFindingCordas](#).

Parameters:

obj Ponteiro para o obj [RootFindingCordas](#)

Definition at line 127 of file RootFindingCordas.c.

5.4.3.5 int RootFindingCordasGetErrorCode (RootFindingCordasT * *cordasObj*)

Obtem o codigo de erro.

Definition at line 181 of file RootFindingCordas.c.

References RootFindingCordas::errorCode.

5.4.3.6 const char* RootFindingCordasGetErrorMessage (RootFindingCordasT * *cordasObj*)

Obtem a mensagem de erro.

Definition at line 202 of file RootFindingCordas.c.

References RootFindingBase::a, RootFindingBase::b, RootFindingCordas::errorCode, msg, MSG_CORDAS_2NDIFF_TEST_ERROR, MSG_CORDAS_NO_ERROR, MSG_CORDAS_X_ISINF_OR_ISNAN_ERROR, MSG_ROOTS_UNKNOW_ERROR, and RootFindingCordas::rootsObj.

5.4.3.7 int RootFindingCordasGetStateCode (RootFindingCordasT * cordasObj)

Obtem o codigo referente ao estado do objeto.

Definition at line 195 of file RootFindingCordas.c.

References RootFindingCordas::state.

5.4.3.8 const char* RootFindingCordasGetStateMessage (RootFindingCordasT * cordasObj)

Obtem a mensagem referente ao estado do objeto.

Definition at line 219 of file RootFindingCordas.c.

References RootFindingBase::e, RootFindingCordas::maxIterations, msg, MSG_CORDAS_APPROXIMATION_ROOT_FOUND, MSG_CORDAS_ERROR_FOUND, MSG_CORDAS_INITIALIZED, MSG_CORDAS_MAX_ITERATIONS_LIMIT_REACHED, MSG_CORDAS_NOT_INIT, MSG_ROOTS_UNKNOW_STATE, RootFindingCordas::rootsObj, RootFindingCordas::state, RootFindingCordas::tolerance, and RootFindingBase::x.

5.4.3.9 RootFindingBoolT RootFindingCordasHasError (RootFindingCordasT * cordasObj)

Verifica se ha erros.

Returns:

TRUE caso haja erro

Definition at line 244 of file RootFindingCordas.c.

References RootFindingCordas::errorCode.

5.4.3.10 RootFindingBoolT RootFindingCordasInit (RootFindingCordasT * cordasObj)

Inicializa o objeto [RootFindingCordas](#).

Parameters:

cordasObj Ponteiro para o objeto a ser inicializado

Returns:

TRUE em caso de sucesso

Definition at line 64 of file RootFindingCordas.c.

References RootFindingBase::a, RootFindingBase::b, RootFindingCordas::c, RootFindingBase::e, FALSE, RootFindingCordas::fC, RootFindingBase::fX, RootFindingCordas::i, infinity(), resetError(), RootFindingBase2nDiffEval(), RootFindingBaseEval(), RootFindingCordas::rootsObj, setError(), RootFindingCordas::state, TRUE, and RootFindingBase::x.

5.4.3.11 RootFindingBoolT RootFindingCordasPerformIteration (RootFindingCordasT * cordasObj)

Realiza a iteracao.

Parameters:

cordasObj Ponteiro para o objeto

Returns:

TRUE caso haja mais iteracoes a serem realizadas

Definition at line 138 of file RootFindingCordas.c.

References RootFindingBase::e, FALSE, RootFindingBase::fX, getNextX(), RootFindingCordas::i, isInfOrNan(), RootFindingCordas::maxIterations, RootFindingBaseEval(), RootFindingCordas::rootsObj, setError(), RootFindingCordas::state, RootFindingBase::state, RootFindingCordas::tolerance, TRUE, and RootFindingBase::x.

5.4.3.12 static void setError (RootFindingCordasT * cordasObj, int errorCode) [static, private]

Set error code and change state to CORDAS_ERROR_FOUND.

Parameters:

cordasObj Ponteiro para objeto RootFindingCordasT

errorCode Codigo de erro

Definition at line 132 of file RootFindingCordas.c.

References RootFindingCordas::errorCode, and RootFindingCordas::state.

5.5 Metodo de Newton-Rhapson

Data Structures

- struct [RootFindingNewtonRhapson](#)
Estrutura de dados para o Metodo de Newton-Rhapson.

Defines

- #define [NEWTON_DEFAULT_MAX_ITERATIONS](#) 100
- #define [NEWTON_DEFAULT_TOLERANCE](#) 1e-7

Typedefs

- typedef struct [RootFindingNewtonRhapson](#) RootFindingNewtonRhapsonT
Apelido para struct [RootFindingNewtonRhapson](#).

Functions

- [RootFindingNewtonRhapsonT](#) * [RootFindingNewtonRhapsonCreate](#) (RootFindingBaseT *rootsObj)
Cria um objeto do tipo struct [RootFindingNewtonRhapson](#).
- [RootFindingBoolT](#) [RootFindingNewtonRhapsonInit](#) (RootFindingNewtonRhapsonT *newtonObj)
Inicializa o objeto [RootFindingNewtonRhapson](#).

- void `RootFindingNewtonRhapsonDelete` (`RootFindingNewtonRhapsonT *newtonObj`)
Apaga o objeto `RootFindingNewtonRhapson`.
- `RootFindingBoolT` `RootFindingNewtonRhapsonPerformIteration` (`RootFindingNewtonRhapsonT *newtonObj`)
Realiza a iteracao.
- int `RootFindingNewtonRhapsonGetErrorCode` (`RootFindingNewtonRhapsonT *newtonObj`)
Obtem o codigo de erro.
- int `RootFindingNewtonRhapsonGetStateCode` (`RootFindingNewtonRhapsonT *newtonObj`)
Obtem o codigo referente ao estado do objeto.
- const char * `RootFindingNewtonRhapsonGetErrorMessage` (`RootFindingNewtonRhapsonT *newtonObj`)
Obtem a mensagem de erro.
- const char * `RootFindingNewtonRhapsonGetStateMessage` (`RootFindingNewtonRhapsonT *newtonObj`)
Obtem a mensagem referente ao estado do objeto.
- `RootFindingBoolT` `RootFindingNewtonRhapsonHasError` (`RootFindingNewtonRhapsonT *newtonObj`)
Verifica se ha erros.
- static void `setError` (`RootFindingNewtonRhapsonT *newtonObj`, int `errorCode`)
Set error code and change state to `NEWTON_ERROR_FOUND`.
- static void `resetError` (`RootFindingNewtonRhapsonT *newtonObj`)
- static `RootFindingDoubleT` `getNextX` (`RootFindingNewtonRhapsonT *newtonObj`)
Obtem o valor para o proximo x.

5.5.1 Define Documentation

5.5.1.1 #define NEWTON_DEFAULT_MAX_ITERATIONS 100

Definition at line 38 of file `RootFindingNewtonRhapson.h`.

Referenced by `RootFindingNewtonRhapsonCreate()`.

5.5.1.2 #define NEWTON_DEFAULT_TOLERANCE 1e-7

Definition at line 39 of file `RootFindingNewtonRhapson.h`.

Referenced by `RootFindingNewtonRhapsonCreate()`.

5.5.2 Typedef Documentation

5.5.2.1 typedef struct RootFindingNewtonRhapson RootFindingNewtonRhapsonT

Apelido para struct [RootFindingNewtonRhapson](#).

Definition at line 87 of file RootFindingNewtonRhapson.h.

5.5.3 Function Documentation

5.5.3.1 static RootFindingDoubleT getNextX (RootFindingNewtonRhapsonT * newtonObj) [static, private]

Obtem o valor para o proximo x.

Parameters:

newtonObj Ponteiro para objeto struct [RootFindingNewtonRhapson](#)

Returns:

Valor do proximo x

Definition at line 127 of file RootFindingNewtonRhapson.c.

References [RootFindingBase::fX](#), [RootFindingBaseDiffEval\(\)](#), [RootFindingNewtonRhapson::rootsObj](#), and [RootFindingBase::x](#).

5.5.3.2 static void resetError (RootFindingNewtonRhapsonT * newtonObj) [static, private]

Definition at line 195 of file RootFindingNewtonRhapson.c.

References [RootFindingNewtonRhapson::errorCode](#), and [RootFindingNewtonRhapson::state](#).

5.5.3.3 RootFindingNewtonRhapsonT* RootFindingNewtonRhapsonCreate (RootFindingBaseT * rootsObj)

Cria um objeto do tipo struct [RootFindingNewtonRhapson](#).

Parameters:

rootsObj Ponteiro para objeto do tipo struct [RootFindingBase](#)

Returns:

Ponteiro para o objeto criado

Definition at line 47 of file RootFindingNewtonRhapson.c.

References [RootFindingNewtonRhapson::maxIterations](#), [NEWTON_DEFAULT_MAX_ITERATIONS](#), [NEWTON_DEFAULT_TOLERANCE](#), [resetError\(\)](#), [RootFindingNewtonRhapson::rootsObj](#), [RootFindingNewtonRhapson::state](#), and [RootFindingNewtonRhapson::tolerance](#).

5.5.3.4 void RootFindingNewtonRhapsonDelete (RootFindingNewtonRhapsonT * newtonObj)

Apaga o objeto [RootFindingNewtonRhapson](#).

Parameters:

newtonObj Ponteiro para o objeto

Definition at line 115 of file RootFindingNewtonRhapson.c.

5.5.3.5 int RootFindingNewtonRhapsonGetErrorCode (RootFindingNewtonRhapsonT * *newtonObj*)

Obtem o codigo de erro.

Definition at line 190 of file RootFindingNewtonRhapson.c.

References RootFindingNewtonRhapson::errorCode.

5.5.3.6 const char* RootFindingNewtonRhapsonGetErrorMessage (RootFindingNewtonRhapsonT * *newtonObj*)

Obtem a mensagem de erro.

Definition at line 211 of file RootFindingNewtonRhapson.c.

References RootFindingBase::a, RootFindingBase::b, RootFindingBase::e, RootFindingNewtonRhapson::errorCode, msg, MSG_NEWTON_2NDIFF_TEST_ERROR, MSG_NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_EQUALS_ERROR, MSG_NEWTON_NO_ERROR, MSG_NEWTON_X_ISINF_OR_ISNAN_ERROR, MSG_ROOTS_UNKNOW_ERROR, RootFindingNewtonRhapson::rootsObj, and RootFindingBase::x.

5.5.3.7 int RootFindingNewtonRhapsonGetStateCode (RootFindingNewtonRhapsonT * *newtonObj*)

Obtem o codigo referente ao estado do objeto.

Definition at line 204 of file RootFindingNewtonRhapson.c.

References RootFindingNewtonRhapson::state.

5.5.3.8 const char* RootFindingNewtonRhapsonGetStateMessage (RootFindingNewtonRhapsonT * *newtonObj*)

Obtem a mensagem referente ao estado do objeto.

Definition at line 235 of file RootFindingNewtonRhapson.c.

References RootFindingBase::e, RootFindingNewtonRhapson::maxIterations, msg, MSG_NEWTON_APPROXIMATION_ROOT_FOUND, MSG_NEWTON_ERROR_FOUND, MSG_NEWTON_INITIALIZED, MSG_NEWTON_MAX_ITERATIONS_LIMIT_REACHED, MSG_NEWTON_NOT_INIT, MSG_ROOTS_UNKNOW_STATE, RootFindingNewtonRhapson::rootsObj, RootFindingNewtonRhapson::state, RootFindingNewtonRhapson::tolerance, and RootFindingBase::x.

5.5.3.9 RootFindingBoolT RootFindingNewtonRhapsonHasError (RootFindingNewtonRhapsonT * *newtonObj*)

Verifica se ha erros.

Returns:

TRUE caso haja erro

Definition at line 260 of file RootFindingNewtonRhapson.c.

References RootFindingNewtonRhapson::errorCode.

5.5.3.10 RootFindingBoolT RootFindingNewtonRhapsonInit (RootFindingNewtonRhapsonT * *newtonObj*)

Inicializa o objeto [RootFindingNewtonRhapson](#).

Parameters:

newtonObj Ponteiro para o objeto a ser inicializado

Returns:

TRUE em caso de sucesso, FALSE caso contrario

Definition at line 65 of file RootFindingNewtonRhapson.c.

References RootFindingBase::a, RootFindingBase::b, RootFindingBase::e, FALSE, RootFindingBase::fX, RootFindingNewtonRhapson::i, infinity(), RootFindingBase2nDiffEval(), RootFindingBaseEval(), RootFindingNewtonRhapson::rootsObj, setError(), RootFindingNewtonRhapson::state, TRUE, and RootFindingBase::x.

5.5.3.11 RootFindingBoolT RootFindingNewtonRhapsonPerformIteration (RootFindingNewtonRhapsonT * *newtonObj*)

Realiza a iteracao.

Parameters:

newtonObj Ponteiro para o objeto

Returns:

TRUE caso haja mais iteracoes a serem realizadas

Definition at line 146 of file RootFindingNewtonRhapson.c.

References RootFindingBase::e, FALSE, RootFindingBase::fX, getNextX(), RootFindingNewtonRhapson::i, isInfOrNan(), RootFindingNewtonRhapson::maxIterations, RootFindingBaseEval(), RootFindingNewtonRhapson::rootsObj, setError(), RootFindingNewtonRhapson::state, RootFindingBase::state, RootFindingNewtonRhapson::tolerance, TRUE, and RootFindingBase::x.

5.5.3.12 static void setError (RootFindingNewtonRhapsonT * *newtonObj*, int *errorCode*) [static, private]

Set error code and change state to NEWTON_ERROR_FOUND.

Parameters:

newtonObj Ponteiro para objeto [RootFindingNewtonRhapson](#)

errorCode Codigo de erro

Definition at line 140 of file RootFindingNewtonRhapson.c.

References RootFindingNewtonRhapson::errorCode, and RootFindingNewtonRhapson::state.

5.6 Metodo de Pegaso

Data Structures

- struct [RootFindingPegaso](#)
Estrutura de dados para o Metodo de Pegaso.

Defines

- #define [PEGASO_DEFAULT_MAX_ITERATIONS](#) 100
- #define [PEGASO_DEFAULT_TOLERANCE](#) 1e-7

Typedefs

- typedef struct [RootFindingPegaso](#) [RootFindingPegasoT](#)
Apelido para struct [RootFindingPegaso](#).

