

FACULDADE DE CIÊNCIAS EXATAS E DA ENGENHARIA

Licenciatura em Engenharia Informática

3º Ano

Sistemas Operativos

Projeto Prático 2017/2018

**Simulação Feira com Montanha Russa**

Docentes:

Prof. Eduardo Marques

Prof. Luís Gaspar

Discentes:

Eurico Teixeira nº 2083014

Josué Ferreira nº 2066513

Paulo Gil de Freitas nº 2012413

Funchal, 4 janeiro de 2018

Índice

[Introdução 3](#_Toc503011571)

[Funcionamento da Feira 4](#_Toc503011572)

[Feira 4](#_Toc503011573)

[Bilheteira 4](#_Toc503011574)

[Bar 4](#_Toc503011575)

[Montanha-Russa 4](#_Toc503011576)

[Carrossel 4](#_Toc503011577)

[Fluxogramas 6](#_Toc503011578)

[Implementação 10](#_Toc503011579)

[Monitor 10](#_Toc503011580)

[Simulador 10](#_Toc503011581)

[Mensagem 10](#_Toc503011582)

[Ficheiro de Configuração 12](#_Toc503011583)

[Critério de Paragem 12](#_Toc503011584)

[Thread 12](#_Toc503011585)

[Semáforos/Trincos 13](#_Toc503011586)

[Testes 14](#_Toc503011587)

[Conclusão 15](#_Toc503011588)

[Anexos 16](#_Toc503011589)

[Monitor 16](#_Toc503011590)

[Simulador 68](#_Toc503011591)

[Unix 110](#_Toc503011592)

# Introdução

O relatório presente foi elaborado para a unidade curricular de Sistemas Operativos, inserida na licenciatura em Engenharia Informática na Universidade da Madeira.

O propósito deste relatório é a melhor explicação do processo de desenvolvimento que existiu no desenrolar do projeto proposto. Dessa forma este apresenta-se dividido em secções que ajudam a entender o processo e os seus meandros.

Para uma contextualização temos que todo o relatório é representativo das fases anteriores e que a linguagem trabalhada foi em C.

Temos então que na 1ª fase foram inseridas as bibliotecas para a gestão da informação nos ficheiros de texto e o posterior carregamento de parâmetros pelo Monitor e Simulador; 2ª fase tivemos a implementação das bibliotecas anteriormente mencionadas para a comunicação entre o Simulador e o Monitor, criação de um esboço da interface e ainda a elaboração de um relatório percorrendo as funcionalidades a implementar e como seria feito a sincronização; por fim na 3ª fase temos o aglomerar das fases anteriores culminando na explicação final onde surgem todos os elementos e mecanismos e politicas de sincronização.

# Funcionamento da Feira

Para um melhor entendimento, do que abaixo vamos explicar, dizemos aqui que existem três tipos distintos de pessoas, os que apresentam mobilidade reduzida, as crianças e os adultos ditos publico geral.

## Feira

### Bilheteira

A Feira inicia-se com a entrada para a fila das bilheteiras, aqui temos que esta apresenta condicionantes à entrada, as pessoas com mobilidade reduzida têm prioridade de entrada em relação à restante população, sendo que para este caso tanto crianças como adultos são atendidos pela ordem de chegada dos mesmos.

Apos terem passado a fila da bilheteira encontramos o interior da feira, onde existe um bar, uma montanha russa e um carrossel. A ordem pela qual são percorridas as atrações da feira é aleatória sendo pura coincidência que duas pessoas consecutivas façam o mesmo percurso.

### Bar

No caso de chegar ao bar temos, de forma semelhante à bilheteira uma fila de entrada com prioridades, só que nestes casos existe a diferenciação entre crianças e adultos, sendo que a ordem de atendimento final é primeiro a pessoas com mobilidade reduzida, posteriormente os adultos e por fim as crianças.

### Montanha-Russa

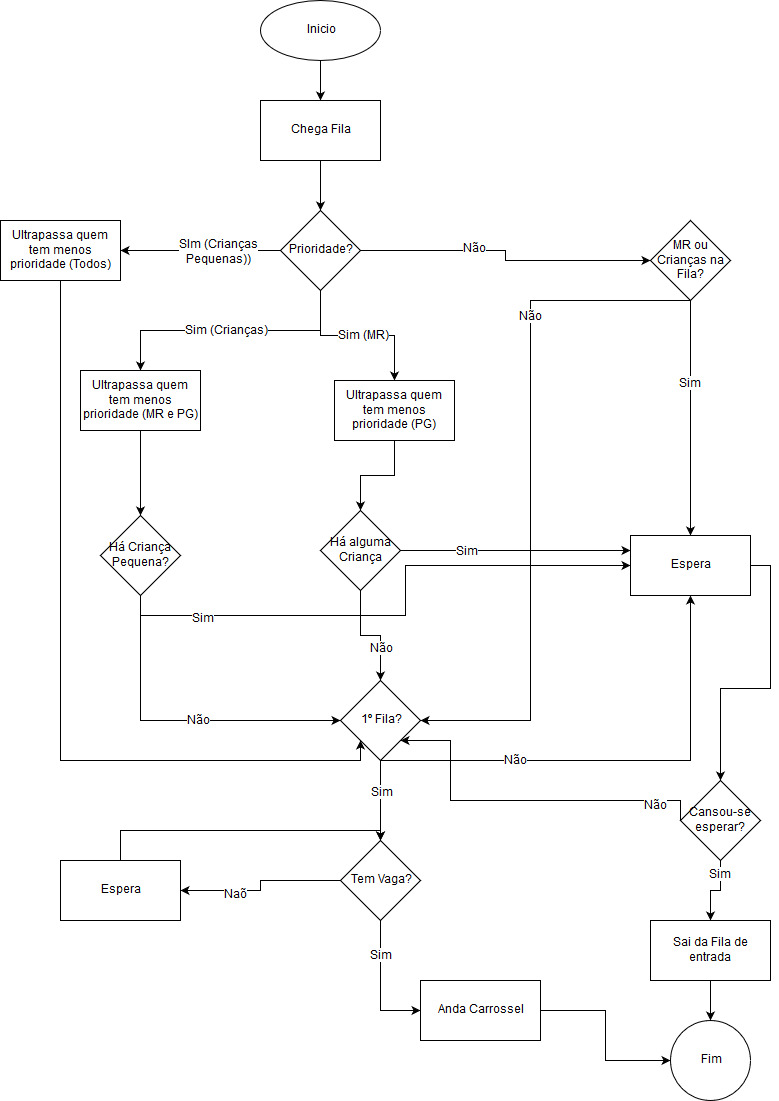
A montanha-russa contrariamente a tudo o descrito até agora não permite o acesso a todas as pessoas, o acesso encontra-se restrito a pessoas com mobilidade reduzida uma vez que estas poderiam apresentar um maior risco de acidente. Desta forma temos que o acesso se efetua conforme a ordem de chegada, sendo que a politica de acesso se centra na especificação de apenas poderem andar nesta atração aos pares, ou seja a atração apenas funciona se estiverem duas pessoas prontas para andar, caso contrario uma pessoa tem de aguardar que outra chegue.

### Carrossel

Nesta atração ao contrario do que se verifica nas restantes filas de espera, onde são os próprios elementos a se orientarem e colocarem em fila segundo o seu tipo de utilizador temos um elemento dedicado a tal tarefa, o Gestor do Carrossel que encaminha cada pessoa para a posição que esta deve de ocupar. Na sua orientação este encaminha para o inicio da fila os mais jovens, sendo que neste caso passam a existir 4 tipos de clientes, as crianças mais jovens (2-10), e as mais velhas (11-17), existindo na mesma os adultos e as pessoas com mobilidade reduzida. Assim sendo a ordem de entrada é de crianças jovens, crianças mais velhas, pessoas com mobilidade reduzida e finalmente os adultos.

## Fluxogramas

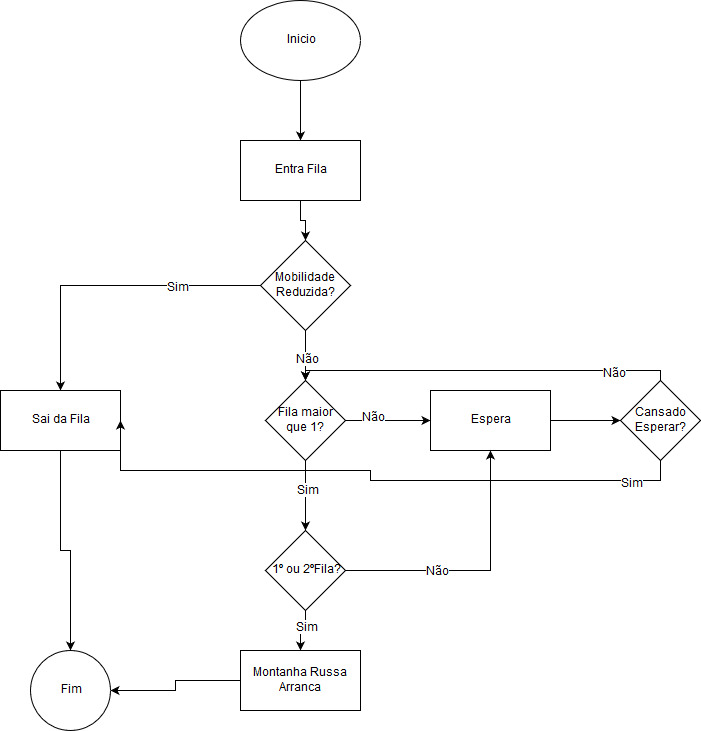
Fluxograma 1 - Bilheteira



Fluxograma 2 - Carrossel



Fluxograma 3 - Bar



Fluxograma 4 - Montanha Russa

# Implementação

## Monitor

O monitor como o próprio nome indica é um elemento de grande importância para o desenvolvimento deste projeto uma vez que é este que possibilita a observação dos eventos que se encontram a decorrer, mais ainda dado que é ele que também deslinda os dados obtidos durante a simulação.

