Objetivo:

"A corrida se caracteriza o Robotino percorrer de um ponto A até um ponto B, no menor tempo possível, evitando obstáculos.

Requisitos:

Funcionais

* Busca de caminho sem interferência externa
* Ser capaz de desviar obstáculos (Complementar tipo de obstáculo)
* Armazenamento da cartografia
* Telemetria da cartografia do percurso do robô
* Interface com usuário para exibir cartografia
* Ir de ponto A a ponto B
* Deve apresentar tempo de resposta aceitável (especificar, perguntar do prof)

Não funcionais:

* Utilizar plataforma robotino
* Não sair da área demarcada
* Usuário não interage com robô durante corrida

Testes:

* Testes modularizados
* Testes devem considerar tipos de obstáculos que serão utilizados
* Testes considerando múltiplos cenários
* Devem haver testes em ambiente real
* Resultados de teste devem apresentar X% de sucesso

Análise

Diagrama de classes

Testes

* Bottom-up
* Testar classes vs testar funcionalidades
* DOcumentação do teste
* Teste de aceita = teste de múltiplos cenários
* Fluxos de teste (diagramas…)

14 novembro – parte de análise

19 de novembro - sem aula

21 de novembro – design

26, 28 de novembro – linguagem de programação

03, 05 dezembro - exercícios

10, 12, 14 dezembro – seminário

17 a 21 dezembro – Robotino corrida – quarta-feira dia 19

quais documentos devem ser apresentados...

importante domínio do processo

O trabalho final da disciplina, como já mencionamos, será a corrida de obstáculos com Robotino. Cada equipe irá desenvolver um software para controlar o Robotino e documentar o processo de desenvolvimento com os documentos já disponibilizados pelo professor Vicente. A seguir uma descrição do que deve ser desenvolvido no projeto, onde vocês podem definir os requisitos do sistema:

"A corrida se caracteriza o Robotino percorrer de um ponto A até um ponto B, no menor tempo possível, evitando obstáculos. O local onde será a corrida, será numa área retangular, sem inclinações, onde o ponto A será uma das bordas do retângulo e o ponto B, a borda oposta. O Robotino não pode sair da área demarcada pela corrida. Os obstáculos serão simples resmas de papel espalhadas pelo caminho em diferentes posições e Robotino não pode empurrar os obstáculos durante a corrida.

O software pode ser desenvolvido em quaisquer uma das linguagens de programação orientadas à objetos da API disponível pela Festo e deve possuir uma interface gráfica de interação com usuário, minha recomendação pessoal seria usar Java e apenas uma linguagem de programação. O Robotino deve ser posicionado no ponto A, e após iniciar o programa, ele deve conseguir chegar até o ponto B sem quaisquer tipos de interferência externa ou input no software, caso haja uma falha durante o percurso, a corrida poderá ser reiniciada somente uma vez. Ao iniciar o percurso o software deve gerar uma cartografia do percurso em execução, para que o caminho escolhido possa ser usado na segunda corrida, onde todos correrão com um caminho salvo do trajeto anterior.

Para que a corrida seja a mais justa possível, todas as equipes correrão no mesmo percurso com os obstáculos nas mesmas posições, e equipe vencedora será aquela que obtiver a menor média dos dois trajetos. Quanto aos algoritmos de “path finding” e como usuário interage com a GUI, ficará a critério das equipes, desde que respeite as regras descritas anteriormente”

2 gerente: Anderson, Mikaela

3 desenvolvimento: Adriano, Edmilson, Rebeca

2 teste: Paulo Henrique, João

Principais documentos:

Plano de projeto - inclui alocação de recursos e tempo, fases do projeto

* diagrama de estrutura do projeto, fases etc relaciona com cronograma
* Marcos - relaciona sempre com documento produzido
* Análise, Design: inclui diagrama de classes, statechart, explicar fluxograma

Documento de requisitos – 16 de novembro, funcionais, não funcionais etc

Implementação - codigo fonte

Testes - documentação de casos de teste, resultado de teste??

Documento de aceitação - ?? Relatório Final