***Robby, o robô que limpa o mundo***

*Esse artigo tem como objetivo descrever nossa implementação do código que foi pedido no trabalho 1 - Meta-heurística. Neste trabalho foi implementado a ideia do algoritmo genético. Todas as variáveis são inteiras.*

Primeiramente, o código começa com as inclusões das bibliotecas necessárias, as definições das constantes usadas no código, para facilitar a implementação e a manutenção, e a  declaração da matriz mundo do robô. Depois é definido um Struct, onde tem-se uma variável para a posição no eixo X ( representando as linhas da matriz ) e outra variável para a posição no eixo Y ( representando as colunas da matriz ).

Tem uma função que recebe um valor e retorna um número aleatório de zero até menor que o valor recebido.

Tem uma função para preencher a matriz que representa o mundo do Robby, as laterais inferiores e superiores são preenchidas com o valor dois que simboliza o limite onde Robby pode andar, os demais valores são preenchidos aleatoriamente com o número um para representar onde haverá uma lata ou com o número zero para representar a ausência de uma lata.

Tem uma função para imprimir a matriz mundo.

Tem uma função para imprimir uma geração de indivíduos.

Tem uma função para gerar aleatoriamente os movimentos de cada indivíduo de cada geração.

Tem uma função que recebe a geração e um indivíduo e retorna sua pontuação.

Tem uma função que recebe uma geração atual, uma geração passada, um indivíduo elegido da geração passada e um indivíduo sorteado da geração atual, onde o indivíduo da geração atual recebe os dados do indivíduo da geração passada.

Tem uma função para fazer o cruzamento, ela recebe uma geração e três indivíduos, onde o terceiro indivíduo recebe a primeira metade dos dados do primeiro indivíduos e a segunda metade do segundo indivíduo.

Tem uma função para fazer a mutação, que recebe uma geração, onde cada indivíduo tem seus dados levemente alterados, provocando assim uma perturbação nos seus registros.

Tem uma função para restaurar as posições dos indivíduos, fazendo os retornar para a posição de origem sempre que uma nova geração é iniciada.

Tem uma função para inicializar as variáveis e as posições.

Tem uma função que recebe uma geração e retorna a melhor pontuação daquela geração.

E finalmente temos o ‘main’. Começamos com uma função para não viciar a geração de números aleatórios, depois a declaração da matriz que guardará todas as gerações de indivíduos, uma matriz auxiliar que guardara uma geração de indivíduos, três variáveis auxiliares que representaram os indivíduos sorteados aleatoriamente e por último, duas variáveis auxiliares que representaram a pontuação de dois indivíduos.

Então preenche o mundo e gera os movimentos de cada indivíduo da primeira geração.

Começa um laço de repetição que vai da segunda geração até a última. Para cada iteração no laço de repetição, a matriz auxiliar recebe a geração anterior, para fins de comparação. Tem um outro laço de repetição para fazer o cruzamento entre os indivíduos da geração, explicado acima. Tem um outro laço de repetição responsável que sortear dois indivíduos, ver qual deles teve a pontuação maior e “jogá-lo” para a próxima geração. Depois é feita a mutação dos indivíduos, também explicado acima e a impressão da matriz de indivíduos da geração atual.