Além dos elementos apresentados anteriormente existe também a função para efetuar a conversão do tempo de unidades de simulação para tempo real e ainda métodos para escrita das estatísticas e dos eventos em ficheiros, todas as estatísticas das simulações e a interface que observamos durante a simulação.

Como forma de comunicação encontramos aqui métodos correlacionados com a *socket* e com a visualização dos dados. Sendo este o meio escolhido para se fazer a comunicação, recebendo os dados do simulador concernentes aos clientes.

## Simulador

O simulador é grande parte do cérebro existente no projeto uma vez que é este que controla diversos aspetos da coordenação e sincronização dos clientes, uma vez que as politicas de acesso encontram-se maioritariamente neste elemento do projeto.

A sincronização e a coordenação encontram-se controladas a partir de mensagens enviadas pela socket, alem deste elemento temos ainda que é aqui que são criadas as pessoas que irão popular a Feira.

Encontramos ainda que é neste elemento que são lidas as configurações dadas passadas do ficheiro de configuração, tornando a feira mais próxima daquilo que o utilizador espera.

## Mensagem

Enquanto a simulação decorre, são enviadas mensagem em cada evento (descrevendo-o e dizendo que é que interveio), sendo que todos os parâmetros existentes são enviados, no entanto os únicos parâmetros que são aproveitados são os necessários, sendo os restantes descartados.

Assim temos que cada mensagem contém os seguintes atributos, pela ordem descrita:

1. ID Cliente
2. Tipo Cliente
3. Evento
4. Hora Atual da Simulação
5. Idade do Cliente
6. Hora de Entrada na Fila do Bar
7. Hora de Desistência da Fila do Bar
8. Hora de Atendimento no Bar
9. Hora Entrada na Fila do Carrossel
10. Hora Desistência da Fila do Carrossel
11. Hora Entrada no Carrossel
12. Hora Entrada na Fila da Montanha Russa
13. Hora Desistência da Fila da Montanha Russa
14. Hora Andou na Montanha Russa
15. Hora de Entrada na Fila das Bilheteiras
16. Hora de Desistência da Fila das Bilheteiras
17. Hora Entrada na Feira
18. Hora Saída da Feira

## Ficheiro de Configuração

Este ficheiro serve como um intermediário entre o que utilizador pretende e o que a simulação vai desenvolver, permitindo assim que exista um equilíbrio entre ambos.

Todos os parâmetros pedidos devem de ser preenchidos, caso contrario a simulação não irá correr.

Neste ficheiro encontramos os seguintes elementos:

1. Tempo Médio de Chegadas
2. Tempo de Simulação
3. Tamanho Máximo do Bar
4. Tamanho Máximo do Carrossel
5. Tamanho Máximo da Feira
6. Tempo Máximo de Espera nas Filas
7. Percentagem Ser Cliente Criança
8. Percentagem Ser Cliente Mobilidade Reduzida
9. Número de Clientes a Criar
10. Tempo Médio Que Cliente Fica nas Zonas
11. Percentagem de Ficar na Esplanada do Bar
12. Tempo Médio de Estadia do Cliente na Feira

## Critério de Paragem

Existem diversos critérios que permitem que a simulação termine, entre eles podemos descrever o Tempo de Simulação (que quando atingido param de ser criadas pessoas), o Tempo de Estadia, o Número de Cliente a Criar e por fim o Tempo Máximo de Espera nas Filas.

## Thread

No desenvolvimento deste projeto implementamos quatro threads distintas entre si, que possibilitaram uma melhor gestão dos diferentes elementos.

No Monitor temos uma thread que se encarrega de receber todas as mensagens enviadas a partir do Simulador, Thread\_Recebe\_Mensagens. A Thread\_Cliente[],a thread do relógio da simulação Thread\_Ciclo\_Tempo\_Simulacao e finalmente a do gestor do carrossel Thread\_Gestor\_Carroussel.

## Semáforos/Trincos

Todas as variáveis globais foram protegidas utilizando trincos, impedindo que caso surgisse mais que uma tarefa em simultâneo esta fosse acedida simultaneamente.

Cada zona criada teve direito a um semáforo que tem em consideração os tamanhos máximos definidos para aquela determinada zona. Recorrendo à definição de semáforo temos que caso uma zona esteja na sua capacidade máxima e tentar aceder uma nova tarefa, esta acaba ficando bloqueada até que vague um novo lugar.

Além das zonas temos ainda que foram criados semáforos auxiliares, para ser possível conjugar aspetos que de outra forma seriam difíceis de controlar. Estes encontram-se no carrossel e são Espera\_Cliente, Espera\_Sinal\_Entrada e Semaforo\_So\_1\_Cliente. O primeiro, inicializado a 0, fica em espera no gestor, sendo assinalado pelo primeiro cliente que chega. Quando este semáforo é assinalado, o gestor prossegue com a sua rotina e verifica a lotação no espaço. Se não existir vaga, espera, caso contrário assinala o segundo semáforo, inicializado a 0, para indicar que o cliente pode frequentar a atracão. Por fim o último, inicializado a 1, funciona como trinco, garantindo que a interação gestor-cliente é feita a um cliente de cada vez.

# Testes

Todos os testes realizados tiveram por intuito a verificação da existência de problemas nas zonas críticas assinaladas por nós.

Verificando os resultados após os testes, chegamos à conclusão que estes coincidem com o esperado em situações extremas, sendo que as únicas variações derivam do facto de existirem no próprio programa situações propositadamente aleatórias.

# Conclusão

Nesta fase de desenvolvimento do projeto ocorreram diversos problemas relativamente ao proposto na fase anterior, sendo então reformuladas algumas ideias e apresentadas outras para a sua substituição.

Durante o processo de desenvolvimento uma das maiores apreensões existentes foi o rácio eficácia vs. eficiência. No momento da conclusão podemos afirmar que apesar dos empreendimentos para criar um equilíbrio entre ambas as partes, houve um enfoque ligeiramente superior na eficácia e em manter a justiça do uso dos recursos.

Em termos de limitações, aquela que realmente se encontra visível é relacionada com o número de threads máximas criadas, uma vez que este valor é estático e foi declarado como sendo de 10000. Assim sendo sabemos que o máximo de clientes a serem criados é o anteriormente dito.

Outro caso especifico é a necessidade de correr novamente o simulador, uma vez que acabada a primeira simulação, caso queiramos correr uma nova simulação.

Limitações ao nível da verificação dos elementos, quando inseridos do tipo correto, foi efetuada. Por outro lado, se num determinado campo for inserido um dado cujo tipo seja diferente do esperado, aí temos que não existe nenhuma forma de corrigir este erro.

Em suma, olhando ao proposto por nós enquanto discentes nas fases anteriores do trabalho e pelos docentes nas diferentes fases de avaliação podemos concluir que o desenvolvimento do projeto foi positivo, uma vez que todos os objetivos foram cumpridos.

# Anexos

## Monitor

#include "unix.h"

int sockfd;

int newsockfd;

pthread\_t Thread\_Recebe\_Mensagens;

//Variaveis a Receber do Simulador (Para a Funcao "Mostra\_Zonas\_Tempo\_Real"):

int Tamanho\_Maximo\_Bar=0;

int Tamanho\_Maximo\_Carroussel=0;

int Tamanho\_Maximo\_Feira=0;

//------------------------------------------------------------------------------------------------------------------

//Funcao Para Ler do Ficheiro Config:

void Ler\_Ficheiro\_Config2()

{

//Declaracao e Inicializacao de Variaveis Auxiliares da Funcao

char Le\_Linha[100];

int Valor\_Variavel\_Lida;

char Nome\_Variavel\_Lida[100];

char Var1[]="Tam\_Max\_Bar"; //"Tamanho\_Maximo\_Bar";

char Var2[]="Tam\_Max\_Carroussel"; //"Tamanho\_Maximo\_Carroussel";

char Var3[]="Tam\_Max\_Feira"; //"Tamanho\_Maximo\_Feira";

FILE \*fp;

fp=fopen("config.conf","r"); //Vai Abrir o Ficheiro Config.conf

if(fp!=NULL)

{

//Ciclo de Leitura:

while(fgets(Le\_Linha,100,fp)!=NULL)

{

/\* %s= Nome da Variavel | %i= Valor da Variavel \*/

sscanf(Le\_Linha, "%s = %i", Nome\_Variavel\_Lida, &Valor\_Variavel\_Lida);

/\* Strcmp Tem Valor 0 se [Elemento1]=[Elemento2] \*/

if(strcmp(Nome\_Variavel\_Lida,Var1)==0)

{

Tamanho\_Maximo\_Bar=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var2)==0)

{

Tamanho\_Maximo\_Carroussel=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var3)==0)

{

Tamanho\_Maximo\_Feira=Valor\_Variavel\_Lida;

}

}

fclose(fp);

}

else

{

printf("Erro Abertura Ficheiro");

//abort();

}

}

//Estatisticas:

//Total Clientes)

// |----------------------------------------------------------------------------|

// | Soma Tempos de Espera dos Clientes Antes de Entrarem e Antes de Desistirem |

// |----------------------------------------------------------------------------|

//Soma Tempos de Espera dos Clientes Em Cada Fila de Cada Zona Antes de ser Atendido/Entrar/Andar nas Zonas

int Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido;//=0;

int Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa;//=0;

int Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel;//=0;

int Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira;//=0;

//Soma Tempos de Espera dos Clientes Antes de Desistirem da Fila de Cada Zona

int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar;//=0;

int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa;//=0;

int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel;//=0;

int Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras;//=0;

// |------------------------------------------------------|

// | Mais Variaveis Auxiliares Para Tratamento dos Tempos |

// |------------------------------------------------------|

int Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira;

int Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Desistir;

int Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar;

int Tempo\_Espera\_Fila\_Bar\_Antes\_Desistir;

int Tempo\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel;

int Tempo\_Espera\_Fila\_Carroussel\_Antes\_Desistir;

int Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa;

int Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Desistir;