Functions

- [RootFindingPegasoT](#) * [RootFindingPegasoCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingPegaso](#).
- [RootFindingBoolT](#) [RootFindingPegasoInit](#) ([RootFindingPegasoT](#) *pegasoObj)
Inicializa o objeto [RootFindingPegaso](#).
- void [RootFindingPegasoDelete](#) ([RootFindingPegasoT](#) *obj)
Apaga a instancia do objeto [RootFindingPegaso](#).
- [RootFindingBoolT](#) [RootFindingPegasoPerformIteration](#) ([RootFindingPegasoT](#) *pegasoObj)
Realiza a iteracao.
- int [RootFindingPegasoGetErrorCode](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem o codigo de erro.
- int [RootFindingPegasoGetStateCode](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingPegasoGetErrorMessage](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem a mensagem de erro.
- const char * [RootFindingPegasoGetStateMessage](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingPegasoHasError](#) ([RootFindingPegasoT](#) *pegasoObj)
Verifica se ha erros.
- static void [setError](#) ([RootFindingPegasoT](#) *pegasoObj, int errorCode)

Set error code and change state to PEGASO_ERROR_FOUND.

- static void [resetError](#) ([RootFindingPegasoT](#) *pegasoObj)
- static [RootFindingDoubleT](#) [getNextX](#) ([RootFindingPegasoT](#) *pegasoObj)

Obtem o valor para o proximo x.

5.6.1 Define Documentation

5.6.1.1 #define PEGASO_DEFAULT_MAX_ITERATIONS 100

Definition at line 38 of file [RootFindingPegaso.h](#).

Referenced by [RootFindingPegasoCreate\(\)](#).

5.6.1.2 #define PEGASO_DEFAULT_TOLERANCE 1e-7

Definition at line 39 of file [RootFindingPegaso.h](#).

Referenced by [RootFindingPegasoCreate\(\)](#).

5.6.2 Typedef Documentation

5.6.2.1 typedef struct RootFindingPegaso RootFindingPegasoT

Apelido para struct [RootFindingPegaso](#).

Definition at line 84 of file [RootFindingPegaso.h](#).

5.6.3 Function Documentation

5.6.3.1 static [RootFindingDoubleT](#) [getNextX](#) ([RootFindingPegasoT](#) * *pegasoObj*) [static, private]

Obtem o valor para o proximo x.

Parameters:

pegasoObj Ponteiro para objeto struct [RootFindingPegaso](#)

Returns:

Valor do proximo x

Definition at line 93 of file [RootFindingPegaso.c](#).

References [RootFindingPegaso::fPrevXi](#), [RootFindingBase::fX](#), [RootFindingPegaso::prevXi](#), [RootFindingPegaso::rootsObj](#), and [RootFindingBase::x](#).

5.6.3.2 static void [resetError](#) ([RootFindingPegasoT](#) * *pegasoObj*) [static, private]

Definition at line 159 of file [RootFindingPegaso.c](#).

References [RootFindingPegaso::errorCode](#), [RootFindingPegaso::PEGASO_ERROR_FOUND](#), [RootFindingPegaso::PEGASO_NO_ERROR](#), [RootFindingPegaso::PEGASO_NOT_INIT](#), and [RootFindingPegaso::state](#).

5.6.3.3 RootFindingPegasoT* RootFindingPegasoCreate (RootFindingBaseT * rootsObj)

Cria um objeto do tipo struct [RootFindingPegaso](#).

Parameters:

rootsObj Ponteiro para o objeto do tipo struct [RootFindingBase](#)

Returns:

Ponteiro para o objeto criado

Definition at line 47 of file RootFindingPegaso.c.

References [RootFindingPegaso::maxIterations](#), [PEGASO_DEFAULT_MAX_ITERATIONS](#), [PEGASO_DEFAULT_TOLERANCE](#), [RootFindingPegaso::PEGASO_NOT_INIT](#), [resetError\(\)](#), [RootFindingPegaso::rootsObj](#), [RootFindingPegaso::state](#), and [RootFindingPegaso::tolerance](#).

5.6.3.4 void RootFindingPegasoDelete (RootFindingPegasoT * obj)

Apaga a instancia do objeto [RootFindingPegaso](#).

Parameters:

obj Ponteiro para o obj [RootFindingPegaso](#)

Definition at line 81 of file RootFindingPegaso.c.

5.6.3.5 int RootFindingPegasoGetErrorCode (RootFindingPegasoT * pegasoObj)

Obtem o codigo de erro.

Definition at line 154 of file RootFindingPegaso.c.

References [RootFindingPegaso::errorCode](#).

5.6.3.6 const char* RootFindingPegasoGetErrorMessage (RootFindingPegasoT * pegasoObj)

Obtem a mensagem de erro.

Definition at line 175 of file RootFindingPegaso.c.

References [RootFindingPegaso::errorCode](#), [MSG_PEGASO_NO_ERROR](#), [MSG_PEGASO_X_ISINF_OR_ISNAN_ERROR](#), [MSG_ROOTS_UNKNOW_ERROR](#), [RootFindingPegaso::PEGASO_NO_ERROR](#), and [RootFindingPegaso::PEGASO_X_ISINF_OR_ISNAN_ERROR](#).

5.6.3.7 int RootFindingPegasoGetStateCode (RootFindingPegasoT * pegasoObj)

Obtem o codigo referente ao estado do objeto.

Definition at line 168 of file RootFindingPegaso.c.

References [RootFindingPegaso::state](#).

5.6.3.8 const char* RootFindingPegasoGetStateMessage (RootFindingPegasoT * pegasoObj)

Obtem a mensagem referente ao estado do objeto.

Definition at line 188 of file RootFindingPegaso.c.

References `RootFindingBase::e`, `RootFindingPegaso::maxIterations`, `msg`, `MSG_PEGASO_APPROXIMATION_ROOT_FOUND`, `MSG_PEGASO_ERROR_FOUND`, `MSG_PEGASO_INITIALIZED`, `MSG_PEGASO_MAX_ITERATIONS_LIMIT_REACHED`, `MSG_PEGASO_NOT_INIT`, `MSG_ROOTS_UNKNOW_STATE`, `RootFindingPegaso::PEGASO_APPROXIMATION_ROOT_FOUND`, `RootFindingPegaso::PEGASO_ERROR_FOUND`, `RootFindingPegaso::PEGASO_INITIALIZED`, `RootFindingPegaso::PEGASO_MAX_ITERATIONS_LIMIT_REACHED`, `RootFindingPegaso::PEGASO_NOT_INIT`, `RootFindingPegaso::rootsObj`, `RootFindingPegaso::state`, `RootFindingPegaso::tolerance`, and `RootFindingBase::x`.

5.6.3.9 `RootFindingBoolT RootFindingPegasoHasError (RootFindingPegasoT * pegasoObj)`

Verifica se ha erros.

Returns:

TRUE caso haja erro

Definition at line 213 of file `RootFindingPegaso.c`.

References `RootFindingPegaso::errorCode`, and `RootFindingPegaso::PEGASO_NO_ERROR`.

5.6.3.10 `RootFindingBoolT RootFindingPegasoInit (RootFindingPegasoT * pegasoObj)`

Inicializa o objeto [RootFindingPegaso](#).

Parameters:

pegasoObj Ponteiro para o objeto a ser inicializado

Returns:

TRUE em caso de sucesso

Definition at line 63 of file `RootFindingPegaso.c`.

References `RootFindingBase::a`, `RootFindingBase::b`, `RootFindingBase::e`, `RootFindingPegaso::fPrevXi`, `RootFindingBase::fX`, `RootFindingPegaso::i`, `RootFindingPegaso::PEGASO_INITIALIZED`, `RootFindingPegaso::prevXi`, `resetError()`, `RootFindingBaseEval()`, `RootFindingPegaso::rootsObj`, `RootFindingPegaso::state`, `TRUE`, and `RootFindingBase::x`.

5.6.3.11 `RootFindingBoolT RootFindingPegasoPerformIteration (RootFindingPegasoT * pegasoObj)`

Realiza a iteracao.

Parameters:

pegasoObj Ponteiro para o objeto

Returns:

TRUE caso haja mais iteracoes a serem realizadas

Definition at line 108 of file `RootFindingPegaso.c`.

References `RootFindingBase::e`, `FALSE`, `RootFindingPegaso::fPrevXi`, `RootFindingBase::fX`, `getNextX()`, `RootFindingPegaso::i`, `isInfOrNan()`, `RootFindingPegaso::maxIterations`, `RootFindingPegaso::PEGASO_APPROXIMATION_ROOT_FOUND`, `RootFindingPegaso::PEGASO_MAX_ITERATIONS_LIMIT_REACHED`, `RootFindingPegaso::PEGASO_X_ISINF_OR_ISNAN_ERROR`, `RootFindingPegaso::prevXi`, `RootFindingBaseEval()`, `RootFindingPegaso::rootsObj`, `setError()`, `RootFindingPegaso::state`, `RootFindingBase::state`, `RootFindingPegaso::tolerance`, `TRUE`, and `RootFindingBase::x`.

5.6.3.12 `static void setError (RootFindingPegasoT * pegasoObj, int errorCode)` [`static`, `private`]

Set error code and change state to `PEGASO_ERROR_FOUND`.

Parameters:

pegasoObj Ponteiro para objeto `RootFindingPegasoT`
errorCode Código de erro

Definition at line 102 of file `RootFindingPegaso.c`.

References `RootFindingPegaso::errorCode`, `RootFindingPegaso::PEGASO_ERROR_FOUND`, and `RootFindingPegaso::state`.

6 Data Structure Documentation

6.1 RootFindingBase Struct Reference

Estrutura de dados para [RootFindingBase](#).

```
#include <RootFindingBase.h>
```

Public Types

- enum {
[ROOTS_NO_ERROR](#) = 0, [ROOTS_MUP_EVAL_ERROR](#), [RANGE_ERRORS_START](#), [ROOTS_RANGE_ERROR_PROD_FA_FB_NOT_LT_0](#),
[ROOTS_RANGE_ERROR_FA_OR_FB_ISINFINITY](#), [RANGE_ERRORS_END](#) }
Código de erro: não acessar diretamente. Veja [RootFindingBaseGetErrorCode](#) e [RootFindingBaseGetErrorMessage](#).
- enum { [ROOTS_RANGE_NOT_SET](#) = 0, [ROOTS_READY](#), [ROOTS_EXACT_ROOT_FOUND](#) }
Código de estado: não acessar diretamente. Veja [RootFindingBaseGetStateCode](#) e [RootFindingBaseGetStateMessage](#).

Data Fields

- [RootFindingDoubleT](#) a
- [RootFindingDoubleT](#) b
- [RootFindingDoubleT](#) x
- [RootFindingDoubleT](#) fX

- [RootFindingDoubleT](#) e
- `muParserHandle_t` [mupObj](#)
- [RootFindingDoubleT](#) * `varPtr`
- `enum RootFindingBase:: { ... } errorCode`

Codigo de erro: nao acessar diretamente. Veja [RootFindingBaseGetErrorCode](#) e [RootFindingBaseGetErrorMessage](#).

- `enum RootFindingBase:: { ... } state`

Codigo de estado: nao acessar diretamente. Veja [RootFindingBaseGetStateCode](#) e [RootFindingBaseGetStateMessage](#).

6.1.1 Detailed Description

Estrutura de dados para [RootFindingBase](#).

Definition at line 39 of file [RootFindingBase.h](#).

6.1.2 Member Enumeration Documentation

6.1.2.1 anonymous enum

Codigo de erro: nao acessar diretamente. Veja [RootFindingBaseGetErrorCode](#) e [RootFindingBaseGetErrorMessage](#).

Enumerator:

ROOTS_NO_ERROR
ROOTS_MUP_EVAL_ERROR
RANGE_ERRORS_START
ROOTS_RANGE_ERROR_PROD_FA_FB_NOT_LT_0
ROOTS_RANGE_ERROR_FA_OR_FB_ISINFINITY
RANGE_ERRORS_END

Definition at line 53 of file [RootFindingBase.h](#).

6.1.2.2 anonymous enum

Codigo de estado: nao acessar diretamente. Veja [RootFindingBaseGetStateCode](#) e [RootFindingBaseGetStateMessage](#).

Enumerator:

ROOTS_RANGE_NOT_SET
ROOTS_READY
ROOTS_EXACT_ROOT_FOUND

Definition at line 72 of file [RootFindingBase.h](#).

6.1.3 Field Documentation

6.1.3.1 RootFindingDoubleT RootFindingBase::a

Definition at line 41 of file RootFindingBase.h.

Referenced by computeX(), RootFindingBaseGetErrorMessage(), RootFindingBaseSetRange(), RootFindingBissecasFindNewRange(), RootFindingCordasGetErrorMessage(), RootFindingCordasInit(), RootFindingNewtonRhapsonGetErrorMessage(), RootFindingNewtonRhapsonInit(), and RootFindingPegasoInit().

6.1.3.2 RootFindingDoubleT RootFindingBase::b

Definition at line 42 of file RootFindingBase.h.

Referenced by computeX(), RootFindingBaseGetErrorMessage(), RootFindingBaseSetRange(), RootFindingBissecasFindNewRange(), RootFindingCordasGetErrorMessage(), RootFindingCordasInit(), RootFindingNewtonRhapsonGetErrorMessage(), RootFindingNewtonRhapsonInit(), and RootFindingPegasoInit().

6.1.3.3 RootFindingDoubleT RootFindingBase::x

Definition at line 43 of file RootFindingBase.h.

Referenced by getNextX(), RootFindingBaseGetStateMessage(), RootFindingBissecasFindNewRange(), RootFindingBissecasGetStateMessage(), RootFindingBissecasInit(), RootFindingBissecasPerformIteration(), RootFindingCordasGetStateMessage(), RootFindingCordasInit(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonGetErrorMessage(), RootFindingNewtonRhapsonGetStateMessage(), RootFindingNewtonRhapsonInit(), RootFindingNewtonRhapsonPerformIteration(), RootFindingPegasoGetStateMessage(), RootFindingPegasoInit(), and RootFindingPegasoPerformIteration().

6.1.3.4 RootFindingDoubleT RootFindingBase::fx

Definition at line 44 of file RootFindingBase.h.

Referenced by getNextX(), RootFindingBissecasFindNewRange(), RootFindingBissecasInit(), RootFindingBissecasPerformIteration(), RootFindingCordasInit(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonInit(), RootFindingNewtonRhapsonPerformIteration(), RootFindingPegasoInit(), and RootFindingPegasoPerformIteration().

6.1.3.5 RootFindingDoubleT RootFindingBase::e

Definition at line 45 of file RootFindingBase.h.

Referenced by RootFindingBaseReset(), RootFindingBissecasGetStateMessage(), RootFindingBissecasInit(), RootFindingBissecasPerformIteration(), RootFindingCordasGetStateMessage(), RootFindingCordasInit(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonGetErrorMessage(), RootFindingNewtonRhapsonGetStateMessage(), RootFindingNewtonRhapsonInit(), RootFindingNewtonRhapsonPerformIteration(), RootFindingPegasoGetStateMessage(), RootFindingPegasoInit(), and RootFindingPegasoPerformIteration().

