

Nome: _____

Matrícula: _____

Data: **04/02/2022**

Disciplina: **DCC305 – PROG. ORIENTADA A OBJETOS**

Período: **2021.2**

AVALIAÇÃO PARCIAL I

1. Crie a classe de agenda do estudante. Nessa classe o aluno poderá inserir e obter informações do seu nome, o curso que ele está fazendo, o período que ele está cursando, as disciplinas que ele está cursando nesse período, seu quadro de horário, informar os professores de cada disciplina. Além disso, ele poderá informar as atividades pendentes em cada disciplina, como, exercícios, seminários e provas. Lembre-se que cada atividade possui uma data de entrega e uma nota a ser atribuída a ela. (Obs.: A classe principal é Agenda, no entanto, outras classes podem ser criadas para implementar agenda).
2. Na agenda adicione duas funcionalidades soma das notas das atividades realizadas na disciplina, e média das notas das atividades. (O método média deve ter dois parâmetros total da soma das notas das atividades e o denominador que dividirá a soma das notas).
3. Crie uma classe cachorro sabendo que um cachorro tem um nome, pode ter um dono, tem um tipo de pelagem (longo ou curto), tipo de focinho (Braquicefálicos, Mesocefálicos e Dolicocefálicos), raça, tem sentimentos, tem uma cor de pelos. O cachorro, ele pode correr, o dono pode passear com ele, pode mudar de dono, pode ficar doente, pode se acidentar, pode perder uma das patas, pode latir. (Obs.: cada raça de cachorro tem características próprias, por exemplo um pastor alemão, pode ser de pelo longo ou curto depende da subespécie, a cor do pelo pode ser ruivo, branco, preto com cinza, capa branca entre outros. Dependendo da espécie o cachorro pode latir e uivar, além disso, cada cachorro tem seu próprio latido. Para raça de cachorro implemente para as seguintes raças: pastor alemão, pitbull, Husky siberiano, pinscher e Dachshund)
4. Programa labirinto de rato. Nesse programa três ratos são soltos em um labirinto procurando uma saída. Os três ratos são soltos ao mesmo tempo no início do labirinto sendo que cada um deles segue o seu próprio caminho e instinto para encontra a saída. No programa os movimentos dos ratos no labirinto são computados em “paralelo”, quando um dos ratos encontrar a saída os outros ratos devem parar de se movimentar (o programa deve ser terminado) e três labirintos devem ser exibidos com a movimentação de cada rato, além de, ser identificado o rato vencedor. O labirinto dos ratos que deve ser implementado é demonstrado abaixo, onde:
 - * representa parede;
 - @ representa caminho disponível para movimentação;
 - E representa entrada do labirinto;
 - S representa saída do labirinto.

Além disso, os ratos só podem se movimentar para frente, atrás, direita e esquerda ($\uparrow \downarrow \rightarrow \leftarrow$) e o tamanho do labirinto é 9×33 (linhas x colunas).

```
*E*****
*@@@@@@*@@@@@@@@@@@@@*@@@@@@@@@@@@@
*@*****@*@@@@*@@@@*@@@@*@@@@*@@@@*
*@@@@@@@@*@@@@*@@@@*@@@@*@@@@*@@@@*
*@*****@*@@@@*@@@@*@@@@*@@@@*@@@@*
*@@@@@@@@*@@@@*@@@@*@@@@*@@@@*@@@@*
*@*****@*@@@@*@@@@*@@@@*@@@@*@@@@*
*@@@@@@@@*@@@@*@@@@*@@@@*@@@@*@@@@*
*****S*****
```

Dica:

- Use thread na implementação;
- Pesquise sobre algoritmos de resolução de labirinto (maze escape algorithm ou Maze-solving algorithm)