

Exercício 01

Desenvolva uma classe **Carro**, com os seguintes atributos:

Marca (não pode ficar vazio);

Modelo (não pode ficar vazio);

Quantidade de combustível, em litros (não pode ser negativo);

Consumo médio, em quilômetros por litro (não pode ser negativo);

Quilometragem (não pode ser negativo);

Encapsule todos os dados (private), e não crie métodos set.

Crie os seguintes métodos:

- **dois construtores**: um que recebe todos os dados, e outro que recebe todos exceto a quilometragem e o nível de combustível (indicando um carro zero km e sem combustível).

- **andar**: Recebe a quilometragem a ser percorrida, que deve ser maior que zero. Caso haja combustível suficiente para rodar a quilometragem pedida (de acordo com o consumo do carro), faz as seguintes ações:

- incrementa a quilometragem do carro;
- diminui a quantidade de combustível (de acordo com o consumo do seu carro);

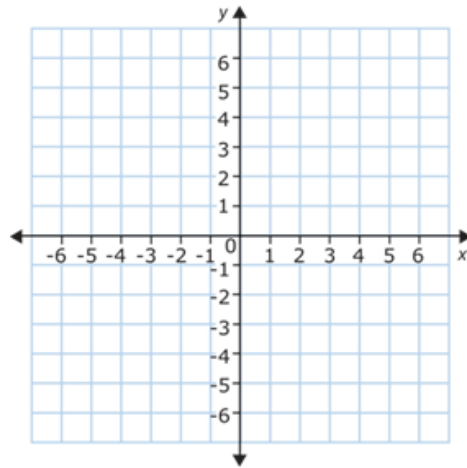
- **abastecer**: Recebe uma quantidade de combustível (valor positivo) e adiciona ao tanque do carro. Não pode receber um valor negativo;

- **exibir (toString)**: retorna uma String com todos os dados do carro;

Crie um objeto na **main** e faça o teste dos métodos desenvolvidos.

Exercício 2

Implemente o problema do robô andando em uma sala, conforme o diagrama abaixo:



Deve existir um método para definir a quantidade máxima de linhas e colunas da sala (tamanho do ambiente) e inicializar o ambiente.

O método que inicia o **Robo** deve ler a posição inicial do robô.

Os métodos da classe **Robo** tem as seguintes funcionalidades:

- **norte()** → faz com que o robô se desloque +1 no eixo y
- **sul()** → faz com que o robô se desloque -1 no eixo y
- **leste()** → faz com que o robô se desloque +1 no eixo x
- **oeste()** → faz com que o robô se desloque -1 no eixo x

O ato de andar modifica as variáveis de posição do robô (x e y).

Você deve verificar se é possível o comando pedido, pois ele deve respeitar o tamanho máximo definido para o ambiente

Escreva um **App** contendo um menu com as seguintes opções:

- 1 - Andar para Frente
- 2 - Andar para Trás
- 3 - Parar
- 4 - Virar para Direita
- 5 - Virar para Esquerda
- 6 - Sair

Exercício 3

Criar a classe **Livro** com os atributos:

título, autor, gênero e preço.

Criar uma classe **Livraria** para gerenciar um *ArrayList<> de Livro*, com os seguintes métodos:

- **adicionar()** : recebe um Livro e adiciona na lista
- **listar()** : retorna uma String com os dados de todos os Livros cadastrados
- **pesquisarLivro()** : recebe uma String como parâmetro que é o título do livro. Retornar os dados do livro.
- **pesquisarGenero()** : recebe uma String como parâmetro que é o gênero do livro. Retornar o total de livros do gênero informado.
- **remover()** : remove da lista um livro pelo título informado como parâmetro
- **calcularTotalAcervo()** : retorna o valor total do acervo da Livraria

Criar a classe **AppLivraria** (main) que utiliza a classe Livraria