6.1.3.6 muParserHandle_t RootFindingBase::mupObj

Definition at line 46 of file RootFindingBase.h.

Referenced by RootFindingBase2nDiffEval(), RootFindingBaseCreate(), RootFindingBaseDiffEval(), RootFindingBaseEval(), and RootFindingBaseReset().

6.1.3.7 RootFindingDoubleT* RootFindingBase::varPtr

Definition at line 47 of file RootFindingBase.h.

Referenced by RootFindingBase2nDiffEval(), RootFindingBaseCreate(), RootFindingBaseDiffEval(), and RootFindingBaseEval().

6.1.3.8 enum { ... } RootFindingBase::errorCode

Codigo de erro: nao acessar diretamente. Veja [RootFindingBaseGetErrorCode](#) e [RootFindingBaseGetErrorMessage](#).

Referenced by isRangeError(), RootFindingBaseGetErrorCode(), RootFindingBaseGetErrorMessage(), RootFindingBaseHasError(), RootFindingBaseReset(), and RootFindingBaseSetRange().

6.1.3.9 enum { ... } RootFindingBase::state

Codigo de estado: nao acessar diretamente. Veja [RootFindingBaseGetStateCode](#) e [RootFindingBaseGetStateMessage](#).

Referenced by RootFindingBaseGetStateCode(), RootFindingBaseGetStateMessage(), RootFindingBaseReset(), RootFindingBaseSetRange(), RootFindingBissecaoPerformIteration(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonPerformIteration(), and RootFindingPegasoPerformIteration().

The documentation for this struct was generated from the following file:

- include/[RootFindingBase.h](#)

6.2 RootFindingBissecao Struct Reference

Estrutura de dados para o Metodo da Bissecao.

```
#include <RootFindingBissecao.h>
```

Public Types

- enum {
[BISSECAO_NOT_INIT](#) = 0, [BISSECAO_INITIALIZED](#), [BISSECAO_MAX_ITERATIONS_LIMIT_REACHED](#), [BISSECAO_APPROXIMATION_ROOT_FOUND](#),
[BISSECAO_ERROR_FOUND](#) }
Codigo de estado: nao acessar diretamente. Veja [RootFindingBissecaoGetStateCode](#) e [RootFindingBissecaoGetStateMessage](#).
- enum { [BISSECAO_NO_ERROR](#) = 0, [BISSECAO_X_ISINF_OR_ISNAN_ERROR](#) }
Codigo de erro: nao acessar diretamente. Veja [RootFindingBissecaoGetErrorCode](#) e [RootFindingBissecaoGetErrorMessage](#).

Data Fields

- unsigned i
- [RootFindingBaseT](#) * rootsObj

- unsigned [maxIterations](#)
- [RootFindingDoubleT](#) tolerance
- enum RootFindingBissecacao:: { ... } state

Codigo de estado: nao acessar diretamente. Veja [RootFindingBissecacaoGetStateCode](#) e [RootFindingBissecacaoGetStateMessage](#).

- enum RootFindingBissecacao:: { ... } errorCode

Codigo de erro: nao acessar diretamente. Veja [RootFindingBissecacaoGetErrorCode](#) e [RootFindingBissecacaoGetErrorMessage](#).

6.2.1 Detailed Description

Estrutura de dados para o Metodo da Bissecacao.

Definition at line 42 of file RootFindingBissecacao.h.

6.2.2 Member Enumeration Documentation

6.2.2.1 anonymous enum

Codigo de estado: nao acessar diretamente. Veja [RootFindingBissecacaoGetStateCode](#) e [RootFindingBissecacaoGetStateMessage](#).

Enumerator:

BISSECAO_NOT_INIT
BISSECAO_INITIALIZED
BISSECAO_MAX_ITERATIONS_LIMIT_REACHED
BISSECAO_APPROXIMATION_ROOT_FOUND
BISSECAO_ERROR_FOUND

Definition at line 53 of file RootFindingBissecacao.h.

6.2.2.2 anonymous enum

Codigo de erro: nao acessar diretamente. Veja [RootFindingBissecacaoGetErrorCode](#) e [RootFindingBissecacaoGetErrorMessage](#).

Enumerator:

BISSECAO_NO_ERROR
BISSECAO_X_ISINF_OR_ISNAN_ERROR

Definition at line 67 of file RootFindingBissecacao.h.

6.2.3 Field Documentation

6.2.3.1 unsigned RootFindingBissecacao::i

Definition at line 44 of file RootFindingBissecacao.h.

Referenced by [RootFindingBissecacaoInit\(\)](#), and [RootFindingBissecacaoPerformIteration\(\)](#).

6.2.3.2 RootFindingBaseT* RootFindingBissecacao::rootsObj

Definition at line 45 of file RootFindingBissecacao.h.

Referenced by computeX(), RootFindingBissecacaoCreate(), RootFindingBissecacaoGetStateMessage(), RootFindingBissecacaoInit(), and RootFindingBissecacaoPerformIteration().

6.2.3.3 unsigned RootFindingBissecacao::maxIterations

Definition at line 46 of file RootFindingBissecacao.h.

Referenced by RootFindingBissecacaoCreate(), RootFindingBissecacaoGetStateMessage(), and RootFindingBissecacaoPerformIteration().

6.2.3.4 RootFindingDoubleT RootFindingBissecacao::tolerance

Definition at line 47 of file RootFindingBissecacao.h.

Referenced by RootFindingBissecacaoCreate(), RootFindingBissecacaoGetStateMessage(), and RootFindingBissecacaoPerformIteration().

6.2.3.5 enum { ... } RootFindingBissecacao::state

Codigo de estado: nao acessar diretamente. Veja [RootFindingBissecacaoGetStateCode](#) e [RootFindingBissecacaoGetStateMessage](#).

Referenced by resetError(), RootFindingBissecacaoCreate(), RootFindingBissecacaoGetStateCode(), RootFindingBissecacaoGetStateMessage(), RootFindingBissecacaoInit(), RootFindingBissecacaoPerformIteration(), and setError().

6.2.3.6 enum { ... } RootFindingBissecacao::errorCode

Codigo de erro: nao acessar diretamente. Veja [RootFindingBissecacaoGetErrorCode](#) e [RootFindingBissecacaoGetErrorMessage](#).

Referenced by resetError(), RootFindingBissecacaoGetErrorCode(), RootFindingBissecacaoGetErrorMessage(), RootFindingBissecacaoHasError(), and setError().

The documentation for this struct was generated from the following file:

- [include/RootFindingBissecacao.h](#)

6.3 RootFindingCordas Struct Reference

Estrutura de dados para o Metodo das Cordas.

```
#include <RootFindingCordas.h>
```

Public Types

- enum {
[CORDAS_NOT_INIT](#) = 0, [CORDAS_INITIALIZED](#), [CORDAS_MAX_ITERATIONS_LIMIT_REACHED](#), [CORDAS_APPROXIMATION_ROOT_FOUND](#),
[CORDAS_ERROR_FOUND](#) }

Codigo de estado: nao acessar diretamente. Veja [RootFindingCordasGetStateCode](#) e [RootFindingCordasGetStateMessage](#).

- enum { [CORDAS_NO_ERROR](#) = 0, [CORDAS_2NDIFF_TEST_ERROR](#), [CORDAS_X_ISINF_OR_ISNAN_ERROR](#) }

Codigo de erro: nao acessar diretamente. Veja [RootFindingCordasGetErrorCode](#) e [RootFindingCordasGetErrorMessage](#).

Data Fields

- unsigned i
- [RootFindingBaseT](#) * rootsObj
- [RootFindingDoubleT](#) c
- [RootFindingDoubleT](#) fC
- unsigned maxIterations
- [RootFindingDoubleT](#) tolerance
- enum [RootFindingCordas](#):: { ... } state

Codigo de estado: nao acessar diretamente. Veja [RootFindingCordasGetStateCode](#) e [RootFindingCordasGetStateMessage](#).

- enum [RootFindingCordas](#):: { ... } errorCode

Codigo de erro: nao acessar diretamente. Veja [RootFindingCordasGetErrorCode](#) e [RootFindingCordasGetErrorMessage](#).

6.3.1 Detailed Description

Estrutura de dados para o Metodo das Cordas.

Definition at line 44 of file [RootFindingCordas.h](#).

6.3.2 Member Enumeration Documentation

6.3.2.1 anonymous enum

Codigo de estado: nao acessar diretamente. Veja [RootFindingCordasGetStateCode](#) e [RootFindingCordasGetStateMessage](#).

Enumerator:

CORDAS_NOT_INIT
CORDAS_INITIALIZED
CORDAS_MAX_ITERATIONS_LIMIT_REACHED
CORDAS_APPROXIMATION_ROOT_FOUND
CORDAS_ERROR_FOUND

Definition at line 59 of file [RootFindingCordas.h](#).

6.3.2.2 anonymous enum

Codigo de erro: nao acessar diretamente. Veja [RootFindingCordasGetErrorCode](#) e [RootFindingCordasGetErrorMessage](#).

Enumerator:

CORDAS_NO_ERROR
CORDAS_2NDIFF_TEST_ERROR
CORDAS_X_ISINF_OR_ISNAN_ERROR

Definition at line 73 of file RootFindingCordas.h.

6.3.3 Field Documentation

6.3.3.1 unsigned RootFindingCordas::i

Definition at line 46 of file RootFindingCordas.h.

Referenced by RootFindingCordasInit(), and RootFindingCordasPerformIteration().

6.3.3.2 RootFindingBaseT* RootFindingCordas::rootsObj

Definition at line 47 of file RootFindingCordas.h.

Referenced by getNextX(), RootFindingCordasCreate(), RootFindingCordasGetErrorMessage(), RootFindingCordasGetStateMessage(), RootFindingCordasInit(), and RootFindingCordasPerformIteration().

6.3.3.3 RootFindingDoubleT RootFindingCordas::c

Definition at line 50 of file RootFindingCordas.h.

Referenced by getNextX(), and RootFindingCordasInit().

6.3.3.4 RootFindingDoubleT RootFindingCordas::fC

Definition at line 51 of file RootFindingCordas.h.

Referenced by getNextX(), and RootFindingCordasInit().

6.3.3.5 unsigned RootFindingCordas::maxIterations

Definition at line 52 of file RootFindingCordas.h.

Referenced by RootFindingCordasCreate(), RootFindingCordasGetStateMessage(), and RootFindingCordasPerformIteration().

6.3.3.6 RootFindingDoubleT RootFindingCordas::tolerance

Definition at line 53 of file RootFindingCordas.h.

Referenced by RootFindingCordasCreate(), RootFindingCordasGetStateMessage(), and RootFindingCordasPerformIteration().

6.3.3.7 enum { ... } RootFindingCordas::state

Codigo de estado: nao acessar diretamente. Veja [RootFindingCordasGetStateCode](#) e [RootFindingCordasGetStateMessage](#).

Referenced by [resetError\(\)](#), [RootFindingCordasCreate\(\)](#), [RootFindingCordasGetStateCode\(\)](#), [RootFindingCordasGetStateMessage\(\)](#), [RootFindingCordasInit\(\)](#), [RootFindingCordasPerformIteration\(\)](#), and [setError\(\)](#).

6.3.3.8 enum { ... } RootFindingCordas::errorCode

Codigo de erro: nao acessar diretamente. Veja [RootFindingCordasGetErrorCode](#) e [RootFindingCordasGetErrorMessage](#).

Referenced by [resetError\(\)](#), [RootFindingCordasGetErrorCode\(\)](#), [RootFindingCordasGetErrorMessage\(\)](#), [RootFindingCordasHasError\(\)](#), and [setError\(\)](#).

The documentation for this struct was generated from the following file:

- [include/RootFindingCordas.h](#)

6.4 RootFindingNewtonRhapson Struct Reference

Estrutura de dados para o Metodo de Newton-Rhapson.

```
#include <RootFindingNewtonRhapson.h>
```

Public Types

- enum {
[NEWTON_NOT_INIT](#) = 0, [NEWTON_INITIALIZED](#), [NEWTON_MAX_ITERATIONS_LIMIT_REACHED](#), [NEWTON_APPROXIMATION_ROOT_FOUND](#),
[NEWTON_ERROR_FOUND](#) }

Codigo de estado: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetStateCode](#) e [RootFindingNewtonRhapsonGetStateMessage](#).

- enum { [NEWTON_NO_ERROR](#) = 0, [NEWTON_2NDIFF_TEST_ERROR](#), [NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_EQUALS_ERROR](#), [NEWTON_X_ISINF_OR_ISNAN_ERROR](#) }

Codigo de erro: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetErrorCode](#) e [RootFindingNewtonRhapsonGetErrorMessage](#).

Data Fields

- unsigned [i](#)
- [RootFindingBaseT](#) * [rootsObj](#)
- unsigned [maxIterations](#)
- [RootFindingDoubleT](#) [tolerance](#)
- enum [RootFindingNewtonRhapson::](#) { ... } [state](#)

Codigo de estado: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetStateCode](#) e [RootFindingNewtonRhapsonGetStateMessage](#).

- enum RootFindingNewtonRhapson:: { ... } [errorCode](#)

Codigo de erro: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetErrorCode](#) e [RootFindingNewtonRhapsonGetErrorMessage](#).

6.4.1 Detailed Description

Estrutura de dados para o Metodo de Newton-Rhapson.

Definition at line 44 of file RootFindingNewtonRhapson.h.

6.4.2 Member Enumeration Documentation

6.4.2.1 anonymous enum

Codigo de estado: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetStateCode](#) e [RootFindingNewtonRhapsonGetStateMessage](#).

Enumerator:

NEWTON_NOT_INIT
NEWTON_INITIALIZED
NEWTON_MAX_ITERATIONS_LIMIT_REACHED
NEWTON_APPROXIMATION_ROOT_FOUND
NEWTON_ERROR_FOUND

Definition at line 58 of file RootFindingNewtonRhapson.h.

6.4.2.2 anonymous enum

Codigo de erro: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetErrorCode](#) e [RootFindingNewtonRhapsonGetErrorMessage](#).

Enumerator:

NEWTON_NO_ERROR
NEWTON_2NDIFF_TEST_ERROR
NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_EQUALS_ERROR
NEWTON_X_ISINF_OR_ISNAN_ERROR

Definition at line 74 of file RootFindingNewtonRhapson.h.

6.4.3 Field Documentation

6.4.3.1 unsigned RootFindingNewtonRhapson::i

Definition at line 46 of file RootFindingNewtonRhapson.h.

Referenced by [RootFindingNewtonRhapsonInit\(\)](#), and [RootFindingNewtonRhapsonPerformIteration\(\)](#).

