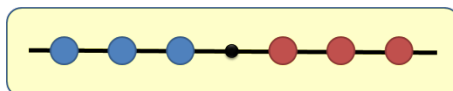


Trabalho de Programação 3
Processador Intel (80x86)

1. Descrição Geral

Neste trabalho, você deverá desenvolver um programa capaz de permitir jogar (executar) o desafio “frog hop”. Na realidade não se trata de um jogo, mas de um desafio individual.



Objetivo do desafio: trocar as peças azuis e vermelhas de lugar, ou seja, mover todas as peças azuis para a direita e todas as peças vermelhas para a esquerda, de tal forma que fique vazia apenas a posição central do tabuleiro.

Regras de movimentação de peças no tabuleiro (Procedimentos para Jogar)

1. As peças podem ser movidas para a posição vazias ao seu lado. Esse movimento só pode ocorrer em direção ao seu destino, do outro lado do tabuleiro;
 - a. Peças azuis só podem ser movidas para direita;
 - b. Peças vermelhas só podem ser movidas para a esquerda;
2. As peças podem “saltar” sobre uma peça de cor diferente, desde que seja na direção do destino e que haja uma posição vazia logo em seguida;
3. Não é permitido recuar peças (mover em direção ao seu lado de origem), nem movendo nem saltando;
4. Ao movimentar as peças, é permitido repetir as cores. Ou seja, não é necessário alternar as cores das peças movimentadas;

Na figura 1 está representada a situação (2): movimentação através de “salto” de uma peça azul em direção ao seu destino.

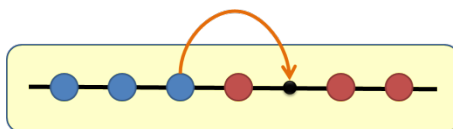


Figura 1. Exemplo de movimentação por salto.

Seu programa deverá ser desenvolvido em linguagem simbólica de montagem do processador 8086 da Intel e executado no ambiente DosBox. Para montagem de seu código fonte será usado o montador MASM 6.11.

2. Especificação do Programa

Na figura 2 estão representadas as ações que seu programa deve realizar.

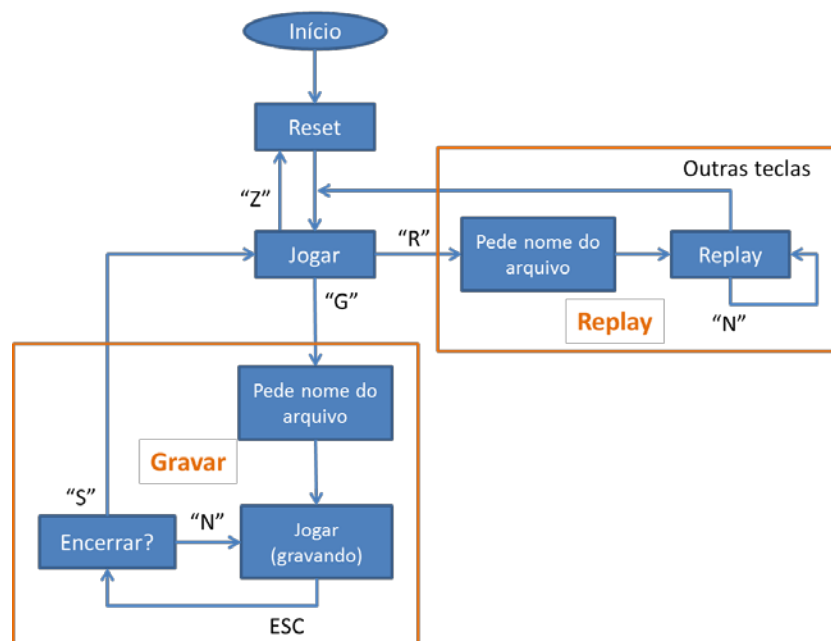


Figura 2. Fluxograma do programa a ser implementado.

Além do estado “Reset”, onde seu programa será inicializado e as peças serão colocadas nas suas posições originais, seu programa pode estar em um dos seguintes estados:

- jogando (conforme estado “Jogar” na figura 2);
- repetindo jogadas de um arquivo (conforme indicado por “Replay” na figura 2);
- gravando um arquivo com jogadas (conforme indicado por “Gravar” na figura 2).

Ao iniciar o programa, passando pelo “Reset”, o programa deve colocar na tela a configuração inicial do tabuleiro, conforme figura 3.