// |----------------|

// | Tempos Maximos |

// |----------------|

//Tempo Maximo de Espera dos Clientes Antes de Desistirem da Fila de Cada Zona

int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar;

int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa;

int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel;

int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras;

//Tempo Maximo de Espera dos Clientes Antes de Entrarem em Cada Zona

int Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar;

int Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa;

int Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel;

int Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira;

// |----------------------------------|

// | Mais Variaveis Para Estatisticas |

// |----------------------------------|

//Clientes Mobilidade Reduzida)

//Total de Clientes Mobilidade Reduzida que Entraram nas Filas

int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar;

int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel;

int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras;

//Total de Clientes Mobilidade Reduzida que Foram Atendidos/Entraram/Andaram nas Zonas

int Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar;

int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel;

int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira;

//Total de Clientes Mobilidade Reduzida que Desistiram

int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar;

int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel;

int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras;

//Clientes Crianças)

//Total de Clientes Crianças que Entraram nas Filas

int Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar;

int Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa;

int Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel;

int Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras;

//Total de Clientes Crianças que Foram Atendidos/Entraram/Andaram nas Zonas

int Total\_Clientes\_Criancas\_Atendidos\_Bar;

int Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa;

int Total\_Clientes\_Criancas\_Entraram\_Carroussel;

int Total\_Clientes\_Criancas\_Entraram\_Feira;

//Total de Clientes Crianças que Desistiram

int Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar;

int Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa;

int Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel;

int Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras;

//Clientes Publico Geral)

//Total de Clientes Publico Geral que Entraram nas Filas

int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar;

int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa;

int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel;

int Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras;

//Total de Clientes Publico Geral que Foram Atendidos/Entraram/Andaram nas Zonas

int Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar;

int Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa;

int Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel;

int Total\_Clientes\_Publico\_Geral\_Entraram\_Feira;

//Total de Clientes Publico Geral que Desistiram

int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar;

int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa;

int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel;

int Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras;

//---------------------------------------------------------------------------------------------------------------------------------

// |-------------------------------------------|

// | Variaveis Para Estatisticas em Tempo Real |

// |-------------------------------------------|

//Total de Clientes ACTUALMENTE nas Filas

int Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar;

int Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa;

int Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel;

int Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras;

//Total de Clientes ACTUALMENTE nas Zonas

int Total\_Clientes\_Acutalmente\_No\_Bar;

int Total\_Clientes\_Acutalmente\_No\_Carroussel;

int Total\_Clientes\_Acutalmente\_Na\_Feira;

// |--------------------------------------|

// | Variaveis Tempo Medio Estadia Feira |

// |--------------------------------------|

int Soma\_Tempos\_Estadia\_Feira;

int Total\_Clientes\_Sairam\_Feira;

//---------------------------------------------------------------------------------------------------------------------------------

//Variaveis da Simulacao:

bool Simulacao\_Em\_Curso=false; //Variavel Vai Indicar que Esta Uma Simulacao em Curso (Se For True Podemos Activar a Ocupacao das Zonas em Tempo Real)

int Escolha\_Menu=0; //Variavel Que Guarda a Escolha do Utilizador. Quando É Igual a 7, Aplicacao Fecha. Tem de Ser Inicializado a 0 Por Causa da Funcao "Escreve\_Cabecalho(int Escolha\_Menu)"

//---------------------------------------------------------------------------------------------------------------------------------

// Cria Cabeçalho do Interface do Monitor

void Escreve\_Cabecalho(int Escolha\_Menu)

{

system("clear");

printf(" ┌──────────────────────────────────────────────────────────────┐\n");

printf(" │ Sistemas Operativos 2017/2018 ─ Simulacao Feira │\n");

printf(" └──────────────────────────────────────────────────────────────┘\n");

printf("┌─────────────────────────────────────────┌────────────────────────────────────────────────────────────────────┐\n");

printf("│ │ ■ 1-Inicio Simulacao ■ 2-Estatisticas Tempo Real │\n");

switch(Escolha\_Menu)

{

case 0: printf("│ Menu Principal │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

case 1: printf("│ Inicio Simulacao │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

case 2: printf("│ Estatisticas Tempo Real │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

case 3: printf("│ Eventos Tempo Real │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

case 4: printf("│ Estado Zonas Tempo Real │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

case 5: printf("│ Sobre │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

case 6: printf("│ Ajuda │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

case 7: printf("│ Sair │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

default: printf("│ Erro: Opcao Errada │ ■ 3-Eventos Tempo Real ■ 4-Estado Zonas Tempo Real │\n"); break;

}

printf("│ │ ■ 5-Sobre ■ 6-Ajuda ■ 7-Sair │\n");

printf("└─────────────────────────────────────────┘────────────────────────────────────────────────────────────────────┘\n");

}

//Funcao Que Coloca Linhas em Branco Para Manter Efeito visual

void Coloca\_Linhas\_Em\_Branco(int Quantidade\_Linhas)

{

int i=0;

for(i=0;i<=Quantidade\_Linhas;i++)

{

if(i==Quantidade\_Linhas)

{

printf("└──────────────────────────────────────────────────────────────────────────────────────────────────────────────┘\n");

}

else

{

printf("│ │\n");

}

}

}

//---------------------------------------------------------------------------------------------------------------------------------

//Auxiliar da Sub Funcao de Ocupcao das Zonas, Mas Que Trata Apenas das Filas

void Preenche\_Ocupacao\_Filas(int Total\_Clientes\_Actual, char\* Zona)

{

//printf("\n");

printf("┌─────────────────────────┐\n");

printf("%s │ =%d Clientes \n", Zona, Total\_Clientes\_Actual);

printf("└─────────────────────────┘\n");

}

//Auxiliar da Sub Funcao de Ocupacao das Zonas. Trata da Parte Estetica. Basicamente a Operacao "Trata\_Ocupacao\_Zonas\_Tempo\_Real" Vai Descodificar o Tipo de Zona Recebeido em Int e Vai Traduzir Para String. Esta Funcao "Preenche\_Ocupacao" Vai Receber os Dados Numericos e a String com o Nome da Zona e Vai Mostrar a Ocupacao Dessa Zona

void Preenche\_Ocupacao(int Tamanho\_Maximo\_Zona, int Total\_Clientes\_Actual, char\* Zona)

{

float Ocupacao\_Actual;

//Salvaguarda Contra Tamanho das Zonas Ser ==0 <=> Erro De Divisao Por 0

if(Tamanho\_Maximo\_Zona==0)

{

Ocupacao\_Actual=0;

}

else

{

Ocupacao\_Actual=Total\_Clientes\_Actual\*100/(float)Tamanho\_Maximo\_Zona;

}

//Se Nao Pusermos (float) no Denominador, Ele Faz a Divisao, Mas Apenas Mostra a Parte Inteira (O Que Vem Antes da Virgula)

if(Ocupacao\_Actual==0)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>0 && Ocupacao\_Actual<=5)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>5 && Ocupacao\_Actual<=10)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>10 && Ocupacao\_Actual<=15)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>15 && Ocupacao\_Actual<=20)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>20 && Ocupacao\_Actual<=25)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>25 && Ocupacao\_Actual<=30)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>30 && Ocupacao\_Actual<=35)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>35 && Ocupacao\_Actual<=40)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>40 && Ocupacao\_Actual<=45)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>45 && Ocupacao\_Actual<=50)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>50 && Ocupacao\_Actual<=55)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>55 && Ocupacao\_Actual<=60)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>60 && Ocupacao\_Actual<=65)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>65 && Ocupacao\_Actual<=70)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>70 && Ocupacao\_Actual<=75)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>75 && Ocupacao\_Actual<=80)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>80 && Ocupacao\_Actual<=85)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>85 && Ocupacao\_Actual<=90)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>90 && Ocupacao\_Actual<=95)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

else if(Ocupacao\_Actual>95 && Ocupacao\_Actual<=100)

{

printf("┌─────────────────────────┐───────────────────────────────────────────────────────────┐\n");

printf("%s │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │■ │ (%.1f%% =%d Clientes|Max=%d)\n", Zona, Ocupacao\_Actual,Total\_Clientes\_Actual,Tamanho\_Maximo\_Zona);

printf("└─────────────────────────┘───────────────────────────────────────────────────────────┘\n");

}

}

//Subfuncao Para Ajudar a Representar a Ocupacao das Zonas

void Trata\_Ocupacao\_Zonas\_Tempo\_Real(int Tamanho\_Maximo\_Zona\_Recebido, int Total\_Clientes\_Actual\_Recebido, int Tipo\_Zona\_Recebido)

{

char Zona1[]="│ Lotacao Sala Bar ";

char Zona2[]="│ Lotacao Feira ";

char Zona3[]="│ Lotacao Carroussel ";

char Zona4[]="│ Lotacao Fila Bar ";

char Zona5[]="│ Lotacao Fila Montanha Russa ";

char Zona6[]="│ Lotacao Fila Feira ";

char Zona7[]="│ Lotacao Fila Carroussel ";

switch(Tipo\_Zona\_Recebido)

{

case 1: //1=Lotacao Bar

Preenche\_Ocupacao(Tamanho\_Maximo\_Zona\_Recebido, Total\_Clientes\_Actual\_Recebido, Zona1);

break;

//---------------------------------------------------------------------------------------

case 2: //2=Lotacao Feira

Preenche\_Ocupacao(Tamanho\_Maximo\_Zona\_Recebido, Total\_Clientes\_Actual\_Recebido, Zona2);

break;

//---------------------------------------------------------------------------------------

case 3: //3=Lotacao Carroussel

Preenche\_Ocupacao(Tamanho\_Maximo\_Zona\_Recebido, Total\_Clientes\_Actual\_Recebido, Zona3);

break;

//---------------------------------------------------------------------------------------

case 4: //4=Lotacao Fila Bar

Preenche\_Ocupacao\_Filas(Total\_Clientes\_Actual\_Recebido, Zona4);

break;