6.4.3.2 RootFindingBaseT* RootFindingNewtonRhapson::rootsObj

Definition at line 47 of file RootFindingNewtonRhapson.h.

Referenced by getNextX(), RootFindingNewtonRhapsonCreate(), RootFindingNewtonRhapsonGetErrorMessage(), RootFindingNewtonRhapsonGetStateMessage(), RootFindingNewtonRhapsonInit(), and RootFindingNewtonRhapsonPerformIteration().

6.4.3.3 unsigned RootFindingNewtonRhapson::maxIterations

Definition at line 50 of file RootFindingNewtonRhapson.h.

Referenced by RootFindingNewtonRhapsonCreate(), RootFindingNewtonRhapsonGetStateMessage(), and RootFindingNewtonRhapsonPerformIteration().

6.4.3.4 RootFindingDoubleT RootFindingNewtonRhapson::tolerance

Definition at line 51 of file RootFindingNewtonRhapson.h.

Referenced by RootFindingNewtonRhapsonCreate(), RootFindingNewtonRhapsonGetStateMessage(), and RootFindingNewtonRhapsonPerformIteration().

6.4.3.5 enum { ... } RootFindingNewtonRhapson::state

Codigo de estado: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetStateCode](#) e [RootFindingNewtonRhapsonGetStateMessage](#).

Referenced by resetError(), RootFindingNewtonRhapsonCreate(), RootFindingNewtonRhapsonGetStateCode(), RootFindingNewtonRhapsonGetStateMessage(), RootFindingNewtonRhapsonInit(), RootFindingNewtonRhapsonPerformIteration(), and setError().

6.4.3.6 enum { ... } RootFindingNewtonRhapson::errorCode

Codigo de erro: nao acessar diretamente. Veja [RootFindingNewtonRhapsonGetErrorCode](#) e [RootFindingNewtonRhapsonGetErrorMessage](#).

Referenced by resetError(), RootFindingNewtonRhapsonGetErrorCode(), RootFindingNewtonRhapsonGetErrorMessage(), RootFindingNewtonRhapsonHasError(), and setError().

The documentation for this struct was generated from the following file:

- [include/RootFindingNewtonRhapson.h](#)

6.5 RootFindingPegaso Struct Reference

Estrutura de dados para o Metodo de Pegaso.

```
#include <RootFindingPegaso.h>
```

Public Types

- enum {
[PEGASO_NOT_INIT](#) = 0, [PEGASO_INITIALIZED](#), [PEGASO_MAX_ITERATIONS_LIMIT_REACHED](#), [PEGASO_APPROXIMATION_ROOT_FOUND](#),
[PEGASO_ERROR_FOUND](#) }

Codigo de estado: nao acessar diretamente. Veja [RootFindingPegasoGetStateCode](#) e [RootFindingPegaso-GetStateMessage](#).

- enum { [PEGASO_NO_ERROR](#) = 0, [PEGASO_X_ISINF_OR_ISNAN_ERROR](#) }

Codigo de erro: nao acessar diretamente. Veja [RootFindingPegasoGetErrorCode](#) e [RootFindingPegaso-GetErrorMessage](#).

Data Fields

- unsigned i
- [RootFindingBaseT](#) * rootsObj
- [RootFindingDoubleT](#) prevXi
- [RootFindingDoubleT](#) fPrevXi
- [RootFindingDoubleT](#) tolerance
- unsigned [maxIterations](#)
- enum [RootFindingPegaso::](#) { ... } [state](#)

Codigo de estado: nao acessar diretamente. Veja [RootFindingPegasoGetStateCode](#) e [RootFindingPegaso-GetStateMessage](#).

- enum [RootFindingPegaso::](#) { ... } [errorCode](#)

Codigo de erro: nao acessar diretamente. Veja [RootFindingPegasoGetErrorCode](#) e [RootFindingPegaso-GetErrorMessage](#).

6.5.1 Detailed Description

Estrutura de dados para o Metodo de Pegaso.

Definition at line 44 of file [RootFindingPegaso.h](#).

6.5.2 Member Enumeration Documentation

6.5.2.1 anonymous enum

Codigo de estado: nao acessar diretamente. Veja [RootFindingPegasoGetStateCode](#) e [RootFindingPegaso-GetStateMessage](#).

Enumerator:

PEGASO_NOT_INIT
PEGASO_INITIALIZED
PEGASO_MAX_ITERATIONS_LIMIT_REACHED
PEGASO_APPROXIMATION_ROOT_FOUND
PEGASO_ERROR_FOUND

Definition at line 59 of file [RootFindingPegaso.h](#).

6.5.2.2 anonymous enum

Codigo de erro: nao acessar diretamente. Veja [RootFindingPegasoGetErrorCode](#) e [RootFindingPegasoGetErrorMessage](#).

Enumerator:

PEGASO_NO_ERROR

PEGASO_X_ISINF_OR_ISNAN_ERROR

Definition at line 73 of file RootFindingPegaso.h.

6.5.3 Field Documentation

6.5.3.1 unsigned RootFindingPegaso::i

Definition at line 46 of file RootFindingPegaso.h.

Referenced by [getAxisVarPtr\(\)](#), [RootFindingPegasoInit\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

6.5.3.2 RootFindingBaseT* RootFindingPegaso::rootsObj

Definition at line 47 of file RootFindingPegaso.h.

Referenced by [getNextX\(\)](#), [RootFindingBaseCreate\(\)](#), [RootFindingPegasoCreate\(\)](#), [RootFindingPegasoGetStateMessage\(\)](#), [RootFindingPegasoInit\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

6.5.3.3 RootFindingDoubleT RootFindingPegaso::prevXi

Definition at line 48 of file RootFindingPegaso.h.

Referenced by [getNextX\(\)](#), [RootFindingPegasoInit\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

6.5.3.4 RootFindingDoubleT RootFindingPegaso::fPrevXi

Definition at line 49 of file RootFindingPegaso.h.

Referenced by [getNextX\(\)](#), [RootFindingPegasoInit\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

6.5.3.5 RootFindingDoubleT RootFindingPegaso::tolerance

Definition at line 52 of file RootFindingPegaso.h.

Referenced by [RootFindingPegasoCreate\(\)](#), [RootFindingPegasoGetStateMessage\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

6.5.3.6 unsigned RootFindingPegaso::maxIterations

Definition at line 53 of file RootFindingPegaso.h.

Referenced by [RootFindingPegasoCreate\(\)](#), [RootFindingPegasoGetStateMessage\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

6.5.3.7 enum { ... } RootFindingPegaso::state

Codigo de estado: nao acessar diretamente. Veja [RootFindingPegasoGetStateCode](#) e [RootFindingPegasoGetStateMessage](#).

Referenced by `resetError()`, `RootFindingPegasoCreate()`, `RootFindingPegasoGetStateCode()`, `RootFindingPegasoGetStateMessage()`, `RootFindingPegasoInit()`, `RootFindingPegasoPerformIteration()`, and `setError()`.

6.5.3.8 enum { ... } RootFindingPegaso::errorCode

Codigo de erro: nao acessar diretamente. Veja [RootFindingPegasoGetErrorCode](#) e [RootFindingPegasoGetErrorMessage](#).

Referenced by `resetError()`, `RootFindingPegasoGetErrorCode()`, `RootFindingPegasoGetErrorMessage()`, `RootFindingPegasoHasError()`, and `setError()`.

The documentation for this struct was generated from the following file:

- [include/RootFindingPegaso.h](#)

7 File Documentation

7.1 include/messages/RootFindingMessages.h File Reference

```
/root-finding/include/messages/RootFindingMessages.h  
#include "RootFindingMessages_PT-BR.h"
```

7.1.1 Detailed Description

```
/root-finding/include/messages/RootFindingMessages.h
```

Date:

02/04/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingMessages.h](#).

7.2 include/messages/RootFindingMessages_PT-BR.h File Reference

```
/root-finding/include/messages/RootFindingMessages_PT-BR.h
```

Defines

- #define [MSG_ROOTS_UNKNOWN_ERROR](#) "Erro desconhecido."
- #define [MSG_ROOTS_UNKNOWN_STATE](#) "Estado desconhecido."
- #define [MSG_GENERIC_APPROXIMATION_ROOT_FOUND](#)
- #define [MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED](#)

- #define MSG_GENERIC_NO_ERROR "Nao ocorreu erro."
- #define MSG_GENERIC_ERROR_FOUND "Ocorreu erro.";
- #define MSG_GENERIC_NOT_INIT "Nao inicializado."
- #define MSG_GENERIC_INITIALIZED "Inicializado."
- #define MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR
- #define MSG_GENERIC_2NDIFF_TEST_ERROR
- #define MSG_ROOTS_NO_ERROR MSG_GENERIC_NO_ERROR
- #define MSG_ROOTS_MUP_EVAL_ERROR "Erro c/ expressao do objeto muParser"
- #define MSG_ROOTS_RANGE_ERROR_PROD_FA_FB_NOT_LT_0
- #define MSG_ROOTS_RANGE_ERROR_FA_OR_FB_ISINFINITY
- #define MSG_ROOTS_RANGE_NOT_SET "Intervalo nao esta definido."
- #define MSG_ROOTS_READY "Objeto RootFindingBase esta pronto."
- #define MSG_ROOTS_EXACT_ROOT_FOUND "Encontrado a raiz exata: %lf."
- #define MSG_BISSECAO_NO_ERROR MSG_GENERIC_NO_ERROR
- #define MSG_BISSECAO_X_ISINF_OR_ISNAN_ERROR MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR
- #define MSG_BISSECAO_NOT_INIT MSG_GENERIC_NOT_INIT
- #define MSG_BISSECAO_INITIALIZED MSG_GENERIC_INITIALIZED
- #define MSG_BISSECAO_MAX_ITERATIONS_LIMIT_REACHED MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED
- #define MSG_BISSECAO_APPROXIMATION_ROOT_FOUND MSG_GENERIC_APPROXIMATION_ROOT_FOUND
- #define MSG_BISSECAO_ERROR_FOUND MSG_GENERIC_ERROR_FOUND
- #define MSG_CORDAS_NO_ERROR MSG_GENERIC_NO_ERROR
- #define MSG_CORDAS_2NDIFF_TEST_ERROR MSG_GENERIC_2NDIFF_TEST_ERROR
- #define MSG_CORDAS_X_ISINF_OR_ISNAN_ERROR MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR
- #define MSG_CORDAS_NOT_INIT MSG_GENERIC_NOT_INIT
- #define MSG_CORDAS_INITIALIZED MSG_GENERIC_INITIALIZED
- #define MSG_CORDAS_MAX_ITERATIONS_LIMIT_REACHED MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED
- #define MSG_CORDAS_APPROXIMATION_ROOT_FOUND MSG_GENERIC_APPROXIMATION_ROOT_FOUND
- #define MSG_CORDAS_ERROR_FOUND MSG_GENERIC_ERROR_FOUND
- #define MSG_NEWTON_NO_ERROR MSG_GENERIC_NO_ERROR
- #define MSG_NEWTON_2NDIFF_TEST_ERROR MSG_GENERIC_2NDIFF_TEST_ERROR
- #define MSG_NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_EQUALS_ERROR
- #define MSG_NEWTON_X_ISINF_OR_ISNAN_ERROR
- #define MSG_NEWTON_NOT_INIT MSG_GENERIC_NOT_INIT
- #define MSG_NEWTON_INITIALIZED MSG_GENERIC_INITIALIZED
- #define MSG_NEWTON_MAX_ITERATIONS_LIMIT_REACHED MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED
- #define MSG_NEWTON_APPROXIMATION_ROOT_FOUND MSG_GENERIC_APPROXIMATION_ROOT_FOUND
- #define MSG_NEWTON_ERROR_FOUND MSG_GENERIC_ERROR_FOUND
- #define MSG_PEGASO_NO_ERROR MSG_GENERIC_NO_ERROR
- #define MSG_PEGASO_X_ISINF_OR_ISNAN_ERROR MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR
- #define MSG_PEGASO_NOT_INIT MSG_GENERIC_NOT_INIT
- #define MSG_PEGASO_INITIALIZED MSG_GENERIC_INITIALIZED

- #define [MSG_PEGASO_MAX_ITERATIONS_LIMIT_REACHED](#) MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED
- #define [MSG_PEGASO_APPROXIMATION_ROOT_FOUND](#) MSG_GENERIC_APPROXIMATION_ROOT_FOUND
- #define [MSG_PEGASO_ERROR_FOUND](#) MSG_GENERIC_ERROR_FOUND

7.2.1 Detailed Description

/root-finding/include/messages/RootFindingMessages_PT-BR.h

Date:

02/04/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingMessages_PT-BR.h](#).

7.3 include/RootFinding.h File Reference

/root-finding/include/RootFinding.h

7.3.1 Detailed Description

/root-finding/include/RootFinding.h

Date:

03/04/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFinding.h](#).

7.4 include/RootFindingBase.h File Reference

/root-finding/include/RootFindingBase.h

```
#include "RootFindingCommon.h"
```

Data Structures

- struct [RootFindingBase](#)
Estrutura de dados para [RootFindingBase](#).

Typedefs

- typedef struct [RootFindingBase](#) [RootFindingBaseT](#)

Apelido para struct [RootFindingBase](#).

Functions

- [RootFindingBaseT](#) * [RootFindingBaseCreate](#) ([muParserHandle_t](#) mupObj, [RootFindingDoubleT](#) *varPtr)

Cria o objeto [RootFindingBase](#).

- void [RootFindingBaseDelete](#) ([RootFindingBaseT](#) *obj)

Apaga o objeto [RootFindingBase](#).

- [RootFindingBoolT](#) [RootFindingBaseSetRange](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) a, [RootFindingDoubleT](#) b)

Define o intervalo para procura da raiz.

- [RootFindingDoubleT](#) [RootFindingBaseEval](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) value)

Avalia a funcao em um ponto.

- [RootFindingDoubleT](#) [RootFindingBase2nDiffEval](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) value)

Avalia a 2a. diferencial da funcao em um ponto.

- [RootFindingDoubleT](#) [RootFindingBaseDiffEval](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) value)

Avalia a diferencial da funcao em um ponto.

- void [RootFindingBaseReset](#) ([RootFindingBaseT](#) *rootsObj)

Reinicializa o objeto struct [RootFindingBase](#).

- int [RootFindingBaseGetErrorCode](#) ([RootFindingBaseT](#) *rootsObj)

Obtem o codigo de erro.

- int [RootFindingBaseGetStateCode](#) ([RootFindingBaseT](#) *rootsObj)

Obtem o codigo referente ao estado do objeto.

