UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL INSTITUTO DE INFORMÁTICA - DEPARTAMENTO DE INFORMÁTICA APLICADA

INF01108 - Arquitetura e Organização de Computadores I – semestre 2012/2 - Turmas A, B e C

Trabalho Prático 3 - Processador INTEL - Jogo da Caixa Preta

Escrever, para o processador Intel, um programa que implemente o jogo da Caixa Preta (http://en.wikipedia.org/wiki/Black_Box_(game)) para o ambiente DOS. O jogador deve descobrir as posições de 4 átomos através da emissão de raios e da observação das respostas obtidas.

Definição

O programa deve começar exibindo a tela inicial, conforme o desenho mostrado na Figura 1. Note que, na figura, as posições que contêm pontos ('.') devem ficar inicialmente em branco até que sejam preenchidas pelo jogador (os caracteres em azul servem apenas para indicar os limites e a numeração de linhas e colunas da tela que deve ser usada para calcular as posições onde escrever ©). Cada posição da projeção horizontal da caixa preta, exibida na tela como uma matriz 8x8, será identificada por um par de coordenadas, que indica os números de linha e coluna (de 0 a 7) daquela posição na matriz.

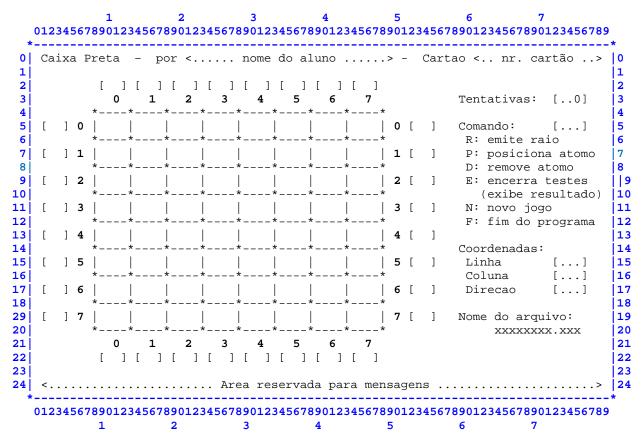


Figura 1. Tela inicial do programa

Após exibir a tela inicial, o programa deverá pedir ao usuário o nome do arquivo a ser lido de disco, usando para isto a área de mensagens na parte inferior da tela para exibir o prompt (mensagem com pedido de informações, por exemplo: 'Informe o nome do arquivo: _'). Note que, ao contrário do Cesar, onde o cursor do prompt era controlado pelo programa, o sistema operacional DOS administra ele mesmo um cursor 'piscante' durante a digitação do texto. Ao digitar o nome do arquivo, o usuário deverá ter a possibilidade de corrigir os caracteres digitados, usando a tecla backspace. O arquivo estará no formato texto do MS-DOS e gravado no mesmo diretório em que o programa estiver sendo executado. O conteúdo do arquivo será formado por 8 linhas, cada uma delas contendo 8 caracteres de dados (um '0' indica uma casa vazia e um '1' indica uma casa contendo um átomo), seguidos por um par de caracteres cr. (carriage return, código ASCII 13) e lf (line feed, código ASCII 10), que indicam fim de linha em arquivos neste formato (texto do MS-DOS). Não é necessário fazer nenhuma verificação dos dados lidos do arquivo, pois é garantido que eles estarão no

formato aqui descrito. A Figura 2 mostra um exemplo de conteúdo do arquivo a ser lido de disco. Note que ele consiste em uma sequência de 80 caracteres codificados em ASCII, cada caractere ocupando 1 byte, incluindo os caracteres indicadores de nova linha (um par cR, LF por linha) mostrados à direita das linhas na figura. Note que não existe separação física entre linhas consecutivas, nem espaços entre dois caracteres consecutivos dentro do arquivo. Os espaços deixados na figura são apenas para tornar a figura mais legível, pois as siglas cR e LF representam apenas 1 caractere cada uma. Os códigos ASCII dos caracteres 0, 1, cR e LF são, respectivamente, 48 (30_{16}), 49 (31_{16}), 13 ($0D_{16}$) e 10 ($0A_{16}$). Portanto, analisando o arquivo mostrado na figura, conclui-se que existem quatro átomos dentro da caixa preta, nas posições com coordenadas (2,1), (2,6), (5,3) e (7,6):

```
0 0 0 0 0 0 0 0 0 0 CR LF
0 1 0 0 0 0 0 0 1 0 CR LF
0 0 0 0 0 0 0 0 0 CR LF
0 0 0 0 0 0 0 0 0 CR LF
0 0 0 0 1 0 0 0 0 CR LF
0 0 0 0 1 0 0 0 CR LF
0 0 0 0 0 0 0 CR LF
0 0 0 0 0 0 CR LF
```

Figura 2. Exemplo de dados no arquivo de entrada, para uma caixa preta com 4 átomos