//---------------------------------------------------------------------------------------

case 5: //5=Lotacao Fila Montanha Russa

Preenche\_Ocupacao\_Filas(Total\_Clientes\_Actual\_Recebido, Zona5);

break;

//---------------------------------------------------------------------------------------

case 6: //6=Lotacao Fila Feira

Preenche\_Ocupacao\_Filas(Total\_Clientes\_Actual\_Recebido, Zona6);

break;

//---------------------------------------------------------------------------------------

case 7: //7=Lotacao Fila Carroussel

Preenche\_Ocupacao\_Filas(Total\_Clientes\_Actual\_Recebido, Zona7);

break;

//---------------------------------------------------------------------------------------

default:

printf("Erro: Tipo de Zona Incorrecto \n");

}

}

//Funcao que Vai Mostrar a Ocupacao das Zonas em Tempo Real

void Mostra\_Zonas\_Tempo\_Real(int Tamanho\_Maximo\_Bar,int Tamanho\_Maximo\_Carroussel,int Tamanho\_Maximo\_Feira,int Total\_Clientes\_Acutalmente\_No\_Bar,int Total\_Clientes\_Acutalmente\_No\_Carroussel,int Total\_Clientes\_Acutalmente\_Na\_Feira,int Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar,int Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa,int Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel,int Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras)

{

Escreve\_Cabecalho(Escolha\_Menu);

printf("\n");

Trata\_Ocupacao\_Zonas\_Tempo\_Real(Tamanho\_Maximo\_Bar,Total\_Clientes\_Acutalmente\_No\_Bar,1);

Trata\_Ocupacao\_Zonas\_Tempo\_Real(Tamanho\_Maximo\_Feira,Total\_Clientes\_Acutalmente\_Na\_Feira,2);

Trata\_Ocupacao\_Zonas\_Tempo\_Real(Tamanho\_Maximo\_Carroussel,Total\_Clientes\_Acutalmente\_No\_Carroussel,3);

printf("\n");

//Filas Nao Tem Tamanho Maximo, Logo Vamos Deixar a 0

Trata\_Ocupacao\_Zonas\_Tempo\_Real(0,Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar,4);

Trata\_Ocupacao\_Zonas\_Tempo\_Real(0,Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa,5);

Trata\_Ocupacao\_Zonas\_Tempo\_Real(0,Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras,6);

Trata\_Ocupacao\_Zonas\_Tempo\_Real(0,Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel,7);

printf("\n");

printf("\n");

printf("\n");

}

//---------------------------------------------------------------------------------------------------------------------------------

//Funcao Que Converte "Tempo de Execucao" em "Tempo Real"

char \* Converte\_Tempo\_Para\_Horas(int Tempo)

{

static char String[8];

int Horas,Minutos,Segundos,Aux;

Horas=Tempo/3600;

Aux=Tempo%3600;

Minutos=Aux/60;

Segundos=Aux%60;

if(Tempo==0)

{

Horas=0;

Minutos=0;

sprintf(String,"%d:%d:%d",Horas,Minutos,Segundos);

}

else if(Tempo!=0)

{

sprintf(String,"%d:%d:%d",Horas,Minutos,Segundos);

}

return String;

}

//---------------------------------------------------------------------------------------------------------------------------------

//Funcao Para Preparar Ficheiro do Log Eventos (Invocado Quando o Utilizador Escolhe a Opcao de Simulacao no Menu Principal):

void Prepara\_Ficheiro\_Eventos()

{

FILE \*fp;

fp=fopen("relatorio\_eventos.c","w");

fprintf(fp, "Eventos Feira: \n");

fprintf(fp, "\n");

fclose(fp);

}

//Funcao Para Escrever Eventos no Ficheiro Log (Precisamos da Funcao de "Preparacao" Anterior Pois Esta Funcao Vai Ser Invocada Sempre Que Houver um Evento. Dessa Forma Temos de Usar Append em Vez de Write, Senao Apagava Sempre o Conteudo do Ficheiro):

void Escreve\_Ficheiro\_Eventos(int Evento, int ID\_Cliente, int Horas\_Evento, int Tipo\_Cliente, int Idade)

{

char \*Nome\_Tipo\_Cliente;

char Tipo\_Cliente1[]="(MR)";

char Tipo\_Cliente2[]="(Cri)";

char Tipo\_Cliente3[]="(PG)";

if(Tipo\_Cliente==1)

{

Nome\_Tipo\_Cliente=Tipo\_Cliente1;

}

else if(Tipo\_Cliente==2)

{

Nome\_Tipo\_Cliente=Tipo\_Cliente2;

}

else if(Tipo\_Cliente==3)

{

Nome\_Tipo\_Cliente=Tipo\_Cliente3;

}

//ID's dos Clientes Começam no 0, Logo Temos de Por ID\_Cliente+1

FILE \*fp;

fp=fopen("relatorio\_eventos.c","a"); //"a" = Append (Ou Seja, a Linha é Adicionada ao Ficheiro Sem Eliminar Nada)

switch(Evento)

{

case 1:

fprintf(fp, "->Cliente Nº %i (%s) Chegou a Fila das Bilheteiras As: %s \n", ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 2:

fprintf(fp, "->Cliente Nº %i Entrou na Feira As: %s \n", ID\_Cliente+1,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 3:

fprintf(fp, "->Cliente Nº %i Desistiu da Fila das Bilheteiras As: %s \n", ID\_Cliente+1, Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 4:

fprintf(fp, "->Cliente Nº %i Chegou a Fila do Bar As: %s \n", ID\_Cliente+1,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 5:

fprintf(fp, "->Cliente Nº %i Foi Atendido no Bar As: %s e Saiu do Bar \n", ID\_Cliente+1, Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 6:

fprintf(fp, "->Cliente Nº %i Foi Atendido no Bar As: %s E Ficou no Bar \n", ID\_Cliente+1, Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 7:

fprintf(fp, "->Cliente Nº %i Desistiu da Fila do Bar As: %s \n", ID\_Cliente+1,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 8:

fprintf(fp, "->Cliente Nº %i Saiu da Sala do Bar As: %s \n", ID\_Cliente+1,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 9:

fprintf(fp, "->Cliente Nº %i (Idade=%d) Chegou a Fila do Carroussel As: %s \n", ID\_Cliente+1,Idade,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 10:

fprintf(fp, "->Cliente Nº %i (Idade=%d) Entrou no Carroussel As: %s \n", ID\_Cliente+1,Idade,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 11:

fprintf(fp, "->Cliente Nº %i (Idade=%d) Desistiu da Fila do Carroussel As: %s \n", ID\_Cliente+1,Idade,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 12:

fprintf(fp, "->Cliente Nº %i Chegou a Fila da Montanha Russa As: %s \n", ID\_Cliente+1,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 13:

fprintf(fp, "->Cliente Nº %i Andou na Montanha Russa As: %s \n", ID\_Cliente+1,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 14:

fprintf(fp, "->Cliente Nº %i Desistiu da Fila da Montanha Russa As: %s \n", ID\_Cliente+1, Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 15:

fprintf(fp, "->Cliente Nº %i Saiu da Feira As: %s \n", ID\_Cliente+1,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 16:

fprintf(fp, "->Cliente Nº %i (Idade=%d) Saiu do Carroussel As: %s \n", ID\_Cliente+1,Idade, Converte\_Tempo\_Para\_Horas(Horas\_Evento));

fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

case 50:

fprintf(fp, "\*Simulacao Terminada\* \n");

fprintf(fp, "\n");

//fprintf(fp, "-----------------------------------------------------\n");

break;

//---------------------------------------------------------------------------------------------------------------------------

default:

fprintf(fp, "-Erro: Evento Nao Definido\n");

fprintf(fp, "-----------------------------------------------------\n");

}

fclose(fp);

}

//---------------------------------------------------------------------------------------------------------------------------------

//Funcao que Mostra o Log de Eventos em Tempo Real

void Mostra\_No\_Monitor\_Eventos\_Tempo\_Real(int Evento, int ID\_Cliente, int Horas\_Evento, int Tipo\_Cliente, int Idade)

{

//☺=Chegou Fila

//☹=Desistiu Fila

//😃=Entrou/Atendido/Andou

char \*Nome\_Tipo\_Cliente;

char Tipo\_Cliente1[]="(MR)";

char Tipo\_Cliente2[]="(Cri)";

char Tipo\_Cliente3[]="(PG)";

if(Tipo\_Cliente==1)

{

Nome\_Tipo\_Cliente=Tipo\_Cliente1;

}

else if(Tipo\_Cliente==2)

{

Nome\_Tipo\_Cliente=Tipo\_Cliente2;

}

else if(Tipo\_Cliente==3)

{

Nome\_Tipo\_Cliente=Tipo\_Cliente3;

}

switch(Evento)

{

case 1:

printf("│ ☺ Cliente Nº%d %s Chegou a Fila das Bilheteiras As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 2:

printf("│ 😃 Cliente Nº%d %s Entrou na Feira As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 3:

printf("│ ☹ Cliente Nº%d %s Desistiu da Fila das Bilheteiras As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 4:

printf("│ ☺ Cliente Nº%d %s Chegou a Fila do Bar As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 5:

printf("│ 😃 Cliente Nº%d %s Foi Atendido no Bar As %s e Saiu do Bar \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 6:

printf("│ 😃 Cliente Nº%d %s Foi Atendido no Bar As %s e Ficou no Bar \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 7:

printf("│ ☹ Cliente Nº%d %s Desistiu da Fila do Bar As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 8:

printf("│ ☹ Cliente Nº%d %s Saiu da Sala do Bar As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 9:

printf("│ ☺ Cliente Nº%d %s (Idade=%d) Chegou a Fila do Carroussel As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Idade,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 10:

printf("│ 😃 Cliente Nº%d %s (Idade=%d) Entrou no Carroussel As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Idade,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 11:

printf("│ ☹ Cliente Nº%d %s (Idade=%d) Desistiu da Fila do Carroussel As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Idade,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 12:

printf("│ ☺ Cliente Nº%d %s Chegou a Fila da Montanha Russa As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 13:

printf("│ 😃 Cliente Nº%d %s Andou na Montanha Russa As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 14:

printf("│ ☹ Cliente Nº%d %s Desistiu da Fila da Montanha Russa As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 15:

printf("│ ☹ Cliente Nº%d %s Saiu da Feira As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 16:

printf("│ ☹ Cliente Nº%d %s (Idade=%d) Saiu do Carroussel As %s \n",ID\_Cliente+1,Nome\_Tipo\_Cliente,Idade,Converte\_Tempo\_Para\_Horas(Horas\_Evento));

break;