- const char * [RootFindingBaseGetErrorMessage](#) ([RootFindingBaseT](#) *rootsObj)

Obtem a mensagem de erro.

- const char * [RootFindingBaseGetStateMessage](#) ([RootFindingBaseT](#) *rootsObj)

Obtem a mensagem referente ao estado do objeto.

- [RootFindingBoolT](#) [RootFindingBaseHasError](#) ([RootFindingBaseT](#) *rootsObj)

Verifica se ha erros.

7.4.1 Detailed Description

/root-finding/include/RootFindingBase.h

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingBase.h](#).

7.5 include/RootFindingBissecacao.h File Reference

/root-finding/include/RootFindingBissecacao.h

```
#include "RootFindingBase.h"
```

Data Structures

- struct [RootFindingBissecacao](#)
Estrutura de dados para o Metodo da Bissecacao.

Defines

- #define [BISSECAO_DEFAULT_MAX_ITERATIONS](#) 100
- #define [BISSECAO_DEFAULT_TOLERANCE](#) 1e-7

Typedefs

- typedef struct [RootFindingBissecacao](#) [RootFindingBissecacaoT](#)
Apelido para struct [RootFindingBissecacao](#).

Functions

- [RootFindingBissecacaoT * RootFindingBissecacaoCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingBissecacao](#).
- [RootFindingBoolT RootFindingBissecacaoInit](#) ([RootFindingBissecacaoT](#) *bissecacaoObj)
Inicializa o objeto [RootFindingBissecacao](#).
- void [RootFindingBissecacaoDelete](#) ([RootFindingBissecacaoT](#) *obj)
Apaga a instancia do objeto [RootFindingBissecacao](#).
- [RootFindingBoolT RootFindingBissecacaoPerformIteration](#) ([RootFindingBissecacaoT](#) *bissecacaoObj)
Realiza a iteracao.

- void [RootFindingBissecaoFindNewRange](#) ([RootFindingBaseT](#) *rootsObj)
Encontra um novo intervalo $[A, B]$ e os altera no objeto [RootFindingBaseT](#) baseado nos $[A, B]$ e x existentes. Utilizado em [RootFindingBissecaoPerformIteration](#) porem principalmente util para alterar o intervalo quando intercambiando entre metodos diferentes.
- int [RootFindingBissecaoGetErrorCode](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem o codigo de erro.
- int [RootFindingBissecaoGetStateCode](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingBissecaoGetErrorMessage](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem a mensagem de erro.
- const char * [RootFindingBissecaoGetStateMessage](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingBissecaoHasError](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Verifica se ha erros.

7.5.1 Detailed Description

/root-finding/include/RootFindingBissecao.h

Date:

26/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingBissecao.h](#).

7.6 include/RootFindingCommon.h File Reference

/root-finding/include/RootFindingCommon.h

```
#include <muParserDLL.h>
```

Defines

- #define [TRUE](#) 1
- #define [FALSE](#) 0

Typedefs

- typedef double [RootFindingDoubleT](#)
- typedef int [RootFindingBoolT](#)

Functions

- [RootFindingBoolT](#) [getAxisVarPtr](#) (muParserHandle_t mupObj, const char *axis, [RootFindingDoubleT](#) **varPtr)
Obtem o endereco para a variavel do eixo onde deve-se.
- [RootFindingDoubleT](#) [Mup2ndDiff](#) (muParserHandle_t mupObj, [RootFindingDoubleT](#) *varPtr)
Aproximacao da segunda diferencial no ponto utilizando "Finite difference".
- [RootFindingDoubleT](#) [MupDiff](#) (muParserHandle_t mupObj, [RootFindingDoubleT](#) *varPtr)
Aproximacao da diferencial no ponto utilizando "regard to a variable" O codigo foi adaptado para C c/ base no metodo Diff da classe muParser (C++), infelizmente a API para C do muParser nao disbonibiliza o metodo Diff.
- [RootFindingDoubleT](#) [infinity](#) ()
Obtem o valor infinito.
- [RootFindingBoolT](#) [isInfOrNan](#) ([RootFindingDoubleT](#) num)
Verifica se um dado numero [RootFindingDoubleT](#) eh infinito ou "Not a Number".

7.6.1 Detailed Description

/root-finding/include/RootFindingCommon.h

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingCommon.h](#).

7.6.2 Define Documentation

7.6.2.1 #define FALSE 0

Definition at line 39 of file RootFindingCommon.h.

Referenced by [getAxisVarPtr\(\)](#), [RootFindingBaseSetRange\(\)](#), [RootFindingBissecasPerformIteration\(\)](#), [RootFindingCordasInit\(\)](#), [RootFindingCordasPerformIteration\(\)](#), [RootFindingNewtonRhapsonInit\(\)](#), [RootFindingNewtonRhapsonPerformIteration\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

7.6.2.2 #define TRUE 1

Definition at line 36 of file RootFindingCommon.h.

Referenced by [getAxisVarPtr\(\)](#), [RootFindingBaseSetRange\(\)](#), [RootFindingBissecasInit\(\)](#), [RootFindingBissecasPerformIteration\(\)](#), [RootFindingCordasInit\(\)](#), [RootFindingCordasPerformIteration\(\)](#), [RootFindingNewtonRhapsonInit\(\)](#), [RootFindingNewtonRhapsonPerformIteration\(\)](#), [RootFindingPegasoInit\(\)](#), and [RootFindingPegasoPerformIteration\(\)](#).

7.6.3 Typedef Documentation

7.6.3.1 typedef int RootFindingBoolT

Definition at line 33 of file RootFindingCommon.h.

7.6.3.2 typedef double RootFindingDoubleT

Definition at line 31 of file RootFindingCommon.h.

7.6.4 Function Documentation

7.6.4.1 RootFindingBoolT getAxisVarPtr (muParserHandle_t *mupObj*, const char * *axis*, RootFindingDoubleT ** *varPtr*)

Obtem o endereco para a variavel do eixo onde deve-se.

Parameters:

mupObj Ponteiro para objeto muParser contendo a expressao

axis Eixo em qual a raiz deve ser procurada

→ *varPtr* Ponteiro para onde o endereco deve ser atribuido

Returns:

TRUE em caso de sucesso FALSE em caso de falha

Definition at line 30 of file RootFindingCommon.c.

References FALSE, RootFindingPegaso::i, and TRUE.

7.6.4.2 RootFindingDoubleT infinity ()

Obtem o valor infinito.

Returns:

inf

Definition at line 55 of file RootFindingCommon.c.

Referenced by RootFindingBaseReset(), RootFindingBissecaoInit(), RootFindingCordasInit(), and RootFindingNewtonRhapsonInit().

7.6.4.3 RootFindingBoolT isInfOrNan (RootFindingDoubleT *num*)

Verifica se um dado numero [RootFindingDoubleT](#) eh infinito ou "Not a Number".

Parameters:

num Numero a verificado

Returns:

TRUE se isinf(num) ou isnan(num)

Definition at line 97 of file RootFindingCommon.c.

Referenced by RootFindingBissecacaoPerformIteration(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonPerformIteration(), and RootFindingPegasoPerformIteration().

7.6.4.4 RootFindingDoubleT Mup2ndDiff (muParserHandle_t *mupObj*, RootFindingDoubleT * *varPtr*)

Aproximacao da segunda diferencial no ponto utilizando "Finite difference".

Parameters:

mupObj Objeto muParser (ponteiro)

varPtr Ponteiro para a variavel relacionada a expressao no objeto muParser referente a qual variavel deve ser diferenciado

Returns:

Valor da segunda diferencial no ponto

Definition at line 61 of file RootFindingCommon.c.

Referenced by RootFindingBase2nDiffEval().

7.6.4.5 RootFindingDoubleT MupDiff (muParserHandle_t *mupObj*, RootFindingDoubleT * *varPtr*)

Aproximacao da diferencial no ponto utilizando "regard to a variable" O codigo foi adaptado para C c/ base no metodo Diff da classe muParser (C++), infelizmente a API para C do muParser nao disponibiliza o metodo Diff.

Parameters:

mupObj Objeto muParser (ponteiro)

varPtr Ponteiro para a variavel relacionada a expressao no objeto muParser referente a qual variavel deve ser diferenciado

Returns:

Valor da diferencial no ponto

Definition at line 77 of file RootFindingCommon.c.

Referenced by RootFindingBaseDiffEval().

7.7 include/RootFindingCordas.h File Reference

/root-finding/include/RootFindingCordas.h

```
#include <muParserDLL.h>
```

```
#include "RootFindingCommon.h"
```

```
#include "RootFindingBase.h"
```

Data Structures

- struct [RootFindingCordas](#)

Estrutura de dados para o Metodo das Cordas.

Defines

- #define [CORDAS_DEFAULT_MAX_ITERATIONS](#) 100
- #define [CORDAS_DEFAULT_TOLERANCE](#) 1e-7

Typedefs

- typedef struct [RootFindingCordas](#) [RootFindingCordasT](#)

Apelido para struct [RootFindingCordas](#).

Functions

- [RootFindingCordasT](#) * [RootFindingCordasCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingCordas](#).
- [RootFindingBoolT](#) [RootFindingCordasInit](#) ([RootFindingCordasT](#) *cordasObj)
Inicializa o objeto [RootFindingCordas](#).
- void [RootFindingCordasDelete](#) ([RootFindingCordasT](#) *obj)
Apaga a instancia do objeto [RootFindingCordas](#).
- [RootFindingBoolT](#) [RootFindingCordasPerformIteration](#) ([RootFindingCordasT](#) *cordasObj)
Realiza a iteracao.
- int [RootFindingCordasGetErrorCode](#) ([RootFindingCordasT](#) *cordasObj)
Obtem o codigo de erro.
- int [RootFindingCordasGetStateCode](#) ([RootFindingCordasT](#) *cordasObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingCordasGetErrorMessage](#) ([RootFindingCordasT](#) *cordasObj)
Obtem a mensagem de erro.
- const char * [RootFindingCordasGetStateMessage](#) ([RootFindingCordasT](#) *cordasObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingCordasHasError](#) ([RootFindingCordasT](#) *cordasObj)
Verifica se ha erros.

7.7.1 Detailed Description

/root-finding/include/RootFindingCordas.h

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingCordas.h](#).

7.8 include/RootFindingNewtonRhapson.h File Reference

/root-finding/include/RootFindingNewtonRhapson.h

```
#include <muParserDLL.h>
#include "RootFindingCommon.h"
#include "RootFindingBase.h"
```

Data Structures

- struct [RootFindingNewtonRhapson](#)
Estrutura de dados para o Metodo de Newton-Rhapson.

Defines

- #define [NEWTON_DEFAULT_MAX_ITERATIONS](#) 100
- #define [NEWTON_DEFAULT_TOLERANCE](#) 1e-7

Typedefs

- typedef struct [RootFindingNewtonRhapson](#) [RootFindingNewtonRhapsonT](#)
Apelido para struct [RootFindingNewtonRhapson](#).

Functions

- [RootFindingNewtonRhapsonT](#) * [RootFindingNewtonRhapsonCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingNewtonRhapson](#).
- [RootFindingBoolT](#) [RootFindingNewtonRhapsonInit](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Inicializa o objeto [RootFindingNewtonRhapson](#).
- void [RootFindingNewtonRhapsonDelete](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Apaga o objeto [RootFindingNewtonRhapson](#).

- [RootFindingBoolT](#) [RootFindingNewtonRhapsonPerformIteration](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Realiza a iteracao.
- int [RootFindingNewtonRhapsonGetErrorCode](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem o codigo de erro.
- int [RootFindingNewtonRhapsonGetStateCode](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingNewtonRhapsonGetErrorMessage](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem a mensagem de erro.
- const char * [RootFindingNewtonRhapsonGetStateMessage](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingNewtonRhapsonHasError](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Verifica se ha erros.

7.8.1 Detailed Description

/root-finding/include/RootFindingNewtonRhapson.h

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingNewtonRhapson.h](#).

7.9 include/RootFindingPegaso.h File Reference

/root-finding/include/RootFindingPegaso.h

```
#include <muParserDLL.h>
#include "RootFindingCommon.h"
#include "RootFindingBase.h"
```

Data Structures

- struct [RootFindingPegaso](#)
Estrutura de dados para o Metodo de Pegaso.

Defines

- #define PEGASO_DEFAULT_MAX_ITERATIONS 100
- #define PEGASO_DEFAULT_TOLERANCE 1e-7

Typedefs

- typedef struct RootFindingPegaso RootFindingPegasoT
Apelido para struct RootFindingPegaso.

Functions

- RootFindingPegasoT * RootFindingPegasoCreate (RootFindingBaseT *rootsObj)
Cria um objeto do tipo struct RootFindingPegaso.
- RootFindingBoolT RootFindingPegasoInit (RootFindingPegasoT *pegasoObj)
Inicializa o objeto RootFindingPegaso.
- void RootFindingPegasoDelete (RootFindingPegasoT *obj)
Apaga a instancia do objeto RootFindingPegaso.
- RootFindingBoolT RootFindingPegasoPerformIteration (RootFindingPegasoT *pegasoObj)
Realiza a iteracao.
- int RootFindingPegasoGetErrorCode (RootFindingPegasoT *pegasoObj)
Obtem o codigo de erro.
- int RootFindingPegasoGetStateCode (RootFindingPegasoT *pegasoObj)
Obtem o codigo referente ao estado do objeto.
- const char * RootFindingPegasoGetErrorMessage (RootFindingPegasoT *pegasoObj)
Obtem a mensagem de erro.
- const char * RootFindingPegasoGetStateMessage (RootFindingPegasoT *pegasoObj)
Obtem a mensagem referente ao estado do objeto.
- RootFindingBoolT RootFindingPegasoHasError (RootFindingPegasoT *pegasoObj)
Verifica se ha erros.

7.9.1 Detailed Description

/root-finding/include/RootFindingPegaso.h

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingPegaso.h](#).

7.10 src/RootFindingBase.c File Reference

/root-finding/src/RootFindingBase.c

```
#include "RootFindingBase.h"
#include "messages/RootFindingMessages.h"
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
```

Functions

- [RootFindingBaseT](#) * [RootFindingBaseCreate](#) (muParserHandle_t mupObj, [RootFindingDoubleT](#) *varPtr)
Cria o objeto [RootFindingBase](#).
- void [RootFindingBaseReset](#) ([RootFindingBaseT](#) *rootsObj)
Reinicializa o objeto struct [RootFindingBase](#).
- void [RootFindingBaseDelete](#) ([RootFindingBaseT](#) *obj)
Apaga o objeto [RootFindingBase](#).
- static [RootFindingBoolT](#) [isRangeError](#) ([RootFindingBaseT](#) *rootsObj)
Verify if errorCode is a RangeError.
- [RootFindingBoolT](#) [RootFindingBaseSetRange](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) a, [RootFindingDoubleT](#) b)
Define o intervalo para procura da raiz.
- [RootFindingDoubleT](#) [RootFindingBaseEval](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) value)
Avalia a funcao em um ponto.
- [RootFindingDoubleT](#) [RootFindingBase2nDiffEval](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) value)
Avalia a 2a. diferencial da funcao em um ponto.
- [RootFindingDoubleT](#) [RootFindingBaseDiffEval](#) ([RootFindingBaseT](#) *rootsObj, [RootFindingDoubleT](#) value)
Avalia a diferencial da funcao em um ponto.
- int [RootFindingBaseGetErrorCode](#) ([RootFindingBaseT](#) *rootsObj)
Obtem o codigo de erro.

