

## 21010 - Arquitetura de Computadores

### Enunciado

Considere o puzzle clássico “puzzle dos 15” ([https://pt.wikipedia.org/wiki/O\\_jogo\\_do\\_15](https://pt.wikipedia.org/wiki/O_jogo_do_15)) ou “fifteen puzzle”, em Inglês ([https://en.wikipedia.org/wiki/15\\_puzzle](https://en.wikipedia.org/wiki/15_puzzle)). O jogo consiste num tabuleiro de 4x4 posições, em que há 15 peças, numeradas de 1 a 15, e que devem ser colocadas por ordem, deixando um espaço livre no último quadrado (canto inferior direito), tal como se apresenta na figura:



As peças encontram-se encaixadas uma nas outras e cada peça pode apenas mover-se na vertical ou horizontal para o espaço livre.

Partindo da posição acima, pode-se demonstrar que existem  $(4 \times 4)!/2$  configurações possíveis de diferentes de disposição das peças.

Um puzzle deste tipo é considerado resolúvel se for possível, a partir da configuração inicial dos números no tabuleiro, atingir a posição final que é descrita acima.

A seguinte figura representa um puzzle resolúvel:



É conhecido um procedimento para se determinar se uma determinada configuração de números no tabuleiro constitui um puzzle resolúvel ou não. Este procedimento consiste em dois passos distintos:

- 1 Calcular o número total de inversões de cada uma das 16 posições do tabuleiro desde a posição inicial até à posição final;
- 2 Calcular a distância de Manhattan da posição inicial do espaço em relação à sua posição final.

Se a soma destes dois valores for par então estamos perante um puzzle resolúvel.

A distância de Manhattan ([https://en.wikipedia.org/wiki/Taxicab\\_distance](https://en.wikipedia.org/wiki/Taxicab_distance)) entre dois pontos num plano é calculada contando 1 por cada posição na vertical que tem que ser percorrida para estarem na mesma linha, a que se soma 1 por cada posição na horizontal para que ambos os pontos estejam na mesma coluna. Para o puzzle de exemplo a distância de Manhattan do espaço vazio até ao canto inferior direito é 4, 2 linhas e 2 colunas de distância.

O número total de inversões é calculado considerando o tabuleiro com uma ordenação linear dos seus elementos, e verificando para cada elemento quantos números aparecem em posições posteriores que são menores que o número no elemento que se está a considerar. Para o puzzle de exemplo temos na segunda linha da tabela as inversões por elemento, correspondendo a primeira linha aos elementos por ordem linear:

5	1	4	8	2	16	7	12	10	11	13	14	6	9	15	3
4	0	2	4	0	10	2	5	3	3	3	3	1	1	1	0

O número total de inversões é neste caso 42. Somando 42 com 4 relativo à distância de Manhattan do espaço em branco temos 46, que é par, o que significa que o puzzle apresentado é resolúvel. Note-se que se usa 16 (ou seja 4x4) para representar o espaço em branco.

a) [1] Desenvolva um programa em Assembly do P3 que para um determinado puzzle calcule a distância de Manhattan do espaço vazio até ao canto inferior direito.

O programa deve ter como entrada um vetor inteiro de 16 posições, que representa um determinado puzzle, apresentados em memória seguidos, a partir da posição 8000h. A saída do programa, ou seja a distância de Manhattan pedida, deve ser apresentada no registo R1. Caso haja uma situação de erro deve ser devolvido em R1 o valor FFFFh.

b) [1] Desenvolva um programa em Assembly do P3 que recebe como entrada um determinado puzzle, no mesmo formato da alínea anterior, e retorne no registo R2 o número total de inversões de cada uma das 16 posições do tabuleiro do puzzle dado até à posição final. Caso haja uma situação de erro deve ser devolvido em R2 o valor FFFFh.

c) [1] Desenvolva um programa em Assembly do P3 que receba um puzzle no formato descrito nas alíneas anteriores e determine se este é resolúvel, caso em que se deve apresentar o valor 1 no registo R3. Caso o puzzle não seja resolúvel o valor retornado no final do programa em R3 deverá ser 0. Caso tenha ocorrido algum erro, o valor retornado em R3 deverá ser FFFFh. São dados no ficheiro de exemplos (puzzles.txt) 5 casos de puzzles de 4x4 para os quais deve registar numa tabela a incluir no relatório o nome do puzzle, se o puzzle é resolúvel, o número de ciclos de relógio e o número de instruções.

d) [1] Sabendo, que o que foi dito a respeito do puzzle de dimensão 4x4 é também válido para um puzzle de dimensão NxN, desenvolva um programa em Assembly do P3 que verifique se um dado puzzle de NxN é resolúvel. A dimensão N será indicada na posição 8000h, seguida dos valores do puzzle, em formato idêntico ao utilizado nas alíneas anteriores, com as devidas adaptações correspondentes à dimensão do puzzle. O resultado deverá ser apresentado em R3, à semelhança da alínea anterior. São apresentados no ficheiro de exemplos 5 puzzles (puzzles.txt) para as dimensões 3, 4, 5, 7 e 10 (total de 25 casos de teste), para os quais deve registar numa tabela a incluir no relatório o nome do puzzle, a sua dimensão (N), se o puzzle é resolúvel, o número de ciclos de relógio e o número de instruções.

**BOM TRABALHO!**

## Avaliação

### Cotação:

A cotação encontra-se junto de cada uma das alíneas, entre [].

### CrITÉrios de Correção:

Funcionalidade: 50%

Simplicidade e Modularidade: 10%

Eficiência (serão contabilizados o número de instruções e ciclos de relógio): 10%

Apresentação do código (indentação e comentários): 20%

Relatório (Legibilidade e Justificação dos Resultados e das Opções): 10%

### Descontos:

Trabalhos entregues que não estejam em conformidade com as regras de entrega do e-fólio B: até 10%

Código sem comentários, ou apenas com comentários a reflectir o significado da instrução (exemplo MOV R1,R2 ;mover o conteúdo de R2 para R1) : até 50%

Deteção de fraude (total ou parcial): 100%

Trabalhos entregues após a data limite (o recurso de entrega estará aberto até ao início da manhã do dia seguinte, no entanto entregas após a data/hora indicada, têm desconto na nota) : 10%

## Regras para entrega do e-fólio B:

### Forma de entrega:

Um ficheiro zipado com o nome correspondente ao número de aluno.

O ficheiro zipado deverá por sua vez conter um ficheiro de Assembly do P3 por alínea, cujo nome será o número de aluno mais a letra da alínea ex: 999999a.as.

Deverá ainda ser entregue um pequeno relatório em formato pdf de até 5 páginas A4, com todos os cálculos e todas as opções tomadas na construção dos programas.

Regras relativas à escrita dos programas de Assembly:

Todos os programas devem acabar com a seguinte instrução:

FIM:           JMP FIM                   ;Fim do programa

Não são aceites entregas fora da plataforma Moodle.