

Relatório relativo ao simulador de corridas

ESTRUTURAS DE DADOS	3
Data	3
Pilotos	3
Carros	3
Par_carros_pilotos	3
Classificado	3
Voltas	4
ESTRUTURAS DINÂMICAS:	4
Estrutura dinâmica que guarda as classificações/ resultados em cada volta	4
Estrutura dinâmica que guarda as voltas	4
Estrutura dinâmica que permite guardar as pontuações ao longo dos campeonatos	4
FICHEIROS IMPLEMENTADOS	5
main.c	5
pilotos.h	5
corridas.c	5
utils.h	5
utils.c	5
MANUAL DE UTILIZAÇÃO	6
Menu Principal	6
Caso escolha a opção 1	6
Caso escolha a opção 2	6
Caso escolha a opção 3	6
Caso escolha a opção 4	7
Caso escolha a opção 5	7
Caso escolha a opção 0	8
Principais funções	8
faz_voltas	8
Recebe	8
Procedimento	8
cria_par	9
Recebe	9
Procedimento	9

ESTRUTURAS DE DADOS

Data

permite guardar uma data

ano - usado para guardar um ano

mês - usado para guardar um mês

día - usado para guardar um dia

Pilotos

serve para guardar toda a informação relativa a um piloto

nome - nome do piloto

id - id do piloto

born - data de nascimento do piloto (ver estrutura "data")

peso - peso do piloto

xp - experiência do piloto

impedimento - impedimento, seja por lesionamento comportamento inaceitável de um piloto

Carros

serve para guardar toda a informação relativa a carros

id - id do carro

power - potencia do carro avaria- avaria do carro

Par_carros_pilotos

serve para guardar pares de carros e pilotos

prox - ponteiro que aponta para estrutura deste tipo

pil - serve para guardar informação referente a um piloto (ver estrutura "pilotos")

car - serve para guardar informação referente a um carro (ver estrutura "carros")

Classificado

serve para guardar a classificação de um par de carros-pilotos em determinada volta de uma corrida específica

nome - nome do piloto

pilotoID - id do piloto

carID - id do carro

tempo - tempo feito pelo par carro-piloto nessa volta

acidente - indicação de um acidente (é inicializada com valor 0, porém, em caso de acidente, guarda o valor da volta em que este aconteceu)

tempo_total - tempo total em todas as voltas dessa corrida que o par piloto-carro tem até ao momento.

idade - idade do piloto

Voltas

serve para guardar dados relativos a volta, como o número da volta, um ponteiro para a volta seguinte e um para o melhor classificado no momento dessa volta

volta_n - número da volta na corrida em que esta

prox - ponteiro para um estrutura do tipo "voltas", usado para indicar a volta seguinte

proxC - ponteiro para uma estrutura do tipo "classificado", usado para apontar para o melhor classificado no momento dessa volta

ESTRUTURAS DINÂMICAS:

Estrutura dinâmica que guarda as classificações/ resultados em cada volta

- Tem base na estrutura de dados "classificado";

- Permite mover facilmente a posição dos pilotos com base na sua classificação;

- Permite guardar um ponteiro para a classificação seguinte, o nome e ID do piloto desse par, o tempo da sua volta, o tempo total já feito na corrida, a idade do piloto e um indicador de se já sofreu ou não acidente.

Estrutura dinâmica que guarda as voltas

- Tem como base a estrutura de dados "voltas";

- Permite guardar o valor da sua volta, um ponteiro para a volta seguinte e um ponteiro para o primeiro classificado da sua volta;

- Permite-me alguma facilidade a chegar a cada uma das voltas e não necessito de enviar informação extra (como o seu tamanho) para funções externas, porém talvez ter criado um array "normal" fosse melhor.

