Manual



Índice

[Requisitos 2](#_Toc357939144)

[Para utilizador 2](#_Toc357939145)

[Para desenvolvedor 2](#_Toc357939146)

[Manual de utilizador 3](#_Toc357939147)

[Barra de ferramentas 4](#_Toc357939148)

[Algoritmos 4](#_Toc357939149)

[Eventos 4](#_Toc357939150)

[Opções 5](#_Toc357939151)

[Controlos 8](#_Toc357939152)

[Informação 8](#_Toc357939153)

[Barra de progresso 9](#_Toc357939154)

[Manual de desenvolvedor 10](#_Toc357939155)

[Compilar aplicação 11](#_Toc357939156)

[Windows 11](#_Toc357939157)

[Linux 11](#_Toc357939158)

[Gerar documentação do código 11](#_Toc357939159)

[Windows 11](#_Toc357939160)

[Linux 11](#_Toc357939161)

# Requisitos

## Para utilizador

|  |  |
| --- | --- |
| Windows | Linux e Mac |
| .NET Framework 3.5 ou maior  <https://www.microsoft.com/pt-pt/download/details.aspx?id=21> | Mono 2.2 ou maior |
| Gnuplot (Opcional) - [http://www.gnuplot.info](http://www.gnuplot.info/) | |

## Para desenvolvedor

|  |  |
| --- | --- |
| Windows | Linux e Mac |
| Compilador Visual Studio C# OU Mono  .NET Framework 3.5 ou maior  <https://www.microsoft.com/pt-pt/download/details.aspx?id=21> | Mono 2.2 ou maior  Mono libraries  Mono devel  Mono utils  xbuild  OU  Mono-complete |
| Doxygen (Gera a documentação do projeto em HTML)  Doxygen-latex (Gera a documentação do projeto em PDF) | |

# Manual de utilizador



Figura

## Barra de ferramentas



Figura

Seleciona todos os algoritmos.

Desselecciona todos os algoritmos.

Inverte a seleção.

Mostra ou esconde o registo de eventos.

Número de algoritmos selecionados.

## Algoritmos



Figura

Escolha de algoritmos a testar.

## Eventos



Figura

Visualizador de eventos, todos os passos da aplicação serão reportados para esta janela.

Esta janela será automaticamente visível ao executar uma tarefa.

## Opções



Figura



**Gnuplot.exe:** Caminho do executável para a aplicação *gnuplot*.



**Save reports to:** Caminho / pasta onde são guardados os relatórios de execução e eventos.



**Auto open generated lot files:** Quando ativo, os relatórios gerados pelo programa são automaticamente mostrados. Caso contrário é necessário abrir manualmente o ficheiro de relatório com o *gnuplot*.



**Number of tests:** Número de testes a realizar por cada algoritmo.



**Compute average:** Calcular a média de **X** execuções para cada teste (Repetições). Caso ative a opção da direita, o menor valor e o maior valor (tempo de execução) irá ser excluído da média, permitindo assim ter uma média mais fiável cortando os picos.

Ao escolher o valor de 1 não será efetuado um cálculo de média, assim a primeira e a única execução será logo usada para construir o gráfico.

**Nota 1:** Esta função permite uma medição mais estável e fiável, construindo um gráfico mais preciso. No entanto o tempo total de execução da tarefa irá ser maior consoante o número de repetições. (Tempo \* Numero de repetições).

**Nota 2:** Com a opção da direita ativada, dois valores serão excluídos da média, contando assim com apenas (numero – 2) valores. Para ativar esta opção é necessário pelo menos 3 repetições.



**Array initial size:** Tamanho inicial do array (número de elementos), neste caso o primeiro teste ao algoritmo vai conter 5000 elementos.

**Array grow factor:** Fator de crescimento do array, crescimento do array em **X** elementos por teste. É possível escolher o tipo de crescimento por soma OU multiplicação.

|  |  |
| --- | --- |
| **Exemplo 1:**  Número de testes: 5  Array initial size: 5000  Array grow factor: += 5000  1º Teste - Array: 5000 elementos  2º Teste - Array: 10000 elementos  3º Teste - Array: 15000 elementos  4º Teste - Array: 20000 elementos  5º Teste - Array: 25000 elementos | **Exemplo 2:**  Número de testes: 5  Array initial size: 500  Array grow factor: \*= 2  1º Teste - Array: 500 elementos  2º Teste - Array: 1000 elementos  3º Teste - Array: 2000 elementos  4º Teste - Array: 4000 elementos  5º Teste - Array: 8000 elementos |