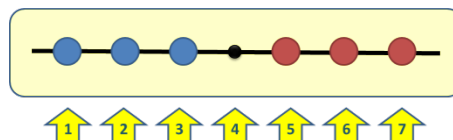


Figura 3. Configuração inicial do tabuleiro.

Jogando (tentando realizar o desafio)

Então, o programa entra no modo “Jogar”. Nesse modo, o programa deve aceitar as teclas dos dígitos “1” até “7” para indicar a posição da peça a ser movimentada. A partir desse estado também pode-se pedir para reinicializar o jogo (com a tecla “Z”, maiúscula ou minúscula), entrar no modo de reprodução das jogadas registradas em um arquivo (tecla “R”, maiúscula ou minúscula) ou entrar no modo de gravação de arquivo com o registro das jogadas realizadas (tecla “G”, maiúscula ou minúscula).

Por exemplo, se for digitado o número “3”, a peça correspondente deve ser movida. Nesse caso, a peça azul será movida para sua posição mais à direita (posição número “4”).

Além dos movimentos válidos (movimento para a posição adjacente ou salto sobre uma peça de cor diferente, conforme descrito anteriormente), durante o jogo podem ocorrer diversas situações, que deverão ser informadas ao usuário através de mensagens.

1. “*Movimento inválido*”, como é o caso de peças que não podem ser movidas (na figura 3, é o caso de “1”, “2”, “6” e “7”) ou quando a posição escolhida não tem qualquer peça (na figura acima, é o caso de “4”);
2. “*Você perdeu*”, quando não há mais possibilidade de movimentação e o objetivo do jogo não foi atingido;

3. “Você venceu”, quando todas as peças vermelhas estiverem à esquerda e todas as peças azuis estiverem à direita do tabuleiro, ou seja, houve a troca de lados das peças e a posição “4” ficou vazia.

Qualquer outra tecla que for pressionada e não responder a uma das operações do programa deve ser ignorada.

Repetindo jogadas de um arquivo: visualização de um jogo registrado

Neste modo de operação, o programa torna-se um visualizador de um jogo já realizado e registrado em arquivo.

O programa entra nesse estado a partir do estado “Jogar”, ao ser acionada a tecla “R” (maiúscula ou minúscula). Veja a figura 2.

A visualização deve ser passo-a-passo. O programa deve avançar um passo (um movimento) sempre que for acionada a tecla “N” (maiúscula ou minúscula).

O programa deve informar quando chegar ao final dos movimentos registrados no arquivo. Quando chegar ao final do arquivo, o programa deve passar a ignorar a tecla “N”.

O programa deve informar a tecla (número) lida do arquivo e o resultado desta ação no tabuleiro. Também deve ser informado quando o movimento for inválido.

Caso o usuário digite qualquer outra tecla, o programa deve retornar para o estado “jogando”.

Gravando um arquivo: registro da sequência das teclas (números) acionadas

Esta opção permite ao usuário gravar em um arquivo a sua tentativa de resolução deste desafio para poder visualizá-la em outro momento.

O programa entra nesse estado a partir do estado “Jogar”, ao ser acionada a tecla “G” (maiúscula ou minúscula). Veja a figura 2.

Ao entrar nesse estado, o programa deve solicitar o nome do arquivo a ser gravado com os movimentos do jogo.

O comportamento do programa neste estado é o mesmo que ocorre no estado “Jogar”. A única diferença é que, nesse estado, todas as teclas digitadas serão gravadas no arquivo.

Para encerrar a gravação do arquivo, o usuário deve digitar a tecla “ESC”. Então, o programa deve pedir a confirmação do encerramento da gravação do arquivo. Se for confirmado o encerramento da gravação, o programa deve fechar o arquivo e retornar para o estado “Jogar”.

Formato do arquivo

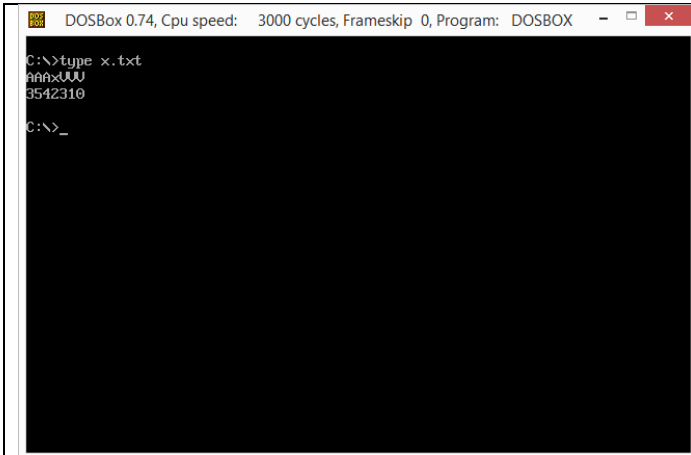
O arquivo com os movimentos realizados (que será usado na repetição das jogadas ou na gravação da sequência de teclas) deve ter o seguinte formato:

- Primeira linha: situação inicial das peças no tabuleiro. Cada caractere (coluna) corresponde a uma posição.
- Segunda linha: teclas digitadas correspondentes aos movimentos.