Estrutura dinâmica que permite guardar as pontuações ao longo dos campeonatos

- Tem como base a estrutura de dados "pontuação";

- Permite guardar o nome e ID de um piloto, um indicativo de se já sofreu um acidente ao longo deste campeonato, a idade do piloto, o id do carro a pontuação do piloto no campeonato e um ponteiro para o piloto a seguir classificado;

- Os pilotos estão ordenados conforme a sua pontuação,

FICHEIROS IMPLEMENTADOS

main.c

Ficheiro principal cuja estrutura se baseia nas opções que o utilizador pode tomar a partir da função principal e que chama todas as funções necessárias à realização do programa

pilotos.h

Ficheiro do tipo header que guarda todas as referências a estruturas de dados e a funções secundárias;

Faz interligação entre o ficheiro main.c e os ficheiros corridas.c e funcoes.c.

funcoes.c

Ficheiro que inclui todas as funções referentes à leitura dos ficheiros necessários a ler para a inicialização do programa, amostragem e, em alguns casos, modificação desses mesmos dados;.

Utiliza pilotos.h como via intercalar entre este ficheiro e o ficheiro main.c.

corridas.c

Ficheiro que inclui todas as funções referentes a criação e simulação de corridas individuais/treino e campeonatos;

Utiliza pilotos.h como via intercalar entre este ficheiro e o ficheiro main.c;

Utiliza utils.h como via intercalar entre este ficheiro e o ficheiro utils.c.

utils.h

Ficheiro do tipo header que guarda todas as referências a funções disponibilizadas para a realização do trabalho;

Faz interligação entre os ficheiros utils.c e corridas.c.

utils.c

Ficheiro do tipo header que guarda todas as funções disponibilizadas para a realização do trabalho.

MANUAL DE UTILIZAÇÃO

O utilizador deverá colocar dois ficheiros específicos na mesma pasta onde se encontra o executável:

“pilotos.txt”- ficheiro de texto que contém todos os dados referentes aos pilotos, ordenados conforme o enunciado do trabalho indica.

“carros.txt”- ficheiro de texto que contém todos os dados referentes aos carros, ordenados conforme o enunciado do trabalho indica.

De seguida deverá iniciar o executável do programa, onde será apresentado o menu principal.

Menu Principal

Ir-lhe-á aparecer o menu principal, onde poderá indicar um valor correspondente a opção que deseja

"1" - mostra a lista e características dos carros.

"2" - mostra a lista e características dos pilotos.

"3" - cria uma corrida/ treino

"4" - cria um campeonato

"5" - penalizar um piloto por mau comportamento

"0"- desligar o programa

Caso escolha a opção 1

O programa mostrará na consola a lista dos carros existentes no mesmo, assim como as suas características.

Caso escolha a opção 2

O programa mostrará na consola a lista dos pilotos existentes no mesmo, assim como as suas características.

Caso escolha a opção 3

O programa pedirá para introduzir o número máximo de pilotos que a pista pode levar, e indicará todos os pilotos e pares que participam, assim como os que não participam e a razão para não participarem

O programa pedirá para introduzir o número de voltas da pista, valor entre 5 e 10

O programa pedirá para introduzir o comprimento da pista, valor entre 500 e 10000

Após isto, o programa irá simular cada uma das voltas, mostrando a classificação dos concorrentes em cada volta.

Entre cada volta existe um intervalo de espera de 5 segundos.

No final de todas as voltas o programa irá mostrar um menu de classificação, o utilizador pode escolher entre:

- Ver a classificação numa volta específica.

- Ver as classificações em todas as voltas (sem o delay de 5 segundos)

- Ver a classificação final dos concorrentes.

- Desligar o programa.

O utilizador pode ver quantas vezes quiser as classificações que desejar, o programa só avançará quando o utilizador decidir sair do menu (utilizando a opção 0)

Após isto, o programa voltará ao seu menu principal, os dados dos carros e dos pilotos já se encontram atualizados.

Caso escolha a opção 4

O programa vai pedir o número de corridas que o campeonato vai ter, valor entre 3 e 8

O programa pedirá para introduzir o número máximo de pilotos que a pista pode levar, e indicará todos os pilotos e pares que participam, assim como os que não participam e a razão para não participarem

O programa pedirá para introduzir o número de voltas da pista, valor entre 5 e 10

O programa pedirá para introduzir o comprimento da pista, valor entre 500 e 10000

Após isso o programa irá simular a primeira corrida.

No final de todas as voltas o programa irá mostrar um menu de classificação, o utilizador pode escolher entre:

- Ver a classificação numa volta específica.

