Projeto de Trainee

Estratégia - VSSS

Crie um projeto em c++, que tem como objetivo simular uma situação de jogo no VSSS.

1. Orientações:

* Crie uma classe com o nome “robots”, que deve possuir as seguintes propriedades:

\*Número de identificação;

\*Posição (x,y, theta);

\*Meta (x, y)

\*Caso necessário crie mais elementos;

* Crie um segunda classe nomeada de “utils” responsável em fornecer funções gerais.

Elementos:

\*função que calcula a distância entre dois pontos (x,y).

Formula: distancia = raiz((x1-x2)² + (y1-y2)²)

\*Função que retorna o ângulo que o robô deve seguir (dado a posição do robô e a meta)

Fórmula: ângulo entre dois pontos

* Crie uma terceira classe nomeada de “game functions” que herda a classe utils e que será responsável pelo controle de metas para os robôs.

Elementos:

\*Vetor de objetos da classe robots.

\*Dois elementos intitulados, centroidAtk e centroidDef, que devem possuir coordenadas (x,y);

\*Duas funções diferentes, uma para controlar a meta do goleiro e outra para controlar a meta do zagueiro. Essas funções devem receber 2 parâmetros, o robo a ser controlado e a posição da bola. Como retorno, elas devem printar o ângulo que o robô deve seguir para atingir a sua meta (considere trajetórias retas).

Essas duas funções devem considerar duas situações diferentes para definir as metas dos robôs, uma quando a bola está no campo de atk e outra quando a bola está no campo de defesa, a definição de limites onde é campo de atk e defesa fica a critério, mas sejam sensatos.Criem a função de forma genérica, isto é, a função deve conseguir abranger diversos valores de entrada para a posição da bola e robô. Por fim a função deve printar os dados do robô (posição, angulo, meta, etc..) , dados da bola(posição) e o ângulo calculado que o robô deve seguir; (Cuidado para não confundir ângulo do robô com o ângulo que ele deve seguir)

* Sugestão: Crie uma struct que armazene variaveis x e y, e utilize essa struct para representar todas as posições que possuem posição (x,y) em todas as classes.
* Ao criar uma classe deve ser feito o arquivo .h e o .cpp, ex: utils.h e utis.cpp e sempre procurar utilizar construtor e destrutor para as classes.

2. Inicialização das classes:

* Os centroids são pontos fixos utilizados para referenciar os lados do campo, isto é, onde é o atk e onde é a defesa. Portanto, inicie os centroids no construtor da classe GameFunctions com as posições fixas,

centroidAtk = (160,65), centroidDef = (10,65).

* Considere que o campo possui dimensões 170 x 130 cm sendo 170 referente ao eixo x e 130 ao eixo y;

3. Programa principal

O seu projeto deve ter um arquivo main.cpp onde os dados serão validados, nele deve ser criado um objeto da classe GameFunctions ,um vetor com elementos da classe robot, um elemento com coordenadas (x,y) para representar a bola. Execute os testes de suas funções no main.