A primeira linha terá, sempre, 7 caracteres. O primeiro caractere corresponde à posição 1; o segundo caractere corresponde à posição 2, e assim por diante até o sétimo caractere. Deve-se usar a letra “A” para as posições que tiverem peças azuis; a letra “V” para as posições que tiverem peças vermelhas; e “x” (minúsculo) para a posição que estiver vazia (sem peças).

A segunda linha terá tantos caracteres quantos forem os movimentos salvos. Cada caractere corresponde ao movimento desejado. Ao final dos caracteres que representam os movimentos, deve haver um caractere “0”.

Exemplo de arquivo:

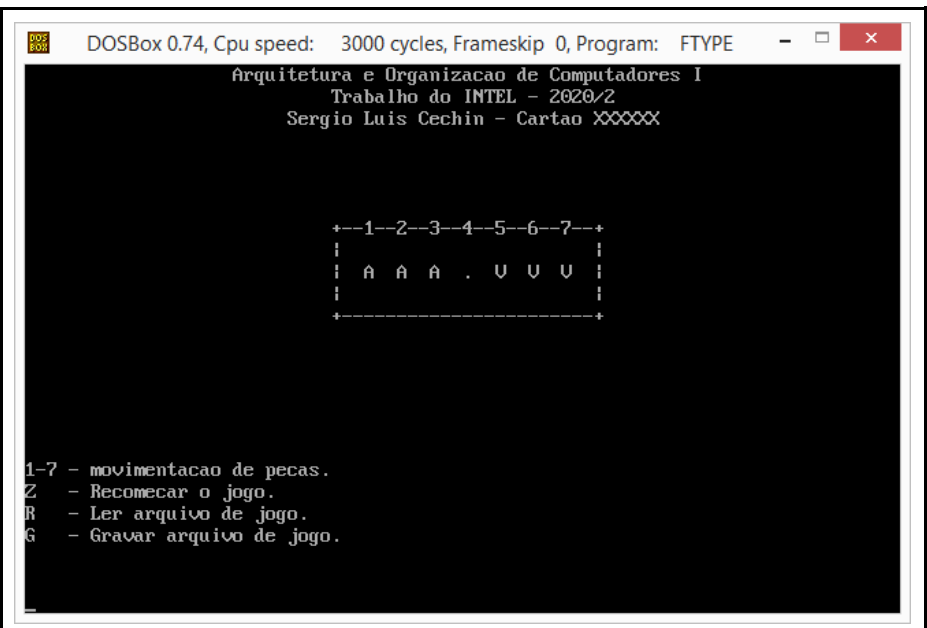


Ao ser processado (repetida a execução), esse arquivo vai gerar a seguinte sequência de estados no tabuleiro:

| Início | A A A x V V V |
|--------|---------------|
| 3 | A A x A V V V |
| 5 | A A V A x V V |
| 4 | A A V x A V V |
| 2 | A x V A A V V |
| 3 | A V x A A V V |
| 1 | X V A A A V V |
| 0 | |

3. Print das telas a serem construídas

Durante o estado de “Jogar”



Pede o nome do arquivo a ser lido ou escrito.



DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: FTYPE

Arquitetura e Organizacao de Computadores I
Trabalho do INTEL - 2020/2
Sergio Luis Cechin - Cartao XXXXXX

