

UNIFEI	Universidade Federal de Itajubá Instituto de Engenharia de Sistemas e Tecnologias da Informação – IESTI
12º Laboratório	Prof. Enzo Seraphim
Exercício 1	Simulador do Jogo Banco Imobiliário

1) Abra o code::blocks e crie novo arquivo usando o menu File| New | Empty File

2) Digite o código abaixo para simular computacionalmente o jogo Banco Imobiliário:

```

1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<time.h>
4  typedef enum{NENHUM, EMPRESA, SORTEREVES, PRISAO, TERRENO} enum_tipo;
5  typedef enum{NINGUEM, JOGADOR1, JOGADOR2} enum_dono;
6  typedef struct {
7      enum_tipo tipo;
8      int preco;
9      enum_dono dono;
10 }struct_casa;
11 struct_casa tab[30];
12 void inicializa(){}//end inicializa
13 void geratabuleiro(){}//end geratabuleiro
14 void mostra(int j1, int j2){}//mostra
15 int main(int argc, char* argv[]){
16     int i, dado, pos1=0, pos2=0, saldo1=100, saldo2=100;
17     inicializa();
18     geratabuleiro();
19     while(saldo1>0 && saldo2 >0){
20         dado=(rand()%3)+1;
21         pos1=(pos1 + dado) % 30;
22         dado=(rand()%3)+1;
23         pos2=(pos2 + dado) % 30;
24         mostra(pos1, pos2);
25         printf("Jogador1=%d Jogador2=%d\n", saldo1, saldo2);
26         getchar();
27     }
28     return 0;
29 }
```

O Banco Imobiliário é um jogo onde os jogadores andam com peões em um tabuleiro que é composto por casa.

Cada casa do tabuleiro:

- tem um dos seguintes tipos: NENHUM, Empresa, Terreno, Sorte ou Revés e Prisão. Para limitar o tipo da casa do tabuleiro foi criado o enumerado enum_tipo (linha 4).
- tem um preço que custa o imóvel, quando for um Terreno ou uma Empresa.
- tem um dos seguintes donos, quando for um Terreno ou uma Empresa: Jogado1, Jogador2 ou Ninguém (linha 5). Para limitar o dono da casa do tabuleiro foi criado o enumerado enum_dono.

Portanto, toda casa tem um tipo que é limitado pelo enum_tipo (linha 7), um preço (linha 8) e um dono que é limitado pelo enum_dono (linha 9), sendo estruturado pelo struct_casa (linha 10). Para representar o tabuleiro de 30 casas foi definido o vetor global chamado tab (linha 11).

Cada partida começa com um tabuleiro NENHUM, sem preço e sem ninguém como dono da casa que será definido na função inicializa (linha 12).

Cada partida da simulação terá um tabuleiro formado randomicamente entre os tipos das casas que será definido pela função geratabuleiro (linha 13).

Finalmente, a função mostra exibe as 3 linhas do tabuleiro na tela com a peça do jogador 1 e a do jogador 2 (linha 14).

O programa que simula jogadas no banco imobiliário enquanto houver saldo para jogador 1 e saldo para jogador 2 (linha 19) entre os jogadores 1 e 2, sendo que cada jogador começa na posição zero do vetor (linha 16).

A próxima posição do jogador será o incremento por um valor sorteado pelo dado que podem ser 1, 2 ou 3 (linha 20 e linha 22). Toda vez que uma posição do Jogador 1 ou Jogador 2 chegar ou ultrapassar a posição 30, deve-se continuar o incremento do valor restante na posição 0 (linha 21 e linha 23)

1) Use os menus Arquivo | Salvar (Ctrl+s), salvando o algoritmo em uma pasta desejada com o nome lab10-1.

2) Coloque o código dentro das chaves da função inicializa (linha 12) deve inicializar como NENHUM o tabuleiro. Para cada posição tabuleiro (tab), deve-se atribuir para: o elemento tipo deve ser NENHUM; o preço deve ser zero; e o dono deve ser NINGUEM. Use um laço de repetição for variando de 0 até 29.

3) Coloque o código dentro das chaves da função geratabuleiro (linha 13) a geração randômica das empresas. Essa função deverá sortear 10 casas em qualquer posição do vetor para receber empresa, use um laço de repetição for variando de 0 até 10. Se a posição sorteada no tabuleiro (tab) estiver NENHUM: o tipo de casa deve mudar para EMPRESA; e o valor da casa deve ser 10 ou 20.

4) Adicione no código da função geratabuleiro a geração randômica do sorte/revés. A função deverá sortear 5 casas em qualquer posição do vetor com tipo NENHUM para receber sorte/revés, use um laço de repetição for variando de 0 até 5. Se a posição sorteada no tabuleiro (tab) estiver NENHUM: o tipo de casa deve mudar para SORTEREVES; e o valor da casa deve estar entre -9 e 9.

5) Adicione no código da função geratabuleiro a geração randômica da prisão. A função deverá sortear 2 casas em qualquer posição do vetor com tipo NENHUM para receber prisão, use um laço de repetição for variando de 0 até 2. Se a posição sorteada no tabuleiro (tab) estiver NENHUM: o tipo de casa deve mudar para PRISAO; e o valor da casa deve ser -9.

6) Adicione no código da função geratabuleiro os terrenos que ocuparão as posições restantes do vetor. Use um laço de repetição for variando de 0 até 30 verificando se o tipo da casa no tabuleiro (tab) está NENHUM: o tipo de casa deve mudar para TERRENO; e o valor da casa deve estar entre 0 e 10.

7) Coloque o código dentro das chaves da função mostra (linha 14) para imprimir a primeira linha do tabuleiro na tela. A função mostra não retorna valores e tem dois parâmetros que são a posição de cada jogador. Conforme imagem abaixo, a primeira linha consiste em mostrar o tipo de cada casa do tabuleiro (tab), sendo: EMPRESA apresente " EM "; TERRENO apresente " TE "; SORTEREVES apresente " ?? "; PRISAO apresente " ## ". Use um laço de repetição for variando de 0 até 30.

```
TE TE ## ?? EM EM TE ?? EM TE EM EM EM TE EM TE TE ?? TE TE EM TE ?? TE EM TE ## ?? EM TE
```


decremente seu saldo e incremente o saldo do adversário com preço da casa multiplicado pelo dado; próprio jogador é o dono não faça nada.

- **DEMAIS SITUAÇÕES:** o saldo do jogador é incrementado ou decrementado pelo preço da casa.