Durante o decorrer do jogo, o jogador escolhe a ação que deseja executar digitando as letras (maiúsculas ou minúsculas) correspondentes a cada 'comando':

• R: injeta um raio na caixa preta. Para este comando, devem ser informadas as coordenadas (linha e coluna), bem como o sentido de entrada do raio (E, D, C ou B, conforme o raio entre pela esquerda, pela direita, por cima ou por baixo – considerando a visão de quem está observando a tela, respectivamente). O resultado da injeção do raio deve ser informado na área de mensagens e deve ser marcado nas bordas correspondentes da caixa, nos espaços entre colchetes ('[' e ']'), de acordo com a seguinte convenção: 'A' para absorção, 'R' para reflexão ou então um número 'i' a ser colocado nas posições de entrada e de saída de um raio, quando este não for refletido nem absorvido. A Figura 3 mostra as informações exibidas na tela após terem sido realizados diversos testes (injeção de 23 raios neste exemplo). A cada teste (injeção de um raio), o contador de tentativas exibido à direita na linha 4 da tela deve ser incrementado e atualizado.

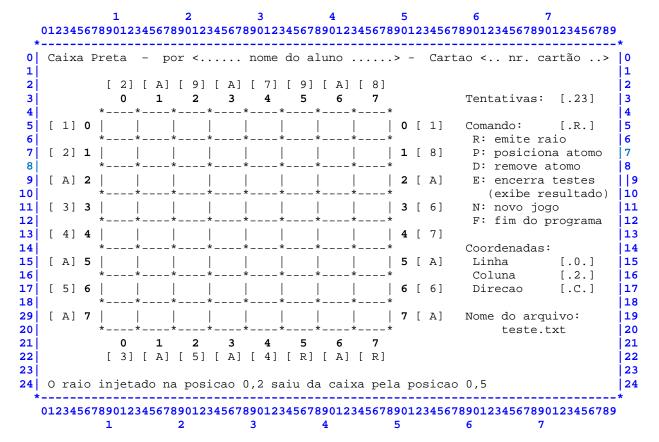


Figura 3. Tela do programa após a realização de 23 testes

Observação: para fins do exemplo mostrado na Figura 3, foram realizados todos os testes possíveis para aquela configuração dos 4 átomos dentro da caixa. Durante um jogo normal, o objetivo é deduzir as posições dos átomos (e marcá-las usando o comando 'P') injetando na caixa o menor número possível de raios.

- P: coloca um átomo na coordenada especificada.
- D: remove o átomo da coordenada especificada.
- E: encerra a fase de testes, verificando se os átomos estão corretamente posicionados ou não.
- F: termina o programa, retornando ao DOS.
- N: novo jogo (limpar a tela e as variáveis usadas pelo programa, ler novo arquivo de disco para reinicializar a matriz que representa a caixa preta e iniciar um novo jogo)

Os comandos 'P' e 'D' permitem a colocação e a retirada na/da caixa de ícones que representam átomos, para auxiliar visualmente o jogador nos seus testes. Note que a verificação se o posicionamento está correto ou não, ou seja, corresponde realmente a um átomo no arquivo de entrada, somente deve ocorrer quando o jogador encerra os testes como comando E. A figura 4 mostra o posicionamento de 4 esferas, realizado pelo jogador em função dos testes realizados. O posicionamento (ou remoção) de um átomo não conta como sendo uma tentativa.

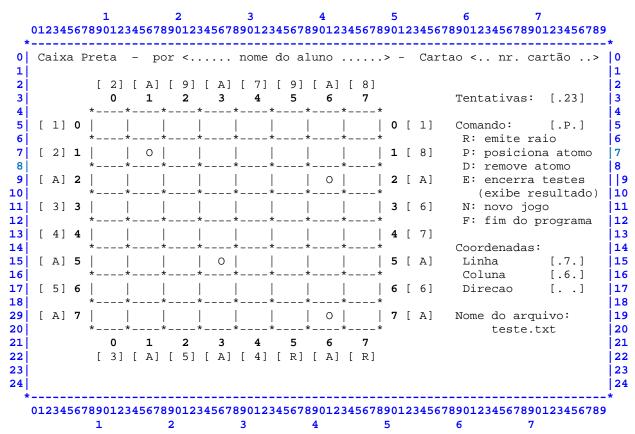


Figura 4. Tela do programa após o posicionamento de 4 átomos usando o comando 'P'

Quando o jogador estiver satisfeito com os testes realizados e deduzir o posicionamento dos átomos, ele deve encerrar os testes como o comando "E" (ou "e"). O programa deve então verificar se o posicionamento está correto (ou não) e informar a pontuação final ao jogador. Cada raio emitido vale 1 ponto, e cada esfera posicionada incorretamente vale 5 pontos (o objetivo é resolver o jogo com o menor número de pontos). A figura 5 mostra o resultado do posicionamento da figura 4, onde o jogador posicionou 1 átomo incorretamente.