//---------------------------------------------------------------------------------------------------------

case 50:

printf("│ Simulacao Terminada! \n");

break;

//---------------------------------------------------------------------------------------------------------

default:

printf("│ Erro: Caso Nao Definido \n");

}

}

//---------------------------------------------------------------------------------------------------------------------------------

//Funcao Para Mostrar as Estatisticas da Feira no Monitor: (Dependendo dos Valores, os Floats Podem Ter Valores Muito Baixos Pois So Consideramos a Parte Inteira e Nao Ha Arredondamentos ao Exagero)

void Mostra\_Estatisticas\_Monitor(int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar,int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa,int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel,int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras,int Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar,int Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa,int Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel,int Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira,int Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido,int Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa,int Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel,int Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras,int Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira,int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar,int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel,int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras,int Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar,int Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa,int Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel,int Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras,int Total\_Clientes\_Criancas\_Atendidos\_Bar,int Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa,int Total\_Clientes\_Criancas\_Entraram\_Carroussel,int Total\_Clientes\_Criancas\_Entraram\_Feira,int Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar,int Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa,int Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel,int Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras,int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar,int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa,int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel,int Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras,int Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar,int Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa,int Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel,int Total\_Clientes\_Publico\_Geral\_Entraram\_Feira,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras,int Soma\_Tempos\_Estadia\_Feira,int Total\_Clientes\_Sairam\_Feira)

{

Escreve\_Cabecalho(Escolha\_Menu);

//---------------------------------------------------------------------------------------------------------------------------------

//1º Fazer Soma de Totais de Clientes dos Varios Tipos

//2º Fazer Medias de Tempos

//---------------------------------------------------------------------------------------------------------------------------------

//Soma Totais:

//Total de Clientes que Foram Atendidos/Entraram/Andaram nas Zonas

int Total\_Clientes\_Atendidos\_Bar=Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar+Total\_Clientes\_Criancas\_Atendidos\_Bar+Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar;

int Total\_Clientes\_Andaram\_MontanhaRussa=Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa+Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa;

int Total\_Clientes\_Entraram\_Carroussel=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel+Total\_Clientes\_Criancas\_Entraram\_Carroussel+Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel;

int Total\_Clientes\_Entraram\_Feira=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira+Total\_Clientes\_Criancas\_Entraram\_Feira+Total\_Clientes\_Publico\_Geral\_Entraram\_Feira;

//Total de Clientes que Entraram nas Filas

int Total\_Clientes\_Entraram\_Fila\_Bar=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar+Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar+Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar;

int Total\_Clientes\_Entraram\_Fila\_MontanhaRussa=Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa+Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa;

int Total\_Clientes\_Entraram\_Fila\_Carroussel=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel+Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel+Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel;

int Total\_Clientes\_Entraram\_Filas\_Bilheteiras=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras+Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras+Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras;

//Total de Clientes que Desistiram

int Total\_Clientes\_Desistiram\_Fila\_Bar=Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar+Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar+Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar;

int Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa=Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa+Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa;

int Total\_Clientes\_Desistiram\_Fila\_Carroussel=Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel+Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel+Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel;

int Total\_Clientes\_Desistiram\_Filas\_Bilheteiras=Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras+Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras+Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras;

//----------------------------------------------------------------------------------------------------------------------------------

//Tempos Medios:

//Tempos Medios nas Filas Antes Clientes Serem Atendidos/Entrarem/Andarem nas Zonas

float Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar=0;//=Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido/Total\_Clientes\_Atendidos\_Bar;

float Tempo\_Medio\_Antes\_Andar\_MontanhaRussa=0;

float Tempo\_Medio\_Antes\_Entrar\_Carroussel=0;

float Tempo\_Medio\_Antes\_Entrar\_Feira=0;

//Tempos Medios nas Filas Antes Clientes Desistirem das Filas das Zonas

float Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar=0;//=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar/Total\_Clientes\_Desistiram\_Fila\_Bar;

float Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa=0;

float Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel=0;

float Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras=0;

if(Total\_Clientes\_Atendidos\_Bar==0) {Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar=0;} else {Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar=Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido/Total\_Clientes\_Atendidos\_Bar;};

if(Total\_Clientes\_Andaram\_MontanhaRussa==0) {Tempo\_Medio\_Antes\_Andar\_MontanhaRussa=0;} else {Tempo\_Medio\_Antes\_Andar\_MontanhaRussa=Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa/Total\_Clientes\_Andaram\_MontanhaRussa;};

if(Total\_Clientes\_Entraram\_Carroussel==0) {Tempo\_Medio\_Antes\_Entrar\_Carroussel=0;} else {Tempo\_Medio\_Antes\_Entrar\_Carroussel=Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel/Total\_Clientes\_Entraram\_Carroussel;};

if(Total\_Clientes\_Entraram\_Feira==0) {Tempo\_Medio\_Antes\_Entrar\_Feira=0;} else {Tempo\_Medio\_Antes\_Entrar\_Feira=Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira/Total\_Clientes\_Entraram\_Feira;};

if(Total\_Clientes\_Desistiram\_Fila\_Bar==0) {Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar=0;} else {Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar/Total\_Clientes\_Desistiram\_Fila\_Bar;};

if(Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa==0) {Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa=0;} else {Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa/Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa;};

if(Total\_Clientes\_Desistiram\_Fila\_Carroussel==0) {Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel=0;} else {Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel/Total\_Clientes\_Desistiram\_Fila\_Carroussel;};

if(Total\_Clientes\_Desistiram\_Filas\_Bilheteiras==0) {Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras=0;} else {Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras=Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras/Total\_Clientes\_Desistiram\_Filas\_Bilheteiras;};

//Tempo Medio Estadia na Feira

float Tempo\_Medio\_Estadia\_Feira=0;

if(Total\_Clientes\_Sairam\_Feira==0) {Tempo\_Medio\_Estadia\_Feira=0;} else {Tempo\_Medio\_Estadia\_Feira=Soma\_Tempos\_Estadia\_Feira/Total\_Clientes\_Sairam\_Feira;};

//Incluir Um If Para Cada 1 Destas Variaveis de Tempos Medios. Se o Total de CLientes Atendidos ou que Desistiram For=0, o Tempo Medio=0 (Para Evitar Erro de Divisao Por 0)

//printf("─────────────────────────────[ Estatisticas ]─────────────────────────────\n");

printf("│ Total Clientes Atendidos Bar: %d \n", Total\_Clientes\_Atendidos\_Bar);

printf("│ Total Clientes Andaram Montanha Russa: %d \n",Total\_Clientes\_Andaram\_MontanhaRussa );

printf("│ Total Clientes Entraram Carroussel: %d \n",Total\_Clientes\_Entraram\_Carroussel );

printf("│ Total Clientes Entraram Feira: %d \n",Total\_Clientes\_Entraram\_Feira );

//Variavel de Controlo Para Ver Se Todos os Clientes Sairam da Feira

printf("│ Total Clientes Sairam Feira: %d \n",Total\_Clientes\_Sairam\_Feira );

printf("│ Total Clientes Entraram Fila Bar: %d \n", Total\_Clientes\_Entraram\_Fila\_Bar);

printf("│ Total Clientes Entraram Fila Montanha Russa: %d \n",Total\_Clientes\_Entraram\_Fila\_MontanhaRussa );

printf("│ Total Clientes Entraram Fila Carroussel: %d \n",Total\_Clientes\_Entraram\_Fila\_Carroussel );

printf("│ Total Clientes Entraram Fila Feira: %d \n",Total\_Clientes\_Entraram\_Filas\_Bilheteiras );

printf("│ Total Clientes Desistiram Fila Bar: %d \n", Total\_Clientes\_Desistiram\_Fila\_Bar);

printf("│ Total Clientes Desistiram Fila Montanha Russa: %d \n",Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa );

printf("│ Total Clientes Desistiram Fila Carroussel: %d \n",Total\_Clientes\_Desistiram\_Fila\_Carroussel );

printf("│ Total Clientes Desistiram Fila Feira: %d \n",Total\_Clientes\_Desistiram\_Filas\_Bilheteiras );

printf("│ Tempo Maximo Antes Entrar Feira: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira));

printf("│ Tempo Maximo Antes Desistir Fila Bilheteiras: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras));

printf("│ Tempo Medio Antes Ser Atendido Bar: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar));

printf("│ Tempo Medio Antes Andar Montanha: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Andar\_MontanhaRussa));

printf("│ Tempo Medio Antes Entrar Carroussel: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Entrar\_Carroussel));

printf("│ Tempo Medio Antes Entrar Feira: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Entrar\_Feira));

printf("│ Tempo Medio Antes Desistir Fila Bar: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar));

printf("│ Tempo Medio Antes Desistir Fila Montanha Russa: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa));

printf("│ Tempo Medio Antes Desistir Fila Carroussel: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel));

printf("│ Tempo Medio Antes Desistir Fila Bilheteiras: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras));

printf("│ Tempo Medio Estadia Feira: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Estadia\_Feira));

Coloca\_Linhas\_Em\_Branco(0);

