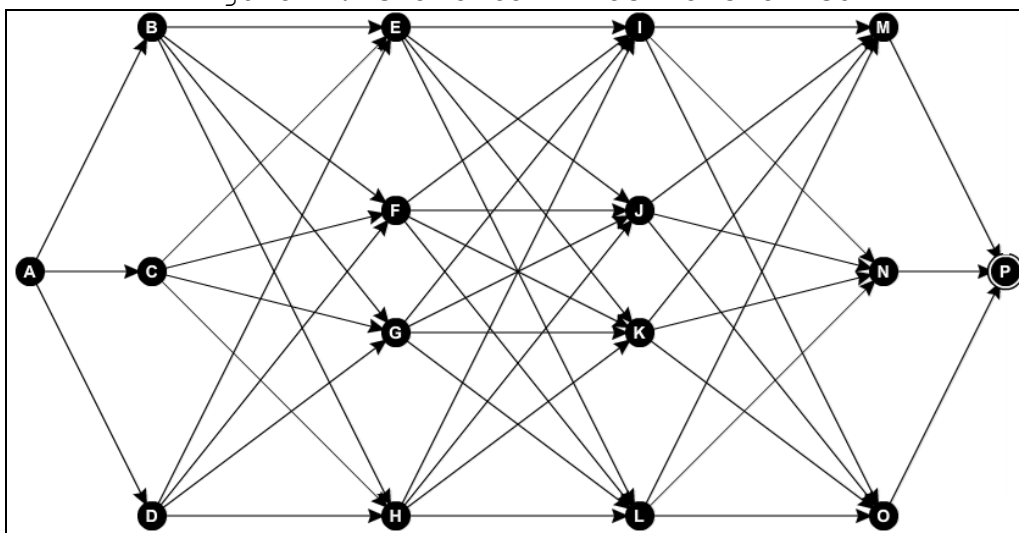




A grande febre, o Pokémon GO, é um jogo de realidade aumentada desenvolvido pela parceria Nintendo/Niantic. Qual a relação com a disciplina? Bem, você foi convidado a desenvolver um projeto que visa mostrar os caminhos mais curtos entre a sua localização e a do Pokémon. Suponha o grafo da Figura 1, onde estamos localizados no vértice A e o Pokémon no vértice P; o objetivo é encontrar o(s) caminho(s) mais curto(s) para que o jogador escolha o caminho mais rápido possível para capturar o Pokémon.

Figura 1: Grafo Caminhos Pokémon GO




Fonte: elaborado pelo autor.

O grafo pode ser representado através de uma matriz de adjacência, conforme apresentado na Figura 2.

Figura 2: Matriz de Adjacência

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
A	0	2	1	3	0	0	0	0	0	0	0	0	0	0	0	0
B	2	0	0	0	7	2	5	3	0	0	0	0	0	0	0	0
C	1	0	0	0	5	4	2	8	0	0	0	0	0	0	0	0
D	3	0	0	0	3	8	3	5	0	0	0	0	0	0	0	0
E	0	7	5	3	0	0	0	0	10	15	8	5	0	0	0	0
F	0	2	4	8	0	0	0	0	6	8	4	9	0	0	0	0
G	0	5	2	3	0	0	0	0	8	2	17	13	0	0	0	0
H	0	3	8	5	0	0	0	0	3	5	5	11	0	0	0	0
I	0	0	0	0	10	6	8	3	0	0	0	0	2	3	6	0
J	0	0	0	0	15	8	2	5	0	0	0	0	1	6	8	0
K	0	0	0	0	8	4	17	5	0	0	0	0	5	5	5	0
L	0	0	0	0	5	9	13	11	0	0	0	0	8	4	4	0
M	0	0	0	0	0	0	0	0	2	1	5	8	0	0	0	3
N	0	0	0	0	0	0	0	0	3	6	5	4	0	0	0	5
O	0	0	0	0	0	0	0	0	6	8	5	4	0	0	0	4
P	0	0	0	0	0	0	0	0	0	0	0	0	3	5	4	0

Fonte: elaborado pelo autor.

	Análise e Desenvolvimento de Sistemas Ciência da Computação Sistemas de Informação	2017/1 Trabalho GA (3,0) Laboratório II
	Prof. Gilberto Irajá Müller	

Ao observar a Figura 2, pode-se fazer as seguintes conclusões:

- Do vértice A é possível ir apenas para os vértices B, C e D;
- Em relação à distância: A até B é 2; A até C é 1 e A até D é 3; considera-se a distância uma unidade qualquer (metros, quilômetros, etc.);
- Ressalta-se ainda que uma matriz de adjacência é espelhada, ou seja, os valores podem ser obtidos tanto a partir da linha como a partir da coluna; como se fôssemos dobrar uma folha; não entraremos em detalhes do grafo ser dirigido;
- As linhas e colunas da matriz (0, 1, ..., n - 1, n) correspondem as letras dos vértices: (A, B, ..., O, P).

Estruturas de Dados


PASSO 1) Você deverá criar a seguinte interface, pois é o contrato obrigatório para a execução da classe de teste.

```
public interface IPokemonGo {
    public void loadMatrix(File file) throws IOException;
    public Chromosome generateChromosome();
    public void insertOrderedChromosome(Chromosome chromosome);
    public int distance(int[] genes);
    public void evolve();
    public void print();
}
```

PASSO 2) Deverá ser criado uma classe denominada PokemonGo que implementa a interface IPokemonGo conforme Quadro 1.

