

Introdução à Computação

Licenciatura em Videojogos

1º Trabalho - Jogo de ordenação de números

1º Semestre - 2019/2020

1 Introdução

O objectivo deste trabalho é desenvolver um jogo utilizando a linguagem de programação C. Trata-se de um jogo para estimular o cérebro em que o jogador terá de ordenar números gerados aleatoriamente pelo computador.

2 Descrição do problema

O jogo tem início no nível 1 e com 0 pontos. Em cada ronda o computador deverá gerar 4 números aleatórios e ficar à espera que o jogador introduza esses mesmo 4 números por ordem crescente. Caso o jogador introduza os números pela ordem certa, ganha essa ronda e a sua pontuação aumenta em 5 pontos. Quando a pontuação do jogador ultrapassa o número de pontos para passar ao nível seguinte (ver Tabela 2.1), o nível deverá aumentar 1 unidade.

Tabela 2.1: Pontuações por nível

Nível Actual	Pontuação para passar para o nível seguinte
1	10
2	20
3	30
4	40
5	50

Em cada nível o intervalo de números gerados varia de acordo com a Tabela 2.2. Ou seja, no nível 1, o programa gera 4 números aleatórios entre 0 e 10. No nível 2, o programa gera 4 números aleatórios entre 0 e 30, e assim sucessivamente. A cada jogada (ronda) o contador de jogadas deverá ser incrementado. O limite máximo de rondas

Tabela 2.2: Pontuações por nível

Nível	Valor mínimo	Valor máximo
1	0	10
2	0	30
3	-50	30
4	-100	0
5	-200	-100

é de 30. Quando esse limite é atingido o jogo deve terminar.

O programa começa com a apresentação do programa e do menu:

WELCOME TO SORTIFY!

```
+-----+
| SORTIFY
| p - next challenge
| q - quit
| m - print this information
| s - sow your status
+-----+
```

O programa fica então à espera que o utilizador introduza uma das opções do menu. Se o utilizador introduzir a opção 'p', o programa apresenta a mensagem:

Sort the following numbers:

e na linha seguinte apresenta os 4 números separados por espaço e fica à espera que o utilizador introduza os mesmo 4 números por ordem crescente. Caso o utilizador introduza os números pela ordem certa o programa imprime a mensagem:

Well done!

caso contrário deverá imprimir a seguinte mensagem:

Wrong answer.

E deverá tomar as seguintes acções:

- Actualizar a pontuação
- Actualizar o nível
- Actualizar o número de rondas
- Verificar se o jogo terminou por ter atingido o número máximo de rondas ou por ter atingido o nível 6.
- Ficar à espera que o utilizador introduza uma das opções do menu.

Se o jogador atingir o máximo de jogadas o programa deve imprimir a seguinte mensagem

You have reached the maximum number of moves.

e em seguida deve mostrar o status do jogador seguido da mensagem

Game Over.

Se o jogador escolher a opção 'q' o programa deverá mostrar o status seguido da mensagem Bye. e depois deverá terminar.

2.1 Código base

Está disponível um repositório no GitHub [1] com o código fonte base que deverá ser o ponto de partida para o desenvolvimento da aplicação. Cada aluno deverá criar a sua conta no GitHub [2] e um dos elementos do grupo deverá fazer um *fork* do repositório [1].

Note que a função `rand_number` deverá ser utilizada para gerar cada um dos números aleatórios. A função `random` necessita de inicializar uma *seed*. Essa seed deverá ser passada como argumento na execução do programa. No seguinte a seed deverá ser o número 7:

```
./sortify 7
```

O programa deverá verificar se foi passada uma seed na linha de comando e caso não tenha sido passada, o programa deverá definir uma seed baseada no tempo actual (`time()`).

Deverá utilizar funções separadas para cada opção do menu.

2.2 Relatório e Qualidade do Código - 3 Valores

Cada grupo deverá incluir um breve relatório composto por:

- Título
- Nomes dos autores (primeiro e último) e respetivos números de alunos
- Nome do grupo de trabalho (de acordo com a plataforma pandora [3])
- Descrição da solução
- Estrutura do software
- Conclusões e matéria aprendida
- Referências, incluindo trocas de ideias com colegas, código aberto reutilizado e bibliotecas utilizadas

O código deverá estar correctamente indentado e comentado. É estritamente proibida a utilização da instrução `goto`. A utilização de variáveis globais (excepto constantes) é também proibida.