}

//Funcao Para Escrever as Estatisticas da Ffeira no Ficheiro Estatisticas (So se Invoca no Final da Simulacao. Esta Funcao Nao Precisa de Uma Funcao "Preparatoria" Porque So Vai Ser Invocada Uma Vez No Final.):

void Escreve\_Estatisticas(int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar,int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa,int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel,int Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras,int Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar,int Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa,int Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel,int Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira,int Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido,int Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa,int Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel,int Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel,int Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras,int Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel,int Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira,int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar,int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel,int Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras,int Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar,int Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa,int Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel,int Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras,int Total\_Clientes\_Criancas\_Atendidos\_Bar,int Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa,int Total\_Clientes\_Criancas\_Entraram\_Carroussel,int Total\_Clientes\_Criancas\_Entraram\_Feira,int Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar,int Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa,int Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel,int Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras,int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar,int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa,int Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel,int Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras,int Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar,int Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa,int Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel,int Total\_Clientes\_Publico\_Geral\_Entraram\_Feira,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRussa,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel,int Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras,int Soma\_Tempos\_Estadia\_Feira,int Total\_Clientes\_Sairam\_Feira)

{

//---------------------------------------------------------------------------------------------------------------------------------

//1º Fazer Soma de Totais de Clientes dos Varios Tipos

//2º Fazer Medias de Tempos

//---------------------------------------------------------------------------------------------------------------------------------

//Soma Totais:

//Total de Clientes que Foram Atendidos/Entraram/Andaram nas Zonas

int Total\_Clientes\_Atendidos\_Bar=Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar+Total\_Clientes\_Criancas\_Atendidos\_Bar+Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar;

int Total\_Clientes\_Andaram\_MontanhaRussa=Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa+Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa;

int Total\_Clientes\_Entraram\_Carroussel=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel+Total\_Clientes\_Criancas\_Entraram\_Carroussel+Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel;

int Total\_Clientes\_Entraram\_Feira=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira+Total\_Clientes\_Criancas\_Entraram\_Feira+Total\_Clientes\_Publico\_Geral\_Entraram\_Feira;

//Total de Clientes que Entraram nas Filas

int Total\_Clientes\_Entraram\_Fila\_Bar=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar+Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar+Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar;

int Total\_Clientes\_Entraram\_Fila\_MontanhaRussa=Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa+Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa;

int Total\_Clientes\_Entraram\_Fila\_Carroussel=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel+Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel+Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel;

int Total\_Clientes\_Entraram\_Filas\_Bilheteiras=Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras+Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras+Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras;

//Total de Clientes que Desistiram

int Total\_Clientes\_Desistiram\_Fila\_Bar=Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar+Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar+Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar;

int Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa=Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa+Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa;

int Total\_Clientes\_Desistiram\_Fila\_Carroussel=Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel+Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel+Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel;

int Total\_Clientes\_Desistiram\_Filas\_Bilheteiras=Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras+Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras+Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras;

//---------------------------------------------------------------------------------------------------------------------------------

//Tempos Medios:

//Tempos Medios nas Filas Antes Clientes Serem Atendidos/Entrarem/Andarem nas Zonas

float Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar=0;//=Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido/Total\_Clientes\_Atendidos\_Bar;

float Tempo\_Medio\_Antes\_Andar\_MontanhaRussa=0;

float Tempo\_Medio\_Antes\_Entrar\_Carroussel=0;

float Tempo\_Medio\_Antes\_Entrar\_Feira=0;

//Tempos Medios nas Filas Antes Clientes Desistirem das Filas das Zonas

float Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar=0;//=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar/Total\_Clientes\_Desistiram\_Fila\_Bar;

float Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa=0;

float Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel=0;

float Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras=0;

if(Total\_Clientes\_Atendidos\_Bar==0) {Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar=0;} else {Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar=Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido/Total\_Clientes\_Atendidos\_Bar;};

if(Total\_Clientes\_Andaram\_MontanhaRussa==0) {Tempo\_Medio\_Antes\_Andar\_MontanhaRussa=0;} else {Tempo\_Medio\_Antes\_Andar\_MontanhaRussa=Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa/Total\_Clientes\_Andaram\_MontanhaRussa;};

if(Total\_Clientes\_Entraram\_Carroussel==0) {Tempo\_Medio\_Antes\_Entrar\_Carroussel=0;} else {Tempo\_Medio\_Antes\_Entrar\_Carroussel=Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel/Total\_Clientes\_Entraram\_Carroussel;};

if(Total\_Clientes\_Entraram\_Feira==0) {Tempo\_Medio\_Antes\_Entrar\_Feira=0;} else {Tempo\_Medio\_Antes\_Entrar\_Feira=Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira/Total\_Clientes\_Entraram\_Feira;};

if(Total\_Clientes\_Desistiram\_Fila\_Bar==0) {Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar=0;} else {Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar/Total\_Clientes\_Desistiram\_Fila\_Bar;};

if(Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa==0) {Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa=0;} else {Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa/Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa;};

if(Total\_Clientes\_Desistiram\_Fila\_Carroussel==0) {Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel=0;} else {Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel/Total\_Clientes\_Desistiram\_Fila\_Carroussel;};

if(Total\_Clientes\_Desistiram\_Filas\_Bilheteiras==0) {Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras=0;} else {Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras=Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras/Total\_Clientes\_Desistiram\_Filas\_Bilheteiras;};

//Tempo Medio Estadia na Feira

float Tempo\_Medio\_Estadia\_Feira=0;

if(Total\_Clientes\_Sairam\_Feira==0) {Tempo\_Medio\_Estadia\_Feira=0;} else {Tempo\_Medio\_Estadia\_Feira=Soma\_Tempos\_Estadia\_Feira/Total\_Clientes\_Sairam\_Feira;};

//Incluir Um If Para Cada 1 Destas Variaveis de Tempos Medios. Se o Total de CLientes Atendidos ou que Desistiram For=0, o Tempo Medio=0 (Para Evitar Erro de Divisao Por 0)

FILE \*fp;

fp=fopen("estatisticas.c","w");

fprintf(fp, "Estatisticas: \n");

fprintf(fp, "\n");

fprintf(fp, "|-------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Atendidos no Bar -----------------| \n");

fprintf(fp, "|-------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Atendidos Bar: %d \n", Total\_Clientes\_Atendidos\_Bar);

fprintf(fp, "Total Clientes Mobilidade Reduzida Atendidos Bar: %d \n", Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar);

fprintf(fp, "Total Clientes Crianças Atendidos Bar: %d \n", Total\_Clientes\_Criancas\_Atendidos\_Bar);

fprintf(fp, "Total Clientes Publico Geral Atendidos Bar: %d \n", Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar);

fprintf(fp, "\n");

fprintf(fp, "|-----------------------------------------------------------| \n");

fprintf(fp, "|-----------------------------------------------------------| \n");

fprintf(fp, "|-------- Clientes Andaram na Montanha Russa ---------------| \n");

fprintf(fp, "|-----------------------------------------------------------| \n");

fprintf(fp, "|-----------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Andaram Montanha Russa: %d \n",Total\_Clientes\_Andaram\_MontanhaRussa );

fprintf(fp, "Total Clientes Crianças Andaram Montanha Russa: %d \n",Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa );

fprintf(fp, "Total Clientes Publico Geral Andaram Montanha Russa: %d \n",Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa );

fprintf(fp, "\n");

fprintf(fp, "|----------------------------------------------------------| \n");

fprintf(fp, "|----------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Entraram no Carroussel---------------| \n");

fprintf(fp, "|----------------------------------------------------------| \n");

fprintf(fp, "|----------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Entraram Carroussel: %d \n",Total\_Clientes\_Entraram\_Carroussel );

fprintf(fp, "Total Clientes Mobilidade Reduzida Entraram Carroussel: %d \n",Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel );

fprintf(fp, "Total Clientes Crianças Entraram Carroussel: %d \n",Total\_Clientes\_Criancas\_Entraram\_Carroussel );

fprintf(fp, "Total Clientes Publico Geral Entraram Carroussel: %d \n",Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel );

fprintf(fp, "\n");

fprintf(fp, "|---------------------------------------------------------| \n");

fprintf(fp, "|---------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Entraram na Feira ------------------| \n");

fprintf(fp, "|---------------------------------------------------------| \n");

fprintf(fp, "|---------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Entraram Feira: %d \n",Total\_Clientes\_Entraram\_Feira );

fprintf(fp, "Total Clientes Mobilidade Reduzida Entraram Feira: %d \n",Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira );

fprintf(fp, "Total Clientes Crianças Entraram Feira: %d \n",Total\_Clientes\_Criancas\_Entraram\_Feira );

fprintf(fp, "Total Clientes Publico Geral Entraram Feira: %d \n",Total\_Clientes\_Publico\_Geral\_Entraram\_Feira );

fprintf(fp, "\n");

fprintf(fp, "|--------------------------------------------------------------| \n");

fprintf(fp, "|--------------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Entraram na Fila do Bar -----------------| \n");

fprintf(fp, "|--------------------------------------------------------------| \n");

fprintf(fp, "|--------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Entraram Fila Bar: %d \n", Total\_Clientes\_Entraram\_Fila\_Bar);

fprintf(fp, "Total Clientes Mobilidade Reduzida Entraram Fila Bar: %d \n", Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar);

fprintf(fp, "Total Clientes Crianças Entraram Fila Bar: %d \n", Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar);

fprintf(fp, "Total Clientes Publico Geral Entraram Fila Bar: %d \n", Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar);

fprintf(fp, "\n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "|-------- Clientes Entraram na Fila do Montanha Russa ---------------| \n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Entraram Fila Montanha Russa: %d \n",Total\_Clientes\_Entraram\_Fila\_MontanhaRussa );

fprintf(fp, "Total Clientes Crianças Entraram Fila Montanha Russa: %d \n",Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa );

fprintf(fp, "Total Clientes Publico Geral Entraram Fila Montanha Russa: %d \n",Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa );

fprintf(fp, "\n");

fprintf(fp, "|------------------------------------------------------------------| \n");

fprintf(fp, "|------------------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Entraram na Fila da Carroussel---------------| \n");

fprintf(fp, "|------------------------------------------------------------------| \n");

