UNIFEI - Universidade Federal de Itajubá
Prof. Gabriel Cirac Mendes Souza (gabrielcirac@unifei.edu.br)

Projeto da Disciplina ECOP01/ECOP11

Você largou a engenharia e foi caçar Pokemons na China. Pokemons são criaturas que duelam umas com as outras e recebem um tipo de treinamento pessoal para isso. Você gostou tanto desse jogo que decidiu fazer uma versão em C para seus amigos.

Esse trabalho irá simular de maneira **simplista** o jogo do Pokemon. As instrução **não** precisam ser seguidas a risca, onde um resultado criativo e funcional é mais importante que um programa complexo. Desse modo, tentem fazer da maneira que achar correto e usem o máximo de recursos para facilitar a implementação.

Outra dica é ler sobre o jogo original e tentar implementar alguma funcionalidade que achar relevante e fácil.

1 – Basicamente todo Pokemon possui características que podem ser representadas por uma estrutura heterogênea, como um nome, um nível de força e um tipo (fogo, ar, terra, eletricidade).

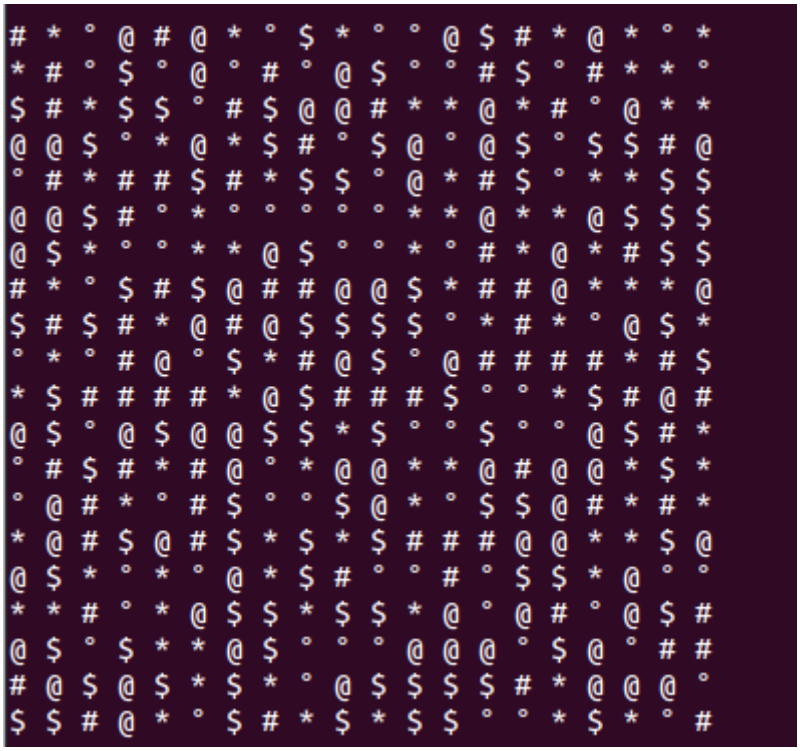
```
typedef struct pokemon {
    char nome[50];
    int level;
    int tipo;
} pokemon_t;
```

2 – Para poder jogar você deve cadastrar o seus Pokemons e os Pokemons dos mestres. Você pode considerar que coleciona apenas 5 animaizinhos e no mapa existem 5 mestres, cada um com uma quantidade de Pokemons definida por você. As figuras abaixo exemplificam o cadastro.

```
Entre com o nome do Pokemon 0
Pikachu
Entre com o level do Pokemon 0
65
Entre com o tipo -1-2-3-4-5 do Pokemon 0
5
Entre com o nome do Pokemon 1
Bulbacharm
Entre com o level do Pokemon 1
85
Entre com o tipo -1-2-3-4-5 do Pokemon 1
3
Nome: Pikachu
Level: 65
Eletrico
----------------
Nome: Bulbacharm
Level: 85
Ar
```