2.3 Compilação

O ficheiro fonte deve ser escrito em linguagem ANSI C e o nome do ficheiro deverá ser `convert.c`. O código deverá ser compilado no gcc [4] com as seguintes opções:

```
-g -Wall -std=c99 -Wpedantic -Wextra
```

3 Exemplo de execução

O seguinte exemplo mostra o decorrer de um jogo :

```
stu@linux:~/IC/projecto1$ ./main 7
WELCOME TO SORTIFY!
+-----+
| SORTIFY          |
| p - next challenge |
| q - quit          |
| m - print this information |
| s - sow your status |
+-----+
p
Sort the following numbers:
4, 7, 8, 4
4 4 7 8
Well done!
p
Sort the following numbers:
1, 5, 6, 9
1 5 6 9
Well done!
p
Sort the following numbers:
8, 13, 28, 13
8 23 13 13
Please sort the numbers:
8 28 13 13
Wrong answer.
p
Sort the following numbers:
```

```

8, 23, 23, 29
8 23 23 29
Well done!
x
Unknown option.
p
Sort the following numbers:
29, 7, 4, 29
4 7 29 29
Well done!
p
Sort the following numbers:
-6, 12, 22, 15
-6 12 15 22
Well done!
s
+-----+
| level: 03          |
| points: 25         |
| plays: 06          |
+-----+
p
Sort the following numbers:
11, -46, -31, -41
-46 -41 -31 11
Well done!
p
Sort the following numbers:
-47, -6, -16, -30
-47 -30 -16 -6
Well done!
p
Sort the following numbers:
-28, -18, -65, -42
-65 -42 -28 -18
Well done!
p
Sort the following numbers:
-192, -178, -196, -100
-196 -192 -178 -100
Well done!
p
Sort the following numbers:
-179, -122, -170, -199
-199 -179 -170 -122
Well done!
Congratulations, you win!
+-----+
| level: 06          |
| points: 50         |
| plays: 11          |
+-----+
Game Over.

```

4 Material a entregar

Cada grupo deverá entregar no moodle um ficheiro zip contendo

- Ficheiro ‘.c’ com código devidamente comentado e indentado.
- A pasta ‘.git’ onde deverá ser possível visualizar *commits* feitos pelos vários elementos do grupo
- Um relatório em formato pdf.

O nome do ficheiro zip deverá ser o nome do grupo de acordo com a plataforma pandora[3]. **Apenas um elemento do grupo deve submeter o ficheiro no Moodle.**

5 Forma e data de entrega

Os alunos deverão submeter os seus trabalhos na plataforma de testes Pandora [3]. A plataforma irá realizar vários testes. O número de testes bem sucedidos irá determinar a nota mínima e máxima do grupo. A plataforma apenas aceitará trabalhos até às 23:55 de 8 de Dezembro de 2019.

Adicionalmente os alunos deverão que entregar através da plataforma moodle o ficheiro zip com o material descrito na secção 4 até à mesma data e hora da submissão no Pandora. Os trabalhos entregues depois desta data não serão avaliados.

6 Peso na avaliação

Este trabalho será cotado de 0 a 20 valores e terá um peso de 15% da nota da componente prática da unidade curricular.

7 Honestidade académica

Nesta disciplina, espera-se que cada aluno siga os mais altos padrões de honestidade académica. Isto significa que cada ideia que não seja do aluno deve ser claramente indicada, com devida referência ao respectivo autor. O não cumprimento desta regra constitui plágio [5].

O plágio inclui a utilização de ideias, código ou conjuntos de soluções de outros alunos ou indivíduos, ou quaisquer outras fontes para além dos textos de apoio à disciplina, sem dar o respectivo crédito a essas fontes. Os alunos são encorajados a discutir os problemas com outros alunos e devem mencionar essa discussão quando submetem os Projectos. Essa menção **não** influenciará a nota. Os alunos não deverão, no entanto, copiar códigos, documentação e relatórios de outros alunos, ou dar os seus próprios códigos, documentação e relatórios a outros em qualquer circunstância. De facto, não devem sequer deixar códigos, documentação e relatórios em computadores de uso partilhado.

Nesta disciplina, a desonestidade académica é considerada fraude, com todas as consequências legais que daí advêm. Qualquer fraude terá como consequência imediata a anulação dos Projectos de todos os alunos envolvidos (incluindo os que possibilitaram a ocorrência). Qualquer suspeita de desonestidade académica será relatada aos órgãos superiores da escola para possível instauração de um processo disciplinar. Este poderá resultar em reprovação à disciplina, reprovação de ano ou mesmo suspensão temporária ou definitiva da ULHT.

8 Licenças

O enunciado e restante documentação são disponibilizados através da licença [CC BY-NC-SA 4.0].

Referências

- [1] Repositório do mini projecto de introdução à computação. Universidade Lusofona de Humanidades e Tecnologias. [Online]. Available: https://github.com/VideojogosLusofona/ic_2019_trabalho1
- [2] Github. GitHub. [Online]. Available: <https://github.com/>

- [3] Pandora. Universidade Lusofona de Humanidades e Tecnologias. [Online]. Available: <https://ikaika.ulusofona.pt>
- [4] GNU Project. GCC, the GNU Compiler Collection. Free Software Foundation. [Online]. Available: <http://www.gnu.org/software/gcc/gcc.html>
- [5] “Texto adaptado da disciplina de algoritmos e estruturas de dados,” Instituto Superior Técnico.
- [6] A. Pereira, *C e Algoritmos*, 2nd ed. Sílabo.
- [7] Moodle. Universidade Lusofona de Humanidades e Tecnologias. [Online]. Available: <https://secure.grupolusofona.pt/ulht/moodle/>