fprintf(fp, "|------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Entraram Fila Carroussel: %d \n",Total\_Clientes\_Entraram\_Fila\_Carroussel );

fprintf(fp, "Total Clientes Mobilidade Reduzida Entraram Fila Carroussel: %d \n",Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel );

fprintf(fp, "Total Clientes Crianças Entraram Fila Carroussel: %d \n",Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel );

fprintf(fp, "Total Clientes Publico Geral Entraram Fila Carroussel: %d \n",Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel );

fprintf(fp, "\n");

fprintf(fp, "|-----------------------------------------------------------------------| \n");

fprintf(fp, "|-----------------------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Entraram na Fila das Bilheteiras -----------------| \n");

fprintf(fp, "|-----------------------------------------------------------------------| \n");

fprintf(fp, "|-----------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Entraram Fila Bilheteiras: %d \n",Total\_Clientes\_Entraram\_Filas\_Bilheteiras );

fprintf(fp, "Total Clientes Mobilidade Reduzida Entraram Fila Bilheteiras: %d \n",Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras );

fprintf(fp, "Total Clientes Crianças Entraram Fila Bilheteiras: %d \n",Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras );

fprintf(fp, "Total Clientes Publico Geral Entraram Fila Bilheteiras: %d \n",Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras );

fprintf(fp, "\n");

fprintf(fp, "|----------------------------------------------------------------| \n");

fprintf(fp, "|----------------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Desistiram da Fila do Bar -----------------| \n");

fprintf(fp, "|----------------------------------------------------------------| \n");

fprintf(fp, "|----------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Desistiram Fila Bar: %d \n", Total\_Clientes\_Desistiram\_Fila\_Bar);

fprintf(fp, "Total Clientes Mobilidade Reduzida Desistiram Fila Bar: %d \n", Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar);

fprintf(fp, "Total Clientes Crianças Desistiram Fila Bar: %d \n", Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar);

fprintf(fp, "Total Clientes Publico Geral Desistiram Fila Bar: %d \n", Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar);

fprintf(fp, "\n");

fprintf(fp, "|----------------------------------------------------------------------| \n");

fprintf(fp, "|----------------------------------------------------------------------| \n");

fprintf(fp, "|--------- Clientes Desistiram da Fila do Montanha Russa --------------| \n");

fprintf(fp, "|----------------------------------------------------------------------| \n");

fprintf(fp, "|----------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Desistiram Fila Montanha Russa: %d \n",Total\_Clientes\_Desistiram\_Fila\_MontanhaRussa );

fprintf(fp, "Total Clientes Crianças Desistiram Fila Montanha Russa: %d \n",Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa );

fprintf(fp, "Total Clientes Publico Geral Desistiram Fila Montanha Russa: %d \n",Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa );

fprintf(fp, "\n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Desistiram da Fila da Carroussel---------------| \n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "|--------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Desistiram Fila Carroussel: %d \n",Total\_Clientes\_Desistiram\_Fila\_Carroussel );

fprintf(fp, "Total Clientes Mobilidade Reduzida Desistiram Fila Carroussel: %d \n",Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel );

fprintf(fp, "Total Clientes Crianças Desistiram Fila Carroussel: %d \n",Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel );

fprintf(fp, "Total Clientes Publico Geral Desistiram Fila Carroussel: %d \n",Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel );

fprintf(fp, "\n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|----------- Clientes Desistiram da Fila das Bilheteiras -----------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Total Clientes Desistiram Fila Bilheteiras: %d \n",Total\_Clientes\_Desistiram\_Filas\_Bilheteiras );

fprintf(fp, "Total Clientes Mobilidade Reduzida Desistiram Fila Bilheteiras: %d \n",Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras );

fprintf(fp, "Total Clientes Crianças Desistiram Fila Bilheteiras: %d \n",Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras );

fprintf(fp, "Total Clientes Publico Geral Desistiram Fila Bilheteiras: %d \n",Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras );

fprintf(fp, "\n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|---------------------------- Tempos Medios ------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Tempo Medio Antes Ser Atendido Bar: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Ser\_Atendido\_Bar));

fprintf(fp, "Tempo Medio Antes Andar Montanha Russa: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Andar\_MontanhaRussa));

fprintf(fp, "Tempo Medio Antes Entrar Carroussel: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Entrar\_Carroussel));

fprintf(fp, "Tempo Medio Antes Entrar Feira: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Entrar\_Feira));

fprintf(fp, "Tempo Medio Antes Desistir Fila Bar: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Fila\_Bar));

fprintf(fp, "Tempo Medio Antes Desistir Fila Montanha Russa: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Fila\_MontanhaRussa));

fprintf(fp, "Tempo Medio Antes Desistir Fila Carroussel: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Fila\_Carroussel));

fprintf(fp, "Tempo Medio Antes Desistir Fila Bilheteiras: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Antes\_Desistir\_Filas\_Bilheteiras));

fprintf(fp, "Tempo Medio Estadia Feira: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Medio\_Estadia\_Feira));

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|--------------------------- Tempos Maximos ------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "|-------------------------------------------------------------------------| \n");

fprintf(fp, "\n");

fprintf(fp, "Tempo Maximo Antes Ser Atendido Bar: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar));

fprintf(fp, "Tempo Maximo Antes Andar Montanha Russa: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa));

fprintf(fp, "Tempo Maximo Antes Entrar Carroussel: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel));

fprintf(fp, "Tempo Maximo Antes Entrar Feira: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira));

fprintf(fp, "Tempo Maximo Antes Desistir Fila Bar: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar));

fprintf(fp, "Tempo Maximo Antes Desistir Fila Montanha Russa: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa));

fprintf(fp, "Tempo Maximo Antes Desistir Fila Carroussel: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel));

fprintf(fp, "Tempo Maximo Antes Desistir Fila Bilheteiras: %s \n",Converte\_Tempo\_Para\_Horas(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras));

fprintf(fp, "\n");

fprintf(fp, "\*Estatisticas Terminadas\* \n");

fprintf(fp, "\n");

fclose(fp);

}

//---------------------------------------------------------------------------------------------------------------------------------

//Funcao Que Recebe Mensagem da Socket e Passa Cada Campo da Mensagem Para Uma Variavel Propria

int \* Recebe\_Mensagem(int sockfd)

{

int Mensagem\_Recebida=0;

int ID\_Cliente=0;

int Tipo\_Cliente=0;

int Estado=0;

int Hora\_Actual=0;

int Hora\_Entrada\_Fila\_Bar=0;

int Hora\_Desistencia\_Fila\_Bar=0;

int Hora\_Atendimento\_Bar=0;

int Hora\_Entrada\_Fila\_Carroussel=0;

int Hora\_Desistencia\_Fila\_Carroussel=0;

int Hora\_Entrada\_Carroussel=0;

int Hora\_Entrada\_Fila\_MontanhaRussa=0;

int Hora\_Desistencia\_Fila\_MontanhaRussa=0;

int Hora\_Andou\_MontanhaRussa=0;

int Hora\_Entrada\_Filas\_Bilheteiras=0;

int Hora\_Desistencia\_Filas\_Bilheteiras=0;

int Hora\_Entrada\_Feira=0;

int Hora\_Saida\_Feira=0;

int Idade=0;

char recvline[MAX\_LINE+1];

static int Mensagem[18]; //Conversao dos Valores da Mensagem Para Array a Ser Enviado Para Interpretacao

Mensagem\_Recebida=read(sockfd, recvline, MAX\_LINE);

if (Mensagem\_Recebida < 0) perror("Erro ao ler mensagem no servidor\n");

recvline[Mensagem\_Recebida] = 0;

sscanf(recvline,"%d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d", &ID\_Cliente, &Tipo\_Cliente, &Estado, &Hora\_Actual, &Idade, &Hora\_Entrada\_Fila\_Bar, &Hora\_Desistencia\_Fila\_Bar, &Hora\_Atendimento\_Bar, &Hora\_Entrada\_Fila\_Carroussel, &Hora\_Desistencia\_Fila\_Carroussel, &Hora\_Entrada\_Carroussel, &Hora\_Entrada\_Fila\_MontanhaRussa, &Hora\_Desistencia\_Fila\_MontanhaRussa, &Hora\_Andou\_MontanhaRussa, &Hora\_Entrada\_Filas\_Bilheteiras, &Hora\_Desistencia\_Filas\_Bilheteiras, &Hora\_Entrada\_Feira,&Hora\_Saida\_Feira);

Mensagem[0] = ID\_Cliente;

Mensagem[1] = Tipo\_Cliente;

Mensagem[2] = Estado;

Mensagem[3] = Hora\_Actual;

Mensagem[4] = Idade;

Mensagem[5] = Hora\_Entrada\_Fila\_Bar;

Mensagem[6] = Hora\_Desistencia\_Fila\_Bar;

Mensagem[7] = Hora\_Atendimento\_Bar;

Mensagem[8] = Hora\_Entrada\_Fila\_Carroussel;

Mensagem[9] = Hora\_Desistencia\_Fila\_Carroussel;

Mensagem[10] = Hora\_Entrada\_Carroussel;

Mensagem[11] = Hora\_Entrada\_Fila\_MontanhaRussa;

Mensagem[12] = Hora\_Desistencia\_Fila\_MontanhaRussa;

Mensagem[13] = Hora\_Andou\_MontanhaRussa;

Mensagem[14] = Hora\_Entrada\_Filas\_Bilheteiras;

Mensagem[15] = Hora\_Desistencia\_Filas\_Bilheteiras;

Mensagem[16] = Hora\_Entrada\_Feira;

Mensagem[17] = Hora\_Saida\_Feira;

return Mensagem;

}

//---------------------------------------------------------------------------------------------------------------------------------

//Funcao que Mostra Informacao do Grupo

void Mostra\_Informacao\_Grupo()