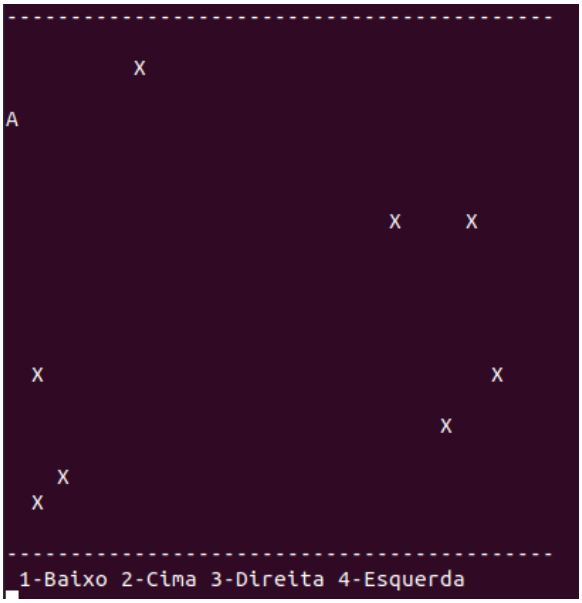
```
----------------
MESTRES
Entre com o nome do Pokemon 0
Zapdos
Entre com o level do Pokemon 0
95
Entre com o tipo -1-2-3-4-5 do Pokemon 0
5
Entre com o nome do Pokemon 1
Articuno
Entre com o level do Pokemon 1
99
Entre com o tipo -1-2-3-4-5 do Pokemon 1
4
Nome: Zapdos
Level: 95
Eletrico
----------------
Nome: Articuno
Level: 99
Terra
```

3 – Cada tipo de Pokemon é mais forte em determinado ambiente. Por exemplo, um Pokemon de pedra é mais forte na montanha, enquanto um Pokemon de água é mais forte em um lago. Desse modo, é possível utilizar um mapa auxiliar que represente diversos tipos de campos ao longo da trajetória, você deve sortear símbolos e associar cada um deles aos tipos que escolher. Por exemplo, na figura abaixo @ pode ser terra, enquanto # pode ser ar.



4 - Esse mapa não precisa aparecer, ele deve ficar escondido e o programa só deve fazer referência a ele quando ocorrer uma batalha. Quando você encontrar um mestre deve verificar o tipo de campo no mapa auxiliar, se o tipo favorecer seu Pokemon ele ganha pontos, se o tipo favorecer o Pokemon do treinador ele ganha pontos. Se não favorecer ninguém o que acontece?

5 – Os treinadores estão espalhados pelo mapa, as posições são sorteados e você deve decidir quantos mestres iniciam o jogo. O mestre pode ser marcado pela letra X e o seu avatar pode ser marcado pela letra A.



6 – Os mestres estão fixos em seus ginásios e o seu avatar deve ser capaz de se mover livremente no mapa. Para isso, é razoável pensar em uma função que permite ao seu jogador se movimentar livremente (robô desenhista?).

7 – Toda vez que o seu avatar chegar na posição de um ginásio você deve simular uma batalha, onde pode escolher 1 Pokemon ou 2, a seu critério. Observem as seguintes recomendações:

- O que determina a vitória é o nível do Pokemon escolhido.

```
if (mestre[mestreNum].level > jogador[poke].level){

    printf("Perdeu!!!!");
    return 0;

}
```

- O tipo de campo pode ser decisivo na hora da batalha.

```
if(piso[linha][coluna]==mestre[mestreNum].tipo){

    printf("Campo do mesmo tipo, MESTRE ganha mais 20 \n");
    mestre[mestreNum].level += 20;
    }
```

8 – Após a batalha o programa deve informar se você venceu ou perdeu a batalha. Na sequência seu avatar deve ser capaz de percorrer o mapa novamente.

9 – Você pode utilizar um saldo, toda vez que ganhar uma batalha ganha uma insígnia (Pedra, elétrica...) e toda vez que perder uma batalha deve perder um pouco de grana. O dinheiro inicial do seu avatar deve ser decidido por você.

10 – Um dos compoenetes principais do trabalho será a criatividade, onde o aluno deverá criar um programa único, original e na apresentação explicar a escolha. A única restrição é com relação ao número de função, que deve ser minimamente 4.

Dicas:

- Você pode manter uma lista dos seus Pokemons e outra com os Pokemons dos mestres;

```
pokemon_t lista[5];

pokemon_t mestre[8];
```

- Todo processo pode acontecer dentro de um laço While(), de acordo com uma condição de parada definida por você.

O trabalho deve ser **apresentado pela dupla** em um vídeo de no máximo 10 minutos, levando em consideração:

- Clareza e objetividade;
- Explicação do código;
- Execução do código.