

Documentação do Código

Seguem descritas abaixo as funções implementadas por nós para a realização do labirinto, as funções não citadas aqui são de bibliotecas conhecidas e já utilizadas em aula.

void drawCube(float size)

Função utilizada para desenhar as paredes do labirinto. As paredes no labirinto são formatas por cubos de arestas de tamanho *size*.

void drawFloor(float size)

Função utilizada para desenhar o chão do labirinto por onde passa o robô. O chão é desenhado juntando 4 vértices por vez em uma superfície quadrada plana de arestas de tamanho *size*.

void drawRobotLegs()

Utilizada na construção do robô, essa função é chamada pela função *drawRobot()* para desenhar as pernas do robô. Para representar as pernas são desenhados dois cilindros os quais são transladados para cima e rotacionados em 90 graus para baixo, assim fazendo com que uma rotação no eixo X faça as pernas imitarem um movimento real.

void drawRobotArms()

Utilizada na construção do robô, essa função é chamada pela função *drawRobot()* para desenhar os braços do robô. Usando da mesma lógica de desenhar as pernas são utilizados dois cilindros também rotacionados em 90 graus para que o movimento destes apresentem mais realismo. Além disso, os cilindros são rotacionados para longe do corpo do robô em 20 graus a fim de dar mais realismo. E por fim é colocado encaixando na abertura de cada cilindro uma esfera imitando os ombros.

void drawRobotBody()

Utilizada na construção do robô, essa função é chamada pela função *drawRobot()* para desenhar o tronco do robô. O tronco do robô é, em suma, um cubo cujo os vértices são dados pelo vetor bidimensional *GLfloat robotBody*.

void drawRobotHead()

Utilizada na construção do robô, essa função é chamada pela função *drawRobot()* para desenhar a cabeça do robô. A cabeça do robô é representada por uma esfera de raio 1. Na cabeça do robô é desenhado um paralelepípedo cinza onde uma faixa vermelha representa os olhos do robô. Além disso, em cima da cabeça é desenhado uma antena que é composta por um cone com uma pequena esfera na ponta e um disco ao redor deste cone.

void drawRobot()

Essa é a função que chama todas as acima explanadas que estão relacionadas com o robô. Ela é responsável por desenhar cada uma das partes do robô e fazer as translações necessárias para desenhar cada parte em seu devido lugar. Além disso, é responsável pelo incremento, decremento e controle de limites das variáveis que coordenam os movimentos das pernas e braços através de rotações nas funções *drawRobotLegs()* e *drawRobotArms()*, respectivamente.

void drawTable()

Função que desenha um dos objetos requisitos do trabalho. Desenha uma mesa com um *teapot* em cima. Os pés da mesa são quatro cilindros e a parte plana da mesa são dois discos, um maior por fora e um menor mais ao centro. E em cima do todo, um *teapot* de tamanho relativo 1.

void drawFountain()

Função que desenha um dos objetos requisitos do trabalho, uma fonte. A qual a base é composta por um *torus* de cor cinza e no meio desse se encontra um disco azul para representar a água. Do centro destes se ergue um pilar que é formado por: um cone com duas pequenas esferas na ponta, as quais estão ligadas a um segundo cone, esse por sua vez está rotacionado em 180 graus para ficar espelhado em relação ao primeiro. Na base desse último cone é colocado um disco com um raio ligeiramente menor que o raio da base do cone, para representar água no topo do pilar.

void draw(void)

Essa função é a função principal de desenho do projeto. Ela é responsável por desenhar o labirinto e todos os objetos acima citados. No começo ela decide qual das câmeras irá usar, sendo que a principal é a visão de todo o labirinto. Depois, o labirinto é construído em cima da matriz *maze* onde os 1's representam os blocos das paredes e os 0's representam o chão do labirinto. Nessa função também é decidido onde serão desenhados os objetos (fonte e mesa), depois, é feito o desenho do robô, transladando ele para onde ele deve ser desenhado. Por último, é feito o trajeto do robô de forma que ele caminhe suavemente e também gire a câmera para que sempre fique atrás dele.

void redraw(int)

Função de redesenhar todo o projeto, passada por parâmetro na função *glutTimerFunc()* e determina que tudo será redesenhado a cada 50 milissegundos.