- int [RootFindingBaseGetStateCode](#) ([RootFindingBaseT](#) *rootsObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingBaseGetErrorMessage](#) ([RootFindingBaseT](#) *rootsObj)
Obtem a mensagem de erro.
- const char * [RootFindingBaseGetStateMessage](#) ([RootFindingBaseT](#) *rootsObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingBaseHasError](#) ([RootFindingBaseT](#) *rootsObj)
Verifica se ha erros.

Variables

- static char [msg](#) [255]

7.10.1 Detailed Description

/root-finding/src/RootFindingBase.c

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingBase.c](#).

7.10.2 Variable Documentation

7.10.2.1 char msg[255] [static]

Definition at line 141 of file RootFindingBase.c.

Referenced by [RootFindingBaseGetErrorMessage\(\)](#), [RootFindingBaseGetStateMessage\(\)](#), [RootFindingBissecacaoGetStateMessage\(\)](#), [RootFindingCordasGetErrorMessage\(\)](#), [RootFindingCordasGetStateMessage\(\)](#), [RootFindingNewtonRhapsonGetErrorMessage\(\)](#), [RootFindingNewtonRhapsonGetStateMessage\(\)](#), and [RootFindingPegasoGetStateMessage\(\)](#).

7.11 src/RootFindingBissecacao.c File Reference

/root-finding/src/RootFindingBissecacao.c

```
#include "RootFindingBissecacao.h"
#include "messages/RootFindingMessages.h"
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
```

Functions

- static void [setError](#) ([RootFindingBissecaoT](#) *bissecaoObj, int errorCode)
Set error code and change state to `BISSECAO_ERROR_FOUND`.
- static void [resetError](#) ([RootFindingBissecaoT](#) *bissecaoObj)
- [RootFindingBissecaoT](#) * [RootFindingBissecaoCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingBissecao](#).
- static [RootFindingDoubleT](#) [computeX](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Calcula o X baseado no intervalo $[a, b]$.
- [RootFindingBoolT](#) [RootFindingBissecaoInit](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Inicializa o objeto [RootFindingBissecao](#).
- void [RootFindingBissecaoDelete](#) ([RootFindingBissecaoT](#) *obj)
Apaga a instancia do objeto [RootFindingBissecao](#).
- [RootFindingBoolT](#) [RootFindingBissecaoPerformIteration](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Realiza a iteracao.
- void [RootFindingBissecaoFindNewRange](#) ([RootFindingBaseT](#) *rootsObj)
Encontra um novo intervalo $[A, B]$ e os altera no objeto [RootFindingBaseT](#) baseado nos $[A, B]$ e x existentes. Utilizado em [RootFindingBissecaoPerformIteration](#) porem principalmente util para alterar o intervalo quando intercambiando entre metodos diferentes.
- int [RootFindingBissecaoGetErrorCode](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem o codigo de erro.
- int [RootFindingBissecaoGetStateCode](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingBissecaoGetErrorMessage](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem a mensagem de erro.
- const char * [RootFindingBissecaoGetStateMessage](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingBissecaoHasError](#) ([RootFindingBissecaoT](#) *bissecaoObj)
Verifica se ha erros.

Variables

- static char [msg](#) [255]

7.11.1 Detailed Description

/root-finding/src/RootFindingBissecacao.c

Date:

26/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingBissecacao.c](#).

7.11.2 Variable Documentation

7.11.2.1 char msg[255] [static]

Definition at line 178 of file RootFindingBissecacao.c.

7.12 src/RootFindingCommon.c File Reference

/root-finding/src/RootFindingCommon.c

```
#include "RootFindingCommon.h"
#include <string.h>
#include <math.h>
```

Functions

- [RootFindingBoolT getAxisVarPtr](#) (muParserHandle_t mupObj, const char *axis, [RootFindingDoubleT **varPtr](#))
Obtem o endereco para a variavel do eixo onde deve-se.
- [RootFindingDoubleT infinity](#) ()
Obtem o valor infinito.
- [RootFindingDoubleT Mup2ndDiff](#) (muParserHandle_t mupObj, [RootFindingDoubleT *varPtr](#))
Aproximacao da segunda diferencial no ponto utilizando "Finite difference".
- [RootFindingDoubleT MupDiff](#) (muParserHandle_t mupObj, [RootFindingDoubleT *varPtr](#))
Aproximacao da diferencial no ponto utilizando "regard to a variable" O codigo foi adaptado para C c/ base no metodo Diff da classe muParser (C++), infelizmente a API para C do muParser nao disponibiliza o metodo Diff.
- [RootFindingBoolT isInfOrNan](#) ([RootFindingDoubleT](#) num)
Verifica se um dado numero [RootFindingDoubleT](#) eh infinito ou "Not a Number".

7.12.1 Detailed Description

/root-finding/src/RootFindingCommon.c

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingCommon.c](#).

7.12.2 Function Documentation

7.12.2.1 RootFindingBoolT getAxisVarPtr (muParserHandle_t *mupObj*, const char * *axis*, RootFindingDoubleT ** *varPtr*)

Obtem o endereco para a variavel do eixo onde deve-se.

Parameters:

mupObj Ponteiro para objeto muParser contendo a expressao

axis Eixo em qual a raiz deve ser procurada

→ *varPtr* Ponteiro para onde o endereco deve ser atribuido

Returns:

TRUE em caso de sucesso FALSE em caso de falha

Definition at line 30 of file RootFindingCommon.c.

References FALSE, RootFindingPegaso::i, and TRUE.

7.12.2.2 RootFindingDoubleT infinity ()

Obtem o valor infinito.

Returns:

inf

Definition at line 55 of file RootFindingCommon.c.

Referenced by RootFindingBaseReset(), RootFindingBissecacaoInit(), RootFindingCordasInit(), and RootFindingNewtonRhapsonInit().

7.12.2.3 RootFindingBoolT isInfOrNan (RootFindingDoubleT *num*)

Verifica se um dado numero [RootFindingDoubleT](#) eh infinito ou "Not a Number".

Parameters:

num Numero a verificado

Returns:

TRUE se isinf(num) ou isnan(num)

Definition at line 97 of file RootFindingCommon.c.

Referenced by RootFindingBissecacaoPerformIteration(), RootFindingCordasPerformIteration(), RootFindingNewtonRhapsonPerformIteration(), and RootFindingPegasoPerformIteration().

7.12.2.4 RootFindingDoubleT Mup2ndDiff (muParserHandle_t mupObj, RootFindingDoubleT * varPtr)

Aproximacao da segunda diferencial no ponto utilizando "Finite difference".

Parameters:

mupObj Objeto muParser (ponteiro)

varPtr Ponteiro para a variavel relacionada a expressao no objeto muParser referente a qual variavel deve ser diferenciado

Returns:

Valor da segunda diferencial no ponto

Definition at line 61 of file RootFindingCommon.c.

Referenced by RootFindingBase2nDiffEval().

7.12.2.5 RootFindingDoubleT MupDiff (muParserHandle_t mupObj, RootFindingDoubleT * varPtr)

Aproximacao da diferencial no ponto utilizando "regard to a variable" O codigo foi adaptado para C c/ base no metodo Diff da classe muParser (C++), infelizmente a API para C do muParser nao disponibiliza o metodo Diff.

Parameters:

mupObj Objeto muParser (ponteiro)

varPtr Ponteiro para a variavel relacionada a expressao no objeto muParser referente a qual variavel deve ser diferenciado

Returns:

Valor da diferencial no ponto

Definition at line 77 of file RootFindingCommon.c.

Referenced by RootFindingBaseDiffEval().

7.13 src/RootFindingCordas.c File Reference

/root-finding/src/RootFindingCordas.c

```
#include "RootFindingCordas.h"
```

```
#include "messages/RootFindingMessages.h"
```

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
```

Functions

- static void [setError](#) ([RootFindingCordasT](#) *cordasObj, int errorCode)
Set error code and change state to CORDAS_ERROR_FOUND.
- static void [resetError](#) ([RootFindingCordasT](#) *cordasObj)
- [RootFindingCordasT](#) * [RootFindingCordasCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingCordas](#).
- [RootFindingBoolT](#) [RootFindingCordasInit](#) ([RootFindingCordasT](#) *cordasObj)
Inicializa o objeto [RootFindingCordas](#).
- static [RootFindingDoubleT](#) [getNextX](#) ([RootFindingCordasT](#) *cordasObj)
Obtem o valor para o proximo x.
- void [RootFindingCordasDelete](#) ([RootFindingCordasT](#) *obj)
Apaga a instancia do objeto [RootFindingCordas](#).
- [RootFindingBoolT](#) [RootFindingCordasPerformIteration](#) ([RootFindingCordasT](#) *cordasObj)
Realiza a iteracao.
- int [RootFindingCordasGetErrorCode](#) ([RootFindingCordasT](#) *cordasObj)
Obtem o codigo de erro.
- int [RootFindingCordasGetStateCode](#) ([RootFindingCordasT](#) *cordasObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingCordasGetErrorMessage](#) ([RootFindingCordasT](#) *cordasObj)
Obtem a mensagem de erro.
- const char * [RootFindingCordasGetStateMessage](#) ([RootFindingCordasT](#) *cordasObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingCordasHasError](#) ([RootFindingCordasT](#) *cordasObj)
Verifica se ha erros.

Variables

- static char [msg](#) [255]

7.13.1 Detailed Description

/root-finding/src/RootFindingCordas.c

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingCordas.c](#).

7.13.2 Variable Documentation

7.13.2.1 char msg[255] [static]

Definition at line 200 of file RootFindingCordas.c.

7.14 src/RootFindingNewtonRhapson.c File Reference

/root-finding/src/RootFindingNewtonRhapson.c

```
#include "RootFindingNewtonRhapson.h"
#include "messages/RootFindingMessages.h"
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
```

Functions

- static void [setError](#) ([RootFindingNewtonRhapsonT](#) *newtonObj, int errorCode)
Set error code and change state to `NEWTON_ERROR_FOUND`.
- static void [resetError](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
- [RootFindingNewtonRhapsonT](#) * [RootFindingNewtonRhapsonCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingNewtonRhapson](#).
- [RootFindingBoolT](#) [RootFindingNewtonRhapsonInit](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Inicializa o objeto [RootFindingNewtonRhapson](#).
- void [RootFindingNewtonRhapsonDelete](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Apaga o objeto [RootFindingNewtonRhapson](#).
- static [RootFindingDoubleT](#) [getNextX](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem o valor para o proximo x.

- [RootFindingBoolT](#) [RootFindingNewtonRhapsonPerformIteration](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Realiza a iteracao.
- int [RootFindingNewtonRhapsonGetErrorCode](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem o codigo de erro.
- int [RootFindingNewtonRhapsonGetStateCode](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingNewtonRhapsonGetErrorMessage](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem a mensagem de erro.
- const char * [RootFindingNewtonRhapsonGetStateMessage](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingNewtonRhapsonHasError](#) ([RootFindingNewtonRhapsonT](#) *newtonObj)
Verifica se ha erros.

Variables

- static char [msg](#) [255]

7.14.1 Detailed Description

/root-finding/src/RootFindingNewtonRhapson.c

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingNewtonRhapson.c](#).

7.14.2 Variable Documentation

7.14.2.1 char msg[255] [static]

Definition at line 209 of file RootFindingNewtonRhapson.c.

7.15 src/RootFindingPegaso.c File Reference

/root-finding/src/RootFindingPegaso.c