- Ver as classificações em todas as voltas (sem o delay de 5 segundos)

- Ver a classificação final dos concorrentes.

- Desligar o programa.

O utilizador pode ver quantas vezes quiser as classificações que desejar, o programa só avançará quando o utilizador decidir sair do menu (utilizando a opção 0);

Após a execução da primeira corrida, será perguntado ao utilizador se deseja continuar ou parar a corrida ali, parando a execução do programa, como a paragem do programa, neste ponto, não foi “programada”, o programa continuará, independentemente da resposta dada, voltando portanto a repetir todos os passos acima descritos até o número de corridas ter chegado ao fim;

No final, será mostrada a tabela de classificações finais de campeonato e, após isso, o utilizador voltará ao menu principal.

Caso escolha a opção 5

O programa pedirá o id do piloto que será penalizado;

Depois pedirá também o valor da penalização, este passo será repetido até que o valor obtido do utilizador se encontre entre 1 e 3;

De seguida, o programa irá indicar se encontrou o piloto com esse id, sendo que nesse caso serão mostrados alguns dados do piloto, ou, então será indicado ao utilizador que não existe nenhum piloto com o id indicado;

Por fim, o programa voltará ao menu principal.

Caso escolha a opção 0

O programa guardará toda a informação dos pilotos e carros que tem nos devidos ficheiros (“pilotos.txt” e “carros.txt”, respetivamente) e encerrará.

Principais funções

faz_voltas

Recebe

Um ponteiro para o array com todos os pares que irão participar (“par”) e o seu tamanho (“tp”);

Um ponteiro para o início do array que contém todos os pilotos (“p”) e o seu tamanho (“Tp”).

Procedimento

Pede ao utilizador o número de voltas e o tamanho da pista;

Cria uma função “for” que vai contando o número de voltas.

Dentro dessa função aloca memória para cada uma das voltas.

A cada volta que vai fazendo aloca também memória para cada uma das simulações dos carros.

À medida que vai simulando cada volta de cada par piloto-carro, a função vai colocando esse par na estrutura dinâmica de maneira ordenada. O melhor classificado da volta acaba por ser o que está ligado à estrutura primeiramente alocada para a volta em questão. A ordem de colocação na estrutura é do melhor classificado para o pior, sendo que os pares que sofreram acidente ficam no fim, sem colocação diferenciada entre si.

No final de todas as simulações e ordenações dessa volta, a função ainda mostra as classificações de cada par da mesma maneira como estão ordenados na estrutura dinâmica.

No final esta função chamará ainda uma função que irá mostrar o “menu de revisão de resultados”, menu esse onde o utilizador poderá escolher rever todos os resultados que desejar.

cria_par

Recebe

- O número de pares que vão ser criados (“tp_final”);
- Um ponteiro para o array com todos os pilotos (“p”) e o seu tamanho (“Tp”);
- Um ponteiro para o array com todos os carros (“c”) e o seu tamanho (“Tc”).

Procedimento

Começa por alocar um array do tipo “par” que irá ser preenchido com informação de todos os pares que serão feitos;

Aloca dois arrays (“buff_carros” e “buff_pilotos”), inicializando-os completamente a zero e que terão como objetivo guardar os ids dos carros/pilotos já escolhidos.

Cria, utilizando a função “intUniformRnd” do ficheiro “utlis.c”, recebe um valor que será usado como índice para busca de um carro/piloto no array de pilotos/carros. Estes mesmos carros/pilotos só serão usados caso estejam aptos a participar na corrida e não tenha já sido escolhidos. Caso esse carro/piloto esteja disponível, o seu id será guardado no array buff_carros/buff_pilotos , o “contador” do respectivo array será aumentado em um (o contador foi iniciado com o valor zero, e representa o número de carros/pilotos já escolhidos) e a sua parte do array de pares será preenchida com a informação relevante.

Após todo este processo de criação de pares será mostrado no ecrã todos os pares que foram escolhidos, assim como todos os carros e pilotos que não foram e uma possível justificação para a sua não participação.

Por fim, liberta os arrays que não serão mais utilizados, “buff_pares” e “buff_pilotos”,