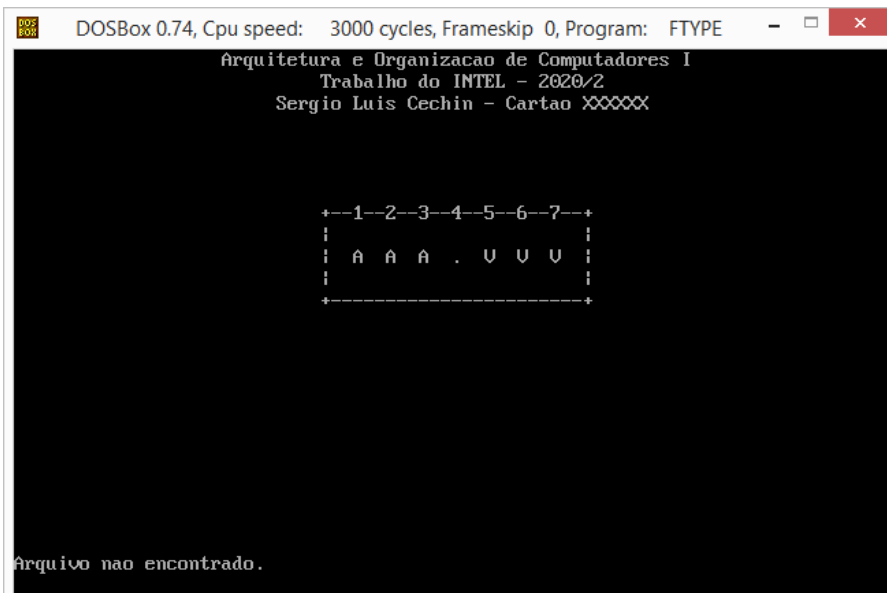
```

+--1--2--3--4--5--6--7--+
| A A A . U U U |
|                         |
+-----+

```

Entrar o nome do arquivo:

Quando o arquivo não for encontrado.



DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: FTYPE

Arquitetura e Organizacao de Computadores I
Trabalho do INTEL - 2020/2
Sergio Luis Cechin - Cartao XXXXXX

```

+--1--2--3--4--5--6--7--+
| A A A . U U U |
|                         |
+-----+

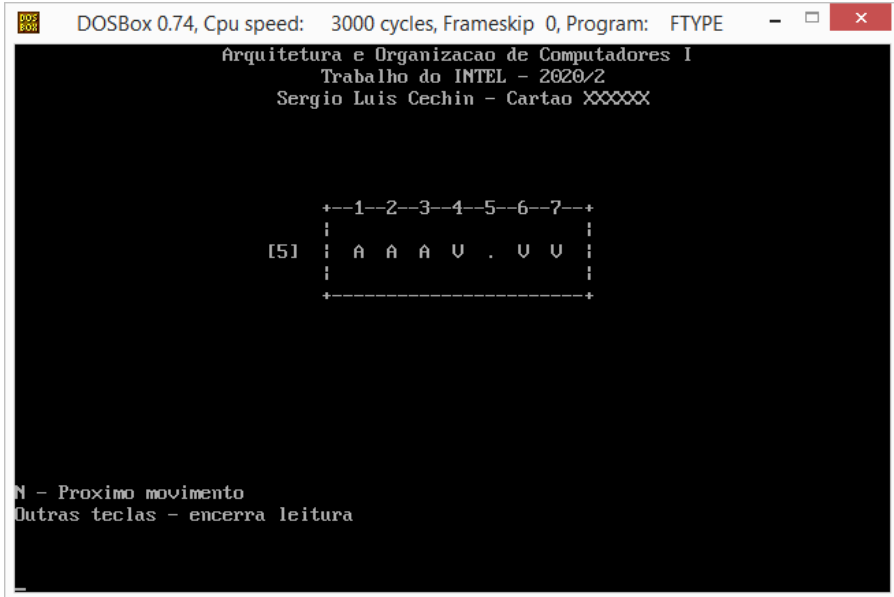
```

Arquivo nao encontrado.

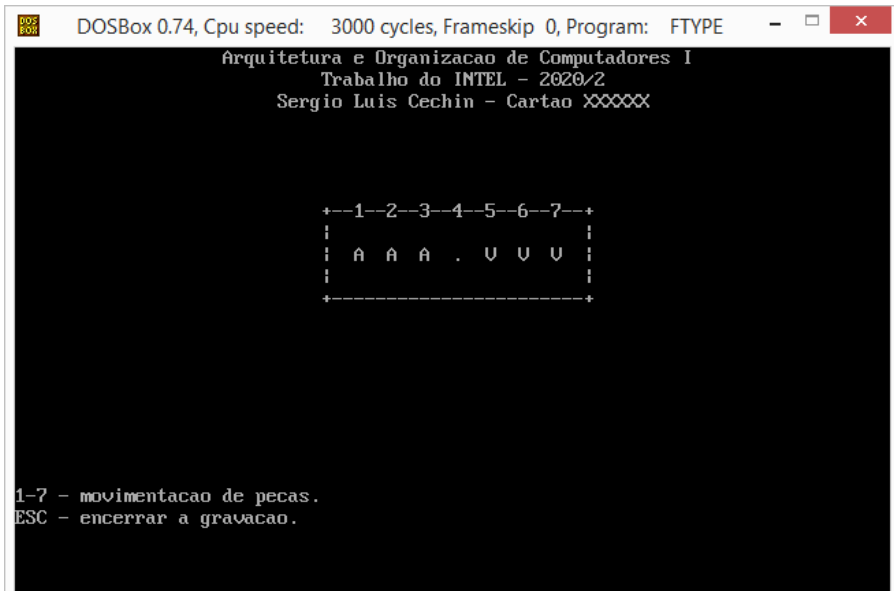
Quando não há mais movimentos possíveis (você perdeu).