```
#include "RootFindingPegaso.h"
#include "messages/RootFindingMessages.h"
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
```

Functions

- static void [setError](#) ([RootFindingPegasoT](#) *pegasoObj, int errorCode)
Set error code and change state to PEGASO_ERROR_FOUND.
- static void [resetError](#) ([RootFindingPegasoT](#) *pegasoObj)
- [RootFindingPegasoT](#) * [RootFindingPegasoCreate](#) ([RootFindingBaseT](#) *rootsObj)
Cria um objeto do tipo struct [RootFindingPegaso](#).
- [RootFindingBoolT](#) [RootFindingPegasoInit](#) ([RootFindingPegasoT](#) *pegasoObj)
Inicializa o objeto [RootFindingPegaso](#).
- void [RootFindingPegasoDelete](#) ([RootFindingPegasoT](#) *pegasoObj)
Apaga a instancia do objeto [RootFindingPegaso](#).
- static [RootFindingDoubleT](#) [getNextX](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem o valor para o proximo x.
- [RootFindingBoolT](#) [RootFindingPegasoPerformIteration](#) ([RootFindingPegasoT](#) *pegasoObj)
Realiza a iteracao.
- int [RootFindingPegasoGetErrorCode](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem o codigo de erro.
- int [RootFindingPegasoGetStateCode](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem o codigo referente ao estado do objeto.
- const char * [RootFindingPegasoGetErrorMessage](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem a mensagem de erro.
- const char * [RootFindingPegasoGetStateMessage](#) ([RootFindingPegasoT](#) *pegasoObj)
Obtem a mensagem referente ao estado do objeto.
- [RootFindingBoolT](#) [RootFindingPegasoHasError](#) ([RootFindingPegasoT](#) *pegasoObj)
Verifica se ha erros.

Variables

- static char [msg](#) [255]

7.15.1 Detailed Description

/root-finding/src/RootFindingPegaso.c

Date:

25/03/2010

Author:

Matheus Neder <matheusneder@gmail.com>

Definition in file [RootFindingPegaso.c](#).

7.15.2 Variable Documentation

7.15.2.1 char msg[255] [static]

Definition at line 173 of file RootFindingPegaso.c.

Index

- `_bissecao`
 - `BISSECAO_DEFAULT_MAX_ITERATIONS`, 16
 - `BISSECAO_DEFAULT_TOLERANCE`, 16
 - `computeX`, 16
 - `resetError`, 16
 - `RootFindingBissecaoCreate`, 16
 - `RootFindingBissecaoDelete`, 17
 - `RootFindingBissecaoFindNewRange`, 17
 - `RootFindingBissecaoGetErrorCode`, 17
 - `RootFindingBissecaoGetErrorMessage`, 17
 - `RootFindingBissecaoGetStateCode`, 17
 - `RootFindingBissecaoGetStateMessage`, 18
 - `RootFindingBissecaoHasError`, 18
 - `RootFindingBissecaoInit`, 18
 - `RootFindingBissecaoPerformIteration`, 18
 - `RootFindingBissecaoT`, 16
 - `setError`, 19
 - `_cordas`
 - `CORDAS_DEFAULT_MAX_ITERATIONS`, 20
 - `CORDAS_DEFAULT_TOLERANCE`, 20
 - `getNextX`, 21
 - `resetError`, 21
 - `RootFindingCordasCreate`, 21
 - `RootFindingCordasDelete`, 21
 - `RootFindingCordasGetErrorCode`, 22
 - `RootFindingCordasGetErrorMessage`, 22
 - `RootFindingCordasGetStateCode`, 22
 - `RootFindingCordasGetStateMessage`, 22
 - `RootFindingCordasHasError`, 22
 - `RootFindingCordasInit`, 22
 - `RootFindingCordasPerformIteration`, 23
 - `RootFindingCordasT`, 21
 - `setError`, 23
 - `_messages`
 - `MSG_BISSECAO_APPROXIMATION_ROOT_FOUND`, 4
 - `MSG_BISSECAO_ERROR_FOUND`, 4
 - `MSG_BISSECAO_INITIALIZED`, 4
 - `MSG_BISSECAO_MAX_ITERATIONS_LIMIT_REACHED`, 4
 - `MSG_BISSECAO_NO_ERROR`, 4
 - `MSG_BISSECAO_NOT_INIT`, 4
 - `MSG_BISSECAO_X_ISINF_OR_ISNAN_ERROR`, 4
 - `MSG_CORDAS_2NDIFF_TEST_ERROR`, 5
 - `MSG_CORDAS_APPROXIMATION_ROOT_FOUND`, 5
 - `MSG_CORDAS_ERROR_FOUND`, 5
 - `MSG_CORDAS_INITIALIZED`, 5
 - `MSG_CORDAS_MAX_ITERATIONS_LIMIT_REACHED`, 5
 - `MSG_CORDAS_NO_ERROR`, 5
 - `MSG_CORDAS_NOT_INIT`, 5
 - `MSG_CORDAS_X_ISINF_OR_ISNAN_ERROR`, 5
 - `MSG_GENERIC_2NDIFF_TEST_ERROR`, 5
 - `MSG_GENERIC_APPROXIMATION_ROOT_FOUND`, 6
 - `MSG_GENERIC_ERROR_FOUND`, 6
 - `MSG_GENERIC_INITIALIZED`, 6
 - `MSG_GENERIC_MAX_ITERATIONS_LIMIT_REACHED`, 6
 - `MSG_GENERIC_NO_ERROR`, 6
 - `MSG_GENERIC_NOT_INIT`, 6
 - `MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR`, 6
 - `MSG_NEWTON_2NDIFF_TEST_ERROR`, 6
 - `MSG_NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_EQUALS_ERROR`, 7
 - `MSG_NEWTON_APPROXIMATION_ROOT_FOUND`, 7
 - `MSG_NEWTON_ERROR_FOUND`, 7
 - `MSG_NEWTON_INITIALIZED`, 7
 - `MSG_NEWTON_MAX_ITERATIONS_LIMIT_REACHED`, 7
 - `MSG_NEWTON_NO_ERROR`, 7
 - `MSG_NEWTON_NOT_INIT`, 7
 - `MSG_NEWTON_X_ISINF_OR_ISNAN_ERROR`, 7
 - `MSG_PEGASO_APPROXIMATION_ROOT_FOUND`, 8
 - `MSG_PEGASO_ERROR_FOUND`, 8
 - `MSG_PEGASO_INITIALIZED`, 8
 - `MSG_PEGASO_MAX_ITERATIONS_LIMIT_REACHED`, 8
 - `MSG_PEGASO_NO_ERROR`, 8
 - `MSG_PEGASO_NOT_INIT`, 8
 - `MSG_PEGASO_X_ISINF_OR_ISNAN_ERROR`, 8
 - `MSG_ROOTS_EXACT_ROOT_FOUND`, 9
 - `MSG_ROOTS_MUP_EVAL_ERROR`, 9
 - `MSG_ROOTS_NO_ERROR`, 9
 - `MSG_ROOTS_RANGE_ERROR_FA_OR_FB_ISINFINITY`, 9
 - `MSG_ROOTS_RANGE_ERROR_PROD_FA_FB_NOT_LT_0`, 9
 - `MSG_ROOTS_RANGE_NOT_SET`, 9
 - `MSG_ROOTS_READY`, 9

- MSG_ROOTS_UNKNOW_ERROR, 9
- MSG_ROOTS_UNKNOW_STATE, 10
- _newton
 - getNextX, 25
 - NEWTON_DEFAULT_MAX_ITERATIONS, 25
 - NEWTON_DEFAULT_TOLERANCE, 25
 - resetError, 25
 - RootFindingNewtonRhapsonCreate, 25
 - RootFindingNewtonRhapsonDelete, 26
 - RootFindingNewtonRhapsonGetErrorcode, 26
 - RootFindingNewtonRhapsonGetErrorMessage, 26
 - RootFindingNewtonRhapsonGetStateCode, 26
 - RootFindingNewtonRhapsonGetStateMessage, 26
 - RootFindingNewtonRhapsonHasError, 27
 - RootFindingNewtonRhapsonInit, 27
 - RootFindingNewtonRhapsonPerformIteration, 27
 - RootFindingNewtonRhapsonT, 25
 - setError, 28
- _pegaso
 - getNextX, 29
 - PEGASO_DEFAULT_MAX_ITERATIONS, 29
 - PEGASO_DEFAULT_TOLERANCE, 29
 - resetError, 30
 - RootFindingPegasoCreate, 30
 - RootFindingPegasoDelete, 30
 - RootFindingPegasoGetErrorcode, 30
 - RootFindingPegasoGetErrorMessage, 30
 - RootFindingPegasoGetStateCode, 31
 - RootFindingPegasoGetStateMessage, 31
 - RootFindingPegasoHasError, 31
 - RootFindingPegasoInit, 31
 - RootFindingPegasoPerformIteration, 32
 - RootFindingPegasoT, 29
 - setError, 32
- _roots
 - isRangeError, 11
 - RootFindingBase2nDiffEval, 11
 - RootFindingBaseCreate, 12
 - RootFindingBaseDelete, 12
 - RootFindingBaseDiffEval, 12
 - RootFindingBaseEval, 12
 - RootFindingBaseGetErrorcode, 13
 - RootFindingBaseGetErrorMessage, 13
 - RootFindingBaseGetStateCode, 13
 - RootFindingBaseGetStateMessage, 13
 - RootFindingBaseHasError, 13
 - RootFindingBaseReset, 14
 - RootFindingBaseSetRange, 14
 - RootFindingBaseT, 11
- a
 - RootFindingBase, 34
- b
 - RootFindingBase, 34
 - BISSECAO_APPROXIMATION_ROOT_FOUND
 - RootFindingBissecao, 36
 - BISSECAO_ERROR_FOUND
 - RootFindingBissecao, 36
 - BISSECAO_INITIALIZED
 - RootFindingBissecao, 36
 - BISSECAO_MAX_ITERATIONS_LIMIT_REACHED
 - RootFindingBissecao, 36
 - BISSECAO_NO_ERROR
 - RootFindingBissecao, 37
 - BISSECAO_NOT_INIT
 - RootFindingBissecao, 36
 - BISSECAO_X_ISINF_OR_ISNAN_ERROR
 - RootFindingBissecao, 37
 - BISSECAO_DEFAULT_MAX_ITERATIONS_bissecao, 16
 - BISSECAO_DEFAULT_TOLERANCE_bissecao, 16
- c
 - RootFindingCordas, 39
- computeX
 - _bissecao, 16
- CORDAS_2NDIFF_TEST_ERROR
 - RootFindingCordas, 39
- CORDAS_APPROXIMATION_ROOT_FOUND
 - RootFindingCordas, 39
- CORDAS_ERROR_FOUND
 - RootFindingCordas, 39
- CORDAS_INITIALIZED
 - RootFindingCordas, 39
- CORDAS_MAX_ITERATIONS_LIMIT_REACHED
 - RootFindingCordas, 39
- CORDAS_NO_ERROR
 - RootFindingCordas, 39
- CORDAS_NOT_INIT
 - RootFindingCordas, 39
- CORDAS_X_ISINF_OR_ISNAN_ERROR
 - RootFindingCordas, 39
- CORDAS_DEFAULT_MAX_ITERATIONS_cordas, 20
- CORDAS_DEFAULT_TOLERANCE_cordas, 20
- e

- RootFindingBase, 34
- errorCode
 - RootFindingBase, 35
 - RootFindingBissecacao, 37
 - RootFindingCordas, 40
 - RootFindingNewtonRhapson, 42
 - RootFindingPegaso, 45
- FALSE
 - RootFindingCommon.h, 51
- fC
 - RootFindingCordas, 39
- fPrevXi
 - RootFindingPegaso, 44
- fX
 - RootFindingBase, 34
- getAxisVarPtr
 - RootFindingCommon.c, 62
 - RootFindingCommon.h, 52
- getNextX
 - _cordas, 21
 - _newton, 25
 - _pegaso, 29
- i
 - RootFindingBissecacao, 37
 - RootFindingCordas, 39
 - RootFindingNewtonRhapson, 42
 - RootFindingPegaso, 44
- include/messages/RootFindingMessages.h, 45
- include/messages/RootFindingMessages_PT-BR.h, 45
- include/RootFinding.h, 47
- include/RootFindingBase.h, 47
- include/RootFindingBissecacao.h, 49
- include/RootFindingCommon.h, 50
- include/RootFindingCordas.h, 53
- include/RootFindingNewtonRhapson.h, 55
- include/RootFindingPegaso.h, 56
- infinity
 - RootFindingCommon.c, 62
 - RootFindingCommon.h, 52
- isInfOrNan
 - RootFindingCommon.c, 62
 - RootFindingCommon.h, 52
- isRangeError
 - _roots, 11
- maxIterations
 - RootFindingBissecacao, 37
 - RootFindingCordas, 40
 - RootFindingNewtonRhapson, 42
 - RootFindingPegaso, 44
- Mensagens de estados e erros, 3
- Metodo da Bissecacao, 14
- Metodo das Cordas, 19
- Metodo de Newton-Rhapson, 23
- Metodo de Pegaso, 28
- msg
 - RootFindingBase.c, 59
 - RootFindingBissecacao.c, 61
 - RootFindingCordas.c, 65
 - RootFindingNewtonRhapson.c, 66
 - RootFindingPegaso.c, 68
- MSG_BISSECAO_APPROXIMATION_-
ROOT_FOUND
 - _messages, 4
- MSG_BISSECAO_ERROR_FOUND
 - _messages, 4
- MSG_BISSECAO_INITIALIZED
 - _messages, 4
- MSG_BISSECAO_MAX_ITERATIONS_LIMIT_-
REACHED
 - _messages, 4
- MSG_BISSECAO_NO_ERROR
 - _messages, 4
- MSG_BISSECAO_NOT_INIT
 - _messages, 4
- MSG_BISSECAO_X_ISINF_OR_ISNAN_-
ERROR
 - _messages, 4
- MSG_CORDAS_2NDIFF_TEST_ERROR
 - _messages, 5
- MSG_CORDAS_APPROXIMATION_ROOT_-
FOUND
 - _messages, 5
- MSG_CORDAS_ERROR_FOUND
 - _messages, 5
- MSG_CORDAS_INITIALIZED
 - _messages, 5
- MSG_CORDAS_MAX_ITERATIONS_LIMIT_-
REACHED
 - _messages, 5
- MSG_CORDAS_NO_ERROR
 - _messages, 5
- MSG_CORDAS_NOT_INIT
 - _messages, 5
- MSG_CORDAS_X_ISINF_OR_ISNAN_ERROR
 - _messages, 5
- MSG_GENERIC_2NDIFF_TEST_ERROR
 - _messages, 5
- MSG_GENERIC_APPROXIMATION_ROOT_-
FOUND
 - _messages, 6
- MSG_GENERIC_ERROR_FOUND
 - _messages, 6
- MSG_GENERIC_INITIALIZED

- [_messages](#), [6](#)
- MSG_GENERIC_MAX_ITERATIONS_LIMIT_-
REACHED
 - [_messages](#), [6](#)
- MSG_GENERIC_NO_ERROR
 - [_messages](#), [6](#)
- MSG_GENERIC_NOT_INIT
 - [_messages](#), [6](#)
- MSG_GENERIC_X_ISINF_OR_ISNAN_ERROR
 - [_messages](#), [6](#)
- MSG_NEWTON_2NDIFF_TEST_ERROR
 - [_messages](#), [6](#)
- MSG_NEWTON_2NDIFFA_2NDIFFB_SIGN_-