Quadro 1: Descrição dos Contratos

Método/operação	Descrição
loadMatrix	Responsável pelo carregamento da matriz de adjacência constante em arquivo csv (ver ANEXO A).
generateChromosome	Responsável por gerar um novo cromossomo de forma aleatória (ver ANEXO B).
insertOrderedChromosome	Responsável pela inserção ordenada em um array de "melhores" cromossomos. Guardaremos os 10 melhores cromossomos em ordem de distância (crescente) neste array. Se o cromossomo a ser inserido é ruim aos que já estão no array e o array está todo preenchido, então, este deverá ser desconsiderado. Se o

	Análise e Desenvolvimento de Sistemas Ciência da Computação Sistemas de Informação	2017/1 Trabalho GA (3,0) Laboratório II
	Prof. Gilberto Irajá Müller	

	cromossomo já consta no array, então, deverá ser desconsiderado. Caso seja um cromossomo bom e o array esteja completo, então, o pior cromossomo deverá ser retirado da lista, sendo que o novo cromossomo deverá ser inserido. Caso o array não esteja cheio, então, deverá ser inserido conforme sua distância (ou seja, qualidade). Poderão existir cromossomos com o mesmo fitness (critério de aptidão/qualidade).
distance	Responsável por calcular a distância total do cromossomo, ou seja, o fitness. Observa-se que o parâmetro se refere aos genes do cromossomo. Utiliza-se a matriz de adjacência no cálculo da distância.
evolve	Responsável pela evolução/seleção dos melhores cromossomos. Faremos 100 iterações, onde cada iteração irá gerar um cromossomo e este inserido no array de melhores cromossomos.
print	Responsável pela impressão dos 10 melhores cromossomos, do menor caminho para o maior caminho.

Fonte: elaborado pelo autor.

PASSO 3) Abaixo, segue estrutura da classe de teste:

```

public class PokemonGoTest {
    public static void main(String[] args) {
        IPokemonGo pokemonGo = new PokemonGo();
        try {
            pokemonGo.loadMatrix(new File("ARQUIVO MATRIZ ADJ."));
        } catch (IOException e) {
            e.printStackTrace();
        }
        pokemonGo.evolve();
        pokemonGo.print();
    }
}


```

O resultado da execução do PokemonGoTest deve ser algo similar ao que está abaixo. A Figura 3 ilustra o primeiro caminho.

```

Short path -> 9 A C G J M P
Short path -> 12 A D G J M P
Short path -> 13 A B G J M P
Short path -> 13 A B H I M P
Short path -> 14 A B H J M P

```


	Análise e Desenvolvimento de Sistemas Ciência da Computação Sistemas de Informação	2017/1 Trabalho GA (3,0) Laboratório II
	Prof. Gilberto Irajá Müller	

ANEXO B) Para gerar um novo caminho, iremos utilizar um conceito básico da genética que é o de cromossomo¹ e gene. Um cromossomo é composto por vários genes, contudo, teremos apenas "6" genes em função do comprimento entre o vértice A (nossa localização) e o vértice P (localização do Pokémon). Devemos utilizar um array unidimensional de inteiros que são os genes e representam o cromossomo. Exemplo de um cromossomo:

Gene 1	Gene 2	Gene 3	Gene 4	Gene 5	Gene 6
0	2	6	9	12	15

A distância total é: **9** e o caminho: **A -> C -> G -> J -> M -> P**.

Como gerar um novo cromossomo? Imagine que a coluna à esquerda do quadro abaixo seja o índice do array do nosso cromossomo, então, para os índices 1 a 4, iremos gerar números aleatórios que representam as opções possíveis daquele vértice no grafo.

Index

0	Sempre será 0 (vértice A).
1	Sortear valores entre 1 (B), 2 (C) ou 3 (D).
2	Sortear valores entre 4 (E), 5 (F), 6 (G) ou 7 (H).
3	Sortear valores entre 8 (I), 9 (J), 10 (K) ou 11 (L).
4	Sortear valores entre 12 (M), 13 (N) ou 14 (O).
5	Sempre será 15 (vértice P).

Após a geração do cromossomo, deverá ser calculada a distância e armazená-la no atributo fitness do cromossomo. A Figura 4 apresenta a classe de sugestão para o cromossomo.


Figura 4: Estrutura Cromossomo

Chromosome
- genes : int[] - fitness : int
+ Chromosome() + Chromosome(genes : int[], fitness : int) + getGenes() : int[] + setGenes(genes : int[]) : void + getFitness() : int + setFitness(fitness : int) : void + toString() : String

powered by Astah

Fonte: elaborado pelo autor.

¹ Leia-se cromossomo bom para aquele que tem um caminho curto. Leia-se cromossomo ruim para aquele com um caminho longo.

	Análise e Desenvolvimento de Sistemas Ciência da Computação Sistemas de Informação	2017/1 Trabalho GA (3,0) Laboratório II
	Prof. Gilberto Irajá Müller	

Avaliação

- Comentários em todas as classes/métodos;
- O código deverá ser claro (coloque nomes que mostre a intenção da ação);
- Não será aceito a apresentação do trabalho após a data limite; observa-se que a nota será considerada apenas mediante apresentação;
- Não será aceito trabalho igual ao de outros colegas ou cuja solução tenha sido copiada da internet ou de livros. Esta prática é chamada de plágio;
- O trabalho deve ser apresentado para o professor em aula.