Para indicar o resultado, utilize a seguinte convenção: 'OO' indica átomo corretamente posicionado, 'XO' indica átomo posicionado incorretamente pelo jogador, e 'OX' significa a posição de um átomo não posicionado pelo jogador.

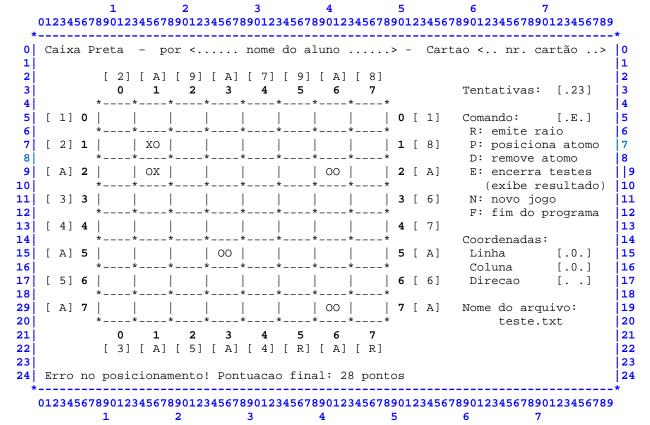


Figura 5. Tela do programa após verificação do posicionamento

Especificações para a implementação do jogo

A escolha das mensagens adequadas para cada situação (erro na entrada, jogada inválida, tentativa de encerramento dos testes sem posicionar átomos, etc) fica a critério de cada programador.

O programa deverá executar as seguintes tarefas, na sequência indicada:

- 1. Inicializar todas as variáveis e estruturas de dados necessárias ao controle do jogo.
- 2. Exibir a tela inicial (formato indicado na Figura 1), com todas as informações variáveis (comando, tentativas, coordenadas, etc.) em branco.
- 3. Exibir mensagem solicitando ao usuário um nome de arquivo e ler o nome de arquivo, criando na memória o string "ASCIIZ" necessário para o processamento do arquivo. Na leitura do nome do arquivo, aceitar e processar corretamente o caractere Backspace (código ASCII 8) e permitir nomes com no máximo 12 caracteres (padrão DOS, com 8 letras para nome, um ponto (.) e 3 letras para sufixo), ignorando caracteres adicionais digitados. Somente considerar encerrada a digitação do nome do arquivo quando receber o caractere Carriage Return (código ASCII 13), gerado pela tecla [ENTER].
- 4. Abrir o arquivo e testar o código de retorno. Se houver erro, exibir uma mensagem adequada, aguardar que o usuário tecle [ENTER] e voltar à etapa 3.
- 5. Ler o arquivo, carregando os valores na memória. Não esquecer de fechar o arquivo quando encontrar o fim (0 caracteres lidos em uma tentativa de ler um caractere do arquivo).
- 6. Ler um comando do usuário, no campo adequado. Um comando será constituído de uma única letra (maiúscula ou minúscula). Deve ser permitido o uso de backspace. O comando é considerando encerrado quando o usuário digitar a tecla [ENTER].
- 7. Processar o comando. Se for necessária entrada extra de dados (linha, coluna, etc), realizar esta entrada no campo adequado, com o mesmo tipo de processamento do item 6 (uso de backspace e [ENTER]).
- 8. Se o comando for "N" (ou "n"), voltar ao passo 1.
- 9. Se o comando for "F" (ou "f"), encerrar o programa e retornar ao DOS.

A interface com o usuário: melhorias serão recompensadas

Aqueles que desejarem melhorar a nota, com um bônus de até 10% (oferta válida somente para trabalhos corretos), têm as seguintes opções (entre outras) para melhorar a interface:

- Uso de cores (sugestão: primeiro desenvolva todo o jogo em P&B, e depois acrescente a formatação de cores antes do código de inicialização da tela).
- Uso do mouse para selecionar posições na matriz
- Uso das teclas direcionais para selecionar posições na matriz
- Implementação de um sistema de dicas de jogo (tecla de help, ou ajuda).

Entrega

O trabalho deverá ser feito de forma individual e entregue via Moodle, em **somente UM** arquivo com o programa fonte comentado (.asm). Para evitar problemas de conflito entre seu navegador e o Moodle (inserção de código HTML), compacte este arquivo fonte. Para nomear o arquivo, utilize a inicial de seu primeiro nome, seguida de seu número de cartão de identificação da UFRGS (seis dígitos). Por exemplo, o aluno **João Silva**, número de cartão **654321** deverá nomear seus arquivos **J654321**.asm e JoaoSilva.zip (ou JoaoSilva.rar). **Importante**: não utilize zeros não significativos do seu número de cartão, pois isso causaria problemas com nomes DOS - se o nome do arquivo fosse J00654321.asm, isto excederia o tamanho máximo de um nome DOS!

Dia e hora limites para entrega: 18/12/2012 às 23h55, via Moodle