

OpenKarel

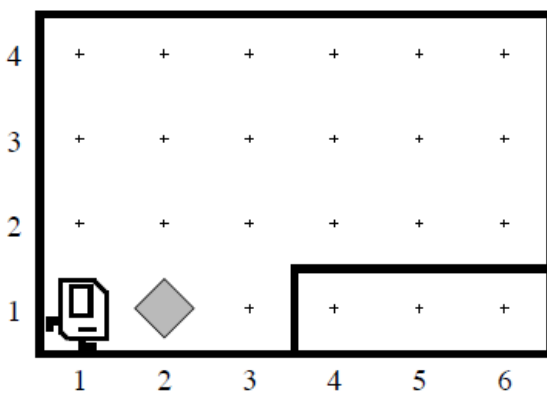
Série Java – Cartão de Referência

Foi criado por **Rich Pattis** para o ensino de fundamentos da programação na Universidade de Stanford, California. O nome é uma homenagem a **Karel Čapek**, que escreveu uma peça sobre o **Mundo dos Robôs**.

1. Classe Básica

```
import openKarel.*;
public class TstKarel extends XKarel {
    public static void main(String [] args) {
        new TstKarel();
    }
    public void run() {
        /* Comandos Aqui */
    }
}
```

Dica: Encerrar a janela com o mundo de Karel para modificar o programa e novos comandos.



Mundo de Karel

O mundo de Karel é definido por **Ruas** (*streets*) na horizontal (leste-oeste) e **Avenidas** (*avenues*) na vertical (norte-sul). O cruzamento entre uma rua e uma avenida é chamado de **Esquina** (*corner*). Karel é posicionado em alguma esquina virado para uma das quatro direções padrões da bússola (norte, sul, leste, oeste). Na figura ao lado, **Karel** está localizado na esquina da 1ª Rua com a 1ª Avenida, virado para o leste.

O objetivo de **Karel** é sinalizar todas as passagens que encontrar no caminho, deve arrumar todos sinais que foram colocados em posições errôneas.

Condições Positivas de Teste

```
frontIsClear()
leftIsClear()
rightIsClear()
beepersPresent()
beepersInBag()
facingNorth()
facingEast()
facingSouth()
facingWest()
```

Condições Opostas de Teste

```
frontIsBlocked()
leftIsBlocked()
rightIsBlocked()
noBeepersPresent()
noBeepersInBag()
notFacingNorth()
notFacingEast()
notFacingSouth()
notFacingWest()
```

Dicas Gerais:

O programa deve ser capaz de trabalhar com comprimentos arbitrários. Não faz sentido conceber um programa que funciona apenas para mapas com um número predeterminado de ruas e avenidas. Em vez disso, crie programas que possam realizar a mesma tarefa em qualquer tipo de mapa. Tais programas, devem possuir a inteligência suficiente para reconhecer seu mundo.

Uma passagem pode ocorrer em qualquer posição no mapa. Não deve haver limites para o número de passagens ou paredes bloqueando o caminho de Karel. Uma passagem é identificada por uma abertura na parede que representa a superfície do mapa.

Passagens existentes já podem ter sido sinalizadas. Qualquer uma das passagens já pode conter um sinalizador deixado por uma equipe anterior de reparos. Nesse caso, **Karel** não deve colocar um sinalizador adicional.

Crie nova funcionalidades para Karel. Crie métodos como `turnRight()`, `putBeeperIfBeeperPresent()`, entre outros, procure especializar as funcionalidades de Karel.

Regra Básica: Crie programas fáceis de ler, divida-os em curtos métodos ou em classes adicionais de modo a facilitar o máximo possível sua leitura.

Instalação: Obter a biblioteca **openKarel.jar** e disponibilizá-la para seu projeto, no **BlueJ** em "Tools | Preferences | Libraries". Se desejar os fontes podem ser conseguidos no mesmo local.