Quando foi atingido o objetivo (você ganhou).

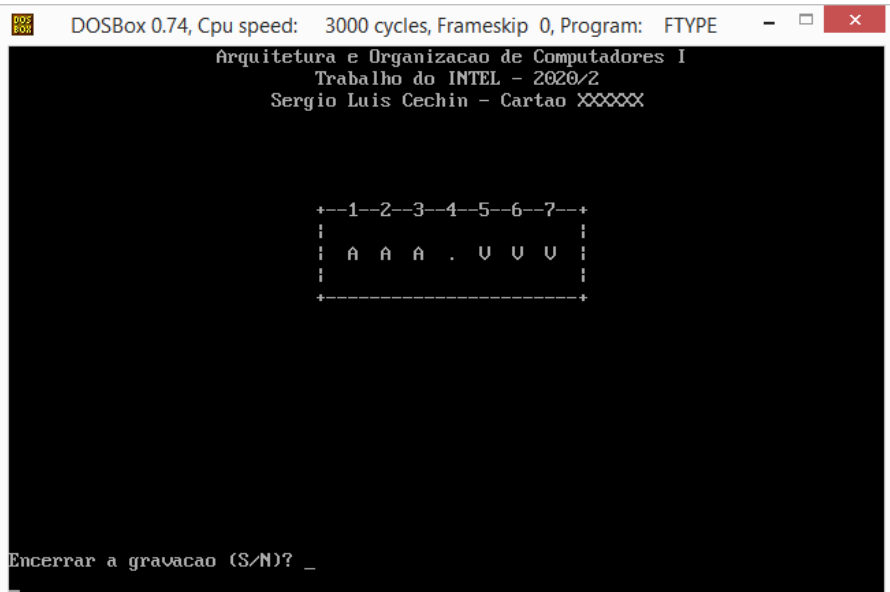
Aguarda comando para a leitura e apresentação do próximo movimento. Qualquer outra tecla encerra a leitura do arquivo.



Permite a realização de movimento das peças e grava esses movimentos no arquivo. Se for digitado ESC, vai perguntar se deve encerrar a gravação no arquivo.



Pergunta se deve parar a gravação em arquivo.
A resposta deve ser “S” ou “N”
(maiúsculas ou minúsculas).



DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: FTYPE

```
Arquitetura e Organizacao de Computadores I
Trabalho do INTEL - 2020/2
Sergio Luis Cechin - Cartao XXXXXX
```

```

+--1--2--3--4--5--6--7--+
| A A A . U U U |
+-----+

```

Encerrar a gravacao (S/N)? _

4. Entregáveis: o que deve ser entregue?

Deverá ser entregue, via Moodle da disciplina, o arquivo fonte com a solução do problema apresentado, escrito *na linguagem simbólica de montagem* dos processadores 80X86 da Intel (arquivo .ASM). Além disso, esse programa fonte deverá conter comentários descritivos da implementação.

Para a correção, os programas serão montados usando o montador **MASM 6.11** no ambiente **DosBox 0.74** e executados com diferentes arquivos de dados de entrada. A nota final do trabalho será proporcional às funcionalidades que forem atendidas pelo programa.

O trabalho deverá ser entregue até a data prevista, conforme programado no MOODLE. **Não será aceita a entrega de trabalhos após a data estabelecida.**

5. Observações

Recomenda-se a troca de ideias entre os alunos. Entretanto, a identificação de cópias de trabalhos acarretará na aplicação do Código Disciplinar Discente e a tomada das medidas cabíveis para essa situação (**tanto o trabalho original quanto os copiados receberão nota zero**).

O professor da disciplina reserva-se o direito, caso necessário, de solicitar uma demonstração do programa, onde o aluno será arguido sobre o trabalho como um todo. Nesse caso, a nota final do trabalho levará em consideração o resultado da demonstração.