**Min rand number:** Numero aleatório mínimo possível no array.

**Max rand number:** Número aleatório máximo possível no array.

**Random numbers between values:** Quando ativo o número gerado tem que estar dentro do valor mínimo e máximo imposto. Caso contrário será utilizado um fator de crescimento do número consoante o teste.

|  |  |
| --- | --- |
| **Exemplo 1:**  Número de testes: 5  Min rand number: 100  Max rand number: 655350  1º Teste –Number: Random(100, 655350)  2º Teste - Number: Random(100, 655350)  3º Teste - Number: Random(100, 655350)  4º Teste - Number: Random(100, 655350)  5º Teste - Number: Random(100, 655350) | **Exemplo 2:**  Número de testes: 5  Min rand number: 100  Number grow factor: \*= 5  1º Teste - Array: Random(100, 500)  2º Teste - Array: Random(100, 2500)  3º Teste - Array: Random(100, 12500)  4º Teste - Array: Random(100, 62500)  5º Teste - Array: Random(100, 312500) |

O valor máximo do número está limitado pela aplicação com um valor de: **2,147,483,647.**

## Controlos



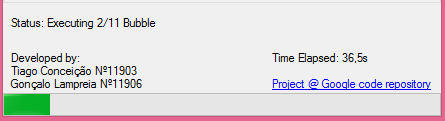
Figura

 Inicia a tarefa em segundo plano, todos os algoritmos selecionados serão testados usando as opções escolhidas.

Pausa a tarefa, apenas disponível se a tarefa estiver ativa. **Nota:** Só irá colocar a tarefa em repouso quando o algoritmo em questão acabar.

Para a tarefa por completo, os relatórios já gerados não serão perdidos.

## Informação



Figura

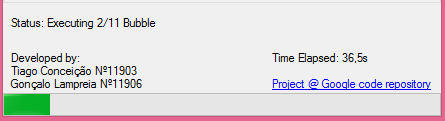
**Status:** Estado da aplicação, mostra o que a aplicação está a processar de momento.

**Time Elapsed:** Tempo decorrido desde o inicio da tarefa até agora ou até terminar.

**Developed by:** Autores / criadores da aplicação.

**Project @Google code repository:** Link para o repositório de código da aplicação, alojada no Google Code.

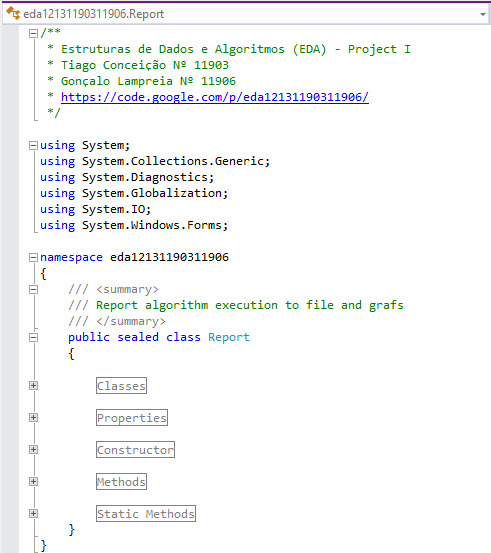
## Barra de progresso



Figura

Progresso atual da tarefa em percentagem.

# Manual de desenvolvedor



Figura

## Compilar aplicação

### Windows

* Abrir o explorador de ficheiros
* Ir à pasta do código do projecto (src)
* Executar o ficheiro: “build.bat”
* Após a execução com sucesso já é possível correr e utilizar o programa. (Será copiado o executável para a pasta: APPROOT/bin)

### Linux

* Abrir um terminal / consola
* Ir para o caminho onde se encontra o código do projecto (src)
* Executar o comando: make ou make all
* Após o comando make executar com sucesso já é possível correr e utilizar o programa. (Será copiado o executável para a pasta: APPROOT/bin)

## Gerar documentação do código

### Windows

* Abrir o explorador de ficheiros
* Ir para o caminho onde se encontra a pasta da documentação (sourcedoc)
* Executar o ficheiro: “build.bat”
* Após a execução com sucesso já é possível consultar a documentação. (Abrir o sourcedoc/índex.html OU Documentation.pdf)

### Linux

* Abrir um terminal / consola
* Ir para o caminho onde se encontra a pasta da documentação (sourcedoc)
* Executar o comando: make ou make all
  + make doxygen -> Apenas compila o HTML
  + make refman.pdf -> Apenas compila o PDF
* Após o comando make executar com sucesso já é possível consultar a documentação. (Abrir o sourcedoc/índex.html OU Documentation.pdf)