NOT_EQUALS_ERROR
 - [_messages](#), [7](#)
- MSG_NEWTON_APPROXIMATION_ROOT_-
FOUND
 - [_messages](#), [7](#)
- MSG_NEWTON_ERROR_FOUND
 - [_messages](#), [7](#)
- MSG_NEWTON_INITIALIZED
 - [_messages](#), [7](#)
- MSG_NEWTON_MAX_ITERATIONS_LIMIT_-
REACHED
 - [_messages](#), [7](#)
- MSG_NEWTON_NO_ERROR
 - [_messages](#), [7](#)
- MSG_NEWTON_NOT_INIT
 - [_messages](#), [7](#)
- MSG_NEWTON_X_ISINF_OR_ISNAN_ERROR
 - [_messages](#), [7](#)
- MSG_PEGASO_APPROXIMATION_ROOT_-
FOUND
 - [_messages](#), [8](#)
- MSG_PEGASO_ERROR_FOUND
 - [_messages](#), [8](#)
- MSG_PEGASO_INITIALIZED
 - [_messages](#), [8](#)
- MSG_PEGASO_MAX_ITERATIONS_LIMIT_-
REACHED
 - [_messages](#), [8](#)
- MSG_PEGASO_NO_ERROR
 - [_messages](#), [8](#)
- MSG_PEGASO_NOT_INIT
 - [_messages](#), [8](#)
- MSG_PEGASO_X_ISINF_OR_ISNAN_ERROR
 - [_messages](#), [8](#)
- MSG_ROOTS_EXACT_ROOT_FOUND
 - [_messages](#), [9](#)
- MSG_ROOTS_MUP_EVAL_ERROR
 - [_messages](#), [9](#)
- MSG_ROOTS_NO_ERROR
 - [_messages](#), [9](#)
- MSG_ROOTS_RANGE_ERROR_FA_OR_FB_-
ISINFINITY
 - [_messages](#), [9](#)
- MSG_ROOTS_RANGE_ERROR_PROD_FA_-
FB_NOT_LT_0
 - [_messages](#), [9](#)
- MSG_ROOTS_RANGE_NOT_SET
 - [_messages](#), [9](#)
- MSG_ROOTS_READY
 - [_messages](#), [9](#)
- MSG_ROOTS_UNKNOW_ERROR
 - [_messages](#), [9](#)
- MSG_ROOTS_UNKNOW_STATE
 - [_messages](#), [10](#)
- Mup2ndDiff
 - RootFindingCommon.c, [63](#)
 - RootFindingCommon.h, [53](#)
- MupDiff
 - RootFindingCommon.c, [63](#)
 - RootFindingCommon.h, [53](#)
- mupObj
 - RootFindingBase, [35](#)
- NEWTON_2NDIFF_TEST_ERROR
 - RootFindingNewtonRhapson, [42](#)
- NEWTON_2NDIFFA_2NDIFFB_SIGN_NOT_-
EQUALS_ERROR
 - RootFindingNewtonRhapson, [42](#)
- NEWTON_APPROXIMATION_ROOT_-
FOUND
 - RootFindingNewtonRhapson, [41](#)
- NEWTON_ERROR_FOUND
 - RootFindingNewtonRhapson, [41](#)
- NEWTON_INITIALIZED
 - RootFindingNewtonRhapson, [41](#)
- NEWTON_MAX_ITERATIONS_LIMIT_-
REACHED
 - RootFindingNewtonRhapson, [41](#)
- NEWTON_NO_ERROR
 - RootFindingNewtonRhapson, [42](#)
- NEWTON_NOT_INIT
 - RootFindingNewtonRhapson, [41](#)
- NEWTON_X_ISINF_OR_ISNAN_ERROR
 - RootFindingNewtonRhapson, [42](#)
- NEWTON_DEFAULT_MAX_ITERATIONS
 - [_newton](#), [25](#)
- NEWTON_DEFAULT_TOLERANCE
 - [_newton](#), [25](#)
- Parte Generica, [10](#)
- PEGASO_APPROXIMATION_ROOT_FOUND
 - RootFindingPegaso, [44](#)
- PEGASO_ERROR_FOUND
 - RootFindingPegaso, [44](#)

- PEGASO_INITIALIZED
 - RootFindingPegaso, [44](#)
- PEGASO_MAX_ITERATIONS_LIMIT_-REACHED
 - RootFindingPegaso, [44](#)
- PEGASO_NO_ERROR
 - RootFindingPegaso, [44](#)
- PEGASO_NOT_INIT
 - RootFindingPegaso, [44](#)
- PEGASO_X_ISINF_OR_ISNAN_ERROR
 - RootFindingPegaso, [44](#)
- PEGASO_DEFAULT_MAX_ITERATIONS
 - _pegaso, [29](#)
- PEGASO_DEFAULT_TOLERANCE
 - _pegaso, [29](#)
- prevXi
 - RootFindingPegaso, [44](#)
- RANGE_ERRORS_END
 - RootFindingBase, [33](#)
- RANGE_ERRORS_START
 - RootFindingBase, [33](#)
- resetError
 - _bisecao, [16](#)
 - _cordas, [21](#)
 - _newton, [25](#)
 - _pegaso, [30](#)
- RootFindingBase, [32](#)
 - a, [34](#)
 - b, [34](#)
 - e, [34](#)
 - errorCode, [35](#)
 - fX, [34](#)
 - mupObj, [35](#)
 - RANGE_ERRORS_END, [33](#)
 - RANGE_ERRORS_START, [33](#)
 - ROOTS_EXACT_ROOT_FOUND, [34](#)
 - ROOTS_MUP_EVAL_ERROR, [33](#)
 - ROOTS_NO_ERROR, [33](#)
 - ROOTS_RANGE_ERROR_FA_OR_FB_-ISINFINITY, [33](#)
 - ROOTS_RANGE_ERROR_PROD_FA_FB_-NOT_LT_0, [33](#)
 - ROOTS_RANGE_NOT_SET, [34](#)
 - ROOTS_READY, [34](#)
 - state, [35](#)
 - varPtr, [35](#)
 - x, [34](#)
- RootFindingBase.c
 - msg, [59](#)
- RootFindingBase2nDiffEval
 - _roots, [11](#)
- RootFindingBaseCreate
 - _roots, [12](#)
- RootFindingBaseDelete
 - _roots, [12](#)
- RootFindingBaseDiffEval
 - _roots, [12](#)
- RootFindingBaseEval
 - _roots, [12](#)
- RootFindingBaseGetErrorCode
 - _roots, [13](#)
- RootFindingBaseGetErrorMessage
 - _roots, [13](#)
- RootFindingBaseGetStateCode
 - _roots, [13](#)
- RootFindingBaseGetStateMessage
 - _roots, [13](#)
- RootFindingBaseHasError
 - _roots, [13](#)
- RootFindingBaseReset
 - _roots, [14](#)
- RootFindingBaseSetRange
 - _roots, [14](#)
- RootFindingBaseT
 - _roots, [11](#)
- RootFindingBisecao, [35](#)
 - BISSECAO_APPROXIMATION_ROOT_-FOUND, [36](#)
 - BISSECAO_ERROR_FOUND, [36](#)
 - BISSECAO_INITIALIZED, [36](#)
 - BISSECAO_MAX_ITERATIONS_LIMIT_-REACHED, [36](#)
 - BISSECAO_NO_ERROR, [37](#)
 - BISSECAO_NOT_INIT, [36](#)
 - BISSECAO_X_ISINF_OR_ISNAN_ERROR, [37](#)
 - errorCode, [37](#)
 - i, [37](#)
 - maxIterations, [37](#)
 - rootsObj, [37](#)
 - state, [37](#)
 - tolerance, [37](#)
- RootFindingBisecao.c
 - msg, [61](#)
- RootFindingBisecaoCreate
 - _bisecao, [16](#)
- RootFindingBisecaoDelete
 - _bisecao, [17](#)
- RootFindingBisecaoFindNewRange
 - _bisecao, [17](#)
- RootFindingBisecaoGetErrorCode
 - _bisecao, [17](#)
- RootFindingBisecaoGetErrorMessage
 - _bisecao, [17](#)
- RootFindingBisecaoGetStateCode
 - _bisecao, [17](#)
- RootFindingBisecaoGetStateMessage

- [_bissecao](#), [18](#)
- RootFindingBissecaoHasError
 - [_bissecao](#), [18](#)
- RootFindingBissecaoInit
 - [_bissecao](#), [18](#)
- RootFindingBissecaoPerformIteration
 - [_bissecao](#), [18](#)
- RootFindingBissecaoT
 - [_bissecao](#), [16](#)
- RootFindingBoolT
 - RootFindingCommon.h, [52](#)
- RootFindingCommon.c
 - [getAxisVarPtr](#), [62](#)
 - [infinity](#), [62](#)
 - [isInfOrNan](#), [62](#)
 - [Mup2ndDiff](#), [63](#)
 - [MupDiff](#), [63](#)
- RootFindingCommon.h
 - [FALSE](#), [51](#)
 - [getAxisVarPtr](#), [52](#)
 - [infinity](#), [52](#)
 - [isInfOrNan](#), [52](#)
 - [Mup2ndDiff](#), [53](#)
 - [MupDiff](#), [53](#)
 - [RootFindingBoolT](#), [52](#)
 - [RootFindingDoubleT](#), [52](#)
 - [TRUE](#), [51](#)
- RootFindingCordas, [38](#)
 - [c](#), [39](#)
 - [CORDAS_2NDIFF_TEST_ERROR](#), [39](#)
 - [CORDAS_APPROXIMANTION_ROOT_-FOUND](#), [39](#)
 - [CORDAS_ERROR_FOUND](#), [39](#)
 - [CORDAS_INITIALIZED](#), [39](#)
 - [CORDAS_MAX_ITERATIONS_LIMIT_-REACHED](#), [39](#)
 - [CORDAS_NO_ERROR](#), [39](#)
 - [CORDAS_NOT_INIT](#), [39](#)
 - [CORDAS_X_ISINF_OR_ISNAN_ERROR](#), [39](#)
 - [errorCode](#), [40](#)
 - [fC](#), [39](#)
 - [i](#), [39](#)
 - [maxIterations](#), [40](#)
 - [rootsObj](#), [39](#)
 - [state](#), [40](#)
 - [tolerance](#), [40](#)
- RootFindingCordas.c
 - [msg](#), [65](#)
- RootFindingCordasCreate
 - [_cordas](#), [21](#)
- RootFindingCordasDelete
 - [_cordas](#), [21](#)
- RootFindingCordasGetErrorCode
 - [_cordas](#), [22](#)
- RootFindingCordasGetErrorMessage
 - [_cordas](#), [22](#)
- RootFindingCordasGetStateCode
 - [_cordas](#), [22](#)
- RootFindingCordasGetStateMessage
 - [_cordas](#), [22](#)
- RootFindingCordasHasError
 - [_cordas](#), [22](#)
- RootFindingCordasInit
 - [_cordas](#), [22](#)
- RootFindingCordasPerformIteration
 - [_cordas](#), [23](#)
- RootFindingCordasT
 - [_cordas](#), [21](#)
- RootFindingDoubleT
 - RootFindingCommon.h, [52](#)
- RootFindingNewtonRhapson, [40](#)
 - [errorCode](#), [42](#)
 - [i](#), [42](#)
 - [maxIterations](#), [42](#)
 - [NEWTON_2NDIFF_TEST_ERROR](#), [42](#)
 - [NEWTON_2NDIFFA_2NDIFFB_SIGN_-NOT_EQUALS_ERROR](#), [42](#)
 - [NEWTON_APPROXIMANTION_ROOT_-FOUND](#), [41](#)
 - [NEWTON_ERROR_FOUND](#), [41](#)
 - [NEWTON_INITIALIZED](#), [41](#)
 - [NEWTON_MAX_ITERATIONS_LIMIT_-REACHED](#), [41](#)
 - [NEWTON_NO_ERROR](#), [42](#)
 - [NEWTON_NOT_INIT](#), [41](#)
 - [NEWTON_X_ISINF_OR_ISNAN_ERROR](#), [42](#)
 - [rootsObj](#), [42](#)
 - [state](#), [42](#)
 - [tolerance](#), [42](#)
- RootFindingNewtonRhapson.c
 - [msg](#), [66](#)
- RootFindingNewtonRhapsonCreate
 - [_newton](#), [25](#)
- RootFindingNewtonRhapsonDelete
 - [_newton](#), [26](#)
- RootFindingNewtonRhapsonGetErrorCode
 - [_newton](#), [26](#)
- RootFindingNewtonRhapsonGetErrorMessage
 - [_newton](#), [26](#)
- RootFindingNewtonRhapsonGetStateCode
 - [_newton](#), [26](#)
- RootFindingNewtonRhapsonGetStateMessage
 - [_newton](#), [26](#)
- RootFindingNewtonRhapsonHasError
 - [_newton](#), [27](#)
- RootFindingNewtonRhapsonInit

- [_newton](#), 27
- RootFindingNewtonRhapsonPerformIteration
 - [_newton](#), 27
- RootFindingNewtonRhapsonT
 - [_newton](#), 25
- RootFindingPegaso, 43
 - [errorCode](#), 45
 - [fPrevXi](#), 44
 - [i](#), 44
 - [maxIterations](#), 44
 - PEGASO_APPROXIMATION_ROOT_-
FOUND, 44
 - PEGASO_ERROR_FOUND, 44
 - PEGASO_INITIALIZED, 44
 - PEGASO_MAX_ITERATIONS_LIMIT_-
REACHED, 44
 - PEGASO_NO_ERROR, 44
 - PEGASO_NOT_INIT, 44
 - PEGASO_X_ISINF_OR_ISNAN_ERROR,
44
 - [prevXi](#), 44
 - [rootsObj](#), 44
 - [state](#), 45
 - [tolerance](#), 44
- RootFindingPegaso.c
 - [msg](#), 68
- RootFindingPegasoCreate
 - [_pegaso](#), 30
- RootFindingPegasoDelete
 - [_pegaso](#), 30
- RootFindingPegasoGetErrorCode
 - [_pegaso](#), 30
- RootFindingPegasoGetErrorMessage
 - [_pegaso](#), 30
- RootFindingPegasoGetStateCode
 - [_pegaso](#), 31
- RootFindingPegasoGetStateMessage
 - [_pegaso](#), 31
- RootFindingPegasoHasError
 - [_pegaso](#), 31
- RootFindingPegasoInit
 - [_pegaso](#), 31
- RootFindingPegasoPerformIteration
 - [_pegaso](#), 32
- RootFindingPegasoT
 - [_pegaso](#), 29
- ROOTS_EXACT_ROOT_FOUND
 - RootFindingBase, 34
- ROOTS_MUP_EVAL_ERROR
 - RootFindingBase, 33
- ROOTS_NO_ERROR
 - RootFindingBase, 33
- ROOTS_RANGE_ERROR_FA_OR_FB_-
ISINFINITY
 - RootFindingBase, 33
- ROOTS_RANGE_ERROR_PROD_FA_FB_-
NOT_LT_0
 - RootFindingBase, 33
- ROOTS_RANGE_NOT_SET
 - RootFindingBase, 34
- ROOTS_READY
 - RootFindingBase, 34
- rootsObj
 - RootFindingBissecas, 37
 - RootFindingCordas, 39
 - RootFindingNewtonRhapson, 42
 - RootFindingPegaso, 44
- setError
 - [_bissecas](#), 19
 - [_cordas](#), 23
 - [_newton](#), 28
 - [_pegaso](#), 32
- src/RootFindingBase.c, 58
- src/RootFindingBissecas.c, 59
- src/RootFindingCommon.c, 61
- src/RootFindingCordas.c, 63
- src/RootFindingNewtonRhapson.c, 65
- src/RootFindingPegaso.c, 67
- state
 - RootFindingBase, 35
 - RootFindingBissecas, 37
 - RootFindingCordas, 40
 - RootFindingNewtonRhapson, 42
 - RootFindingPegaso, 45
- tolerance
 - RootFindingBissecas, 37
 - RootFindingCordas, 40
 - RootFindingNewtonRhapson, 42
 - RootFindingPegaso, 44
- TRUE
 - RootFindingCommon.h, 51
- varPtr
 - RootFindingBase, 35
- x
 - RootFindingBase, 34