{

Escreve\_Cabecalho(Escolha\_Menu);

printf("│ │ Sistemas Operativos │ │\n");

printf("│ └────────────────────────────────┘ │\n");

printf("│ │\n");

printf("│ -Eurico Teixeira │\n");

printf("│ Nº 2083014 │\n");

printf("│ │\n");

printf("│ -Paulo Gil de Freitas │\n");

printf("│ Nº 2012413 │\n");

printf("│ │\n");

printf("│ -Josue Ferreira │\n");

printf("│ Nº 2066513 │\n");

Coloca\_Linhas\_Em\_Branco(13);

}

//Funcao Que Mostra Instrucoes de Funcionamento da Aplicacao

void Mostra\_Funcionamento()

{

Escreve\_Cabecalho(Escolha\_Menu);

printf("│ -Temos 7 Hipóteses que Podem Ser Escolhidas. │\n");

printf("│ -Para Podermos Ter Acesso aos Menus de Tempo Real É Necessario uma Simulação Activa. │\n");

printf("│ -É Criado Automaticamente o Log de Registos e as Estatisticas Finais Após a Realização de Uma Simulação │\n");

Coloca\_Linhas\_Em\_Branco(21);

}

//Funcao de Leitura e Interpretacao da Mensagem Recebida Pela Socket

void Interpreta\_Mensagem()

{

int\* Buffer=Recebe\_Mensagem(sockfd); //(newsockfd);

int ID\_Cliente=Buffer[0]; //Numero Processo

int Tipo\_Cliente=Buffer[1]; //Tipo Cliente= {Mobilidade Reduzida, Crianças, Publico Geral}

int Estado=Buffer[2]; //Ver Tabela de Estados

int Hora\_Actual=Buffer[3]; //Hora de um Acontecimento (Depende do Estado em que Esta e Esteve)

int Idade=Buffer[4]; //Idade do Cliente

int Hora\_Entrada\_Fila\_Bar=Buffer[5];

int Hora\_Desistencia\_Fila\_Bar=Buffer[6];

int Hora\_Atendimento\_Bar=Buffer[7];

int Hora\_Entrada\_Fila\_Carroussel=Buffer[8];

int Hora\_Desistencia\_Fila\_Carroussel=Buffer[9];

int Hora\_Entrada\_Carroussel=Buffer[10];

int Hora\_Entrada\_Fila\_MontanhaRussa=Buffer[11];

int Hora\_Desistencia\_Fila\_MontanhaRussa=Buffer[12];

int Hora\_Andou\_MontanhaRussa=Buffer[13];

int Hora\_Entrada\_Filas\_Bilheteiras=Buffer[14];

int Hora\_Desistencia\_Filas\_Bilheteiras=Buffer[15];

int Hora\_Entrada\_Feira=Buffer[16];

int Hora\_Saida\_Feira=Buffer[17];

switch(Estado) //Evento

{

case 1: //Entra Fila Bilheteiras da Feira

Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras++;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras++;

}

break;

//---------------------------------------------------------------------------------------

case 2: //Sai Fila Bilheteiras da Feira e Entra na Feira

Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras--;

Total\_Clientes\_Acutalmente\_Na\_Feira++;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Entraram\_Feira++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Entraram\_Feira++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira=Hora\_Entrada\_Feira-Hora\_Entrada\_Filas\_Bilheteiras;

if(Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira==0)

{

Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira=Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira;

}

else if(Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira>Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira)

{

Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira=Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira=Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira+Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira;

break;

//---------------------------------------------------------------------------------------

case 3: //Desiste Fila Bilheteiras da Feira

Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras--;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Desistir=Hora\_Desistencia\_Filas\_Bilheteiras-Hora\_Entrada\_Filas\_Bilheteiras;

if(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras==0)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras=Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Desistir;

}

else if(Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Desistir>Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras=Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Desistir;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras=Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras+Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Desistir;

break;

//---------------------------------------------------------------------------------------

case 4: //Entra Fila Bar

Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar++;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar++;

}

break;

//---------------------------------------------------------------------------------------

case 5: //Sai Fila Bar, é Atendido e Sai do Bar

Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar--;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Atendidos\_Bar++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar=Hora\_Atendimento\_Bar-Hora\_Entrada\_Fila\_Bar;

if(Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar==0)

{

Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar=Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar;

}

else if(Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar>Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar)

{

Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar=Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido=Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido+Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar;

break;

//---------------------------------------------------------------------------------------

case 6: //Fica no Bar//Sai Fila Bar, é Atendido e Fica no Bar

Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar--;

Total\_Clientes\_Acutalmente\_No\_Bar++;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Atendidos\_Bar++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar=Hora\_Atendimento\_Bar-Hora\_Entrada\_Fila\_Bar;

if(Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar==0)

{

Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar=Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar;

}

else if(Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar>Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar)

{

Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar=Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido=Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido+Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar;

break;

//---------------------------------------------------------------------------------------

case 7: //Desiste Fila Bar

Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar--;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Fila\_Bar\_Antes\_Desistir=Hora\_Desistencia\_Fila\_Bar-Hora\_Entrada\_Fila\_Bar;

if(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar==0)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar=Tempo\_Espera\_Fila\_Bar\_Antes\_Desistir;

}

else if(Tempo\_Espera\_Fila\_Bar\_Antes\_Desistir>Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar=Tempo\_Espera\_Fila\_Bar\_Antes\_Desistir;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar+Tempo\_Espera\_Fila\_Bar\_Antes\_Desistir;

break;

//---------------------------------------------------------------------------------------

case 8: //Sai Sala Bar

Total\_Clientes\_Acutalmente\_No\_Bar--;

break;

//---------------------------------------------------------------------------------------

case 9: //Entra Fila Carroussel

Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel++;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel++;

}

break;

//---------------------------------------------------------------------------------------

case 10: //Sai Fila Carroussel e Entra no Carroussel

Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel--;

Total\_Clientes\_Acutalmente\_No\_Carroussel++;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Entraram\_Carroussel++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel=Hora\_Entrada\_Carroussel-Hora\_Entrada\_Fila\_Carroussel;

if(Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel==0)

{

Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel=Tempo\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel;

}

else if(Tempo\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel>Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel)

{

Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel=Tempo\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel=Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel+Tempo\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel;

break;

//---------------------------------------------------------------------------------------

case 11: //Desiste Fila Carroussel

Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel--;

if(Tipo\_Cliente==1) //Mobilidade Reduzida

{

Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel++;

}

else if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Fila\_Carroussel\_Antes\_Desistir=Hora\_Desistencia\_Fila\_Carroussel-Hora\_Entrada\_Fila\_Carroussel;

if(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel==0)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel=Tempo\_Espera\_Fila\_Carroussel\_Antes\_Desistir;

}

else if(Tempo\_Espera\_Fila\_Carroussel\_Antes\_Desistir>Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel=Tempo\_Espera\_Fila\_Carroussel\_Antes\_Desistir;

}

//Variaveis Auxiliares Para Tempos Médios:

Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel+Tempo\_Espera\_Fila\_Carroussel\_Antes\_Desistir;

break;

//---------------------------------------------------------------------------------------

case 12: //Entra Fila Montanha Russa

Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa++;

if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa++;

}

break;

//---------------------------------------------------------------------------------------

case 13: //Sai Fila Montanha Russa e Anda na Montanha Russa

Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa--;

if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa=Hora\_Andou\_MontanhaRussa-Hora\_Entrada\_Fila\_MontanhaRussa;

if(Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa==0)

{

Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa=Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa;

}

else if(Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa>Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa)

{

Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa=Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa=Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa+Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa;

break;

//---------------------------------------------------------------------------------------

case 14: //Desiste Fila Montanha Russa

Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa--;

if(Tipo\_Cliente==2) //Criancas

{

Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa++;

}

else if(Tipo\_Cliente==3) //Publico Geral

{

Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa++;

}

//Tratamento dos Tempos Maximos:

Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Desistir=Hora\_Desistencia\_Fila\_MontanhaRussa-Hora\_Entrada\_Fila\_MontanhaRussa;

if(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa==0)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa=Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Desistir;

}

else if(Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Desistir>Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa)

{

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa=Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Desistir;

}

//Variaveis Para Tempos Médios:

Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa=Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa+Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Desistir;

break;

//---------------------------------------------------------------------------------------

case 15: //Sai Feira

Total\_Clientes\_Acutalmente\_Na\_Feira--;

Total\_Clientes\_Sairam\_Feira++;

//Variaveis Para Tempos Medios:

Soma\_Tempos\_Estadia\_Feira=Soma\_Tempos\_Estadia\_Feira+(Hora\_Saida\_Feira - Hora\_Entrada\_Feira); //Tempo\_Estadia\_Feira;

break;

//---------------------------------------------------------------------------------------

case 16: //Sai Carroussel

Total\_Clientes\_Acutalmente\_No\_Carroussel--;

break;

//---------------------------------------------------------------------------------------

case 50: //Fim Simulação

Simulacao\_Em\_Curso=false;

//Escreve Estatisticas Finais no Ficheiro Quando a Simulacao Acaba

Escreve\_Estatisticas(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar, Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido, Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa, Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel, Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar, Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Atendidos\_Bar, Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Feira, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar, Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar, Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Feira, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras, Soma\_Tempos\_Estadia\_Feira, Total\_Clientes\_Sairam\_Feira);

break;

//---------------------------------------------------------------------------------------

default: printf("Erro: Estado do Processo Errado. \n");

break;

}//Fim Switch

//Adiciona ao Ficheiro Eventos o Novo Evento que Aconteceu

Escreve\_Ficheiro\_Eventos(Estado, ID\_Cliente, Hora\_Actual,Tipo\_Cliente, Idade); //Funcao no Envia\_Para\_Ficheiros.c

//Estes If's Sao Para Actualizar o Ecra Dependendo da Hipotese Que o Utilizador Escolheu no Menu Principal

if(Escolha\_Menu==2)

{

Mostra\_Estatisticas\_Monitor(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar, Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido, Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa, Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel, Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar, Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Atendidos\_Bar, Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Feira, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar, Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar, Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Feira, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras, Soma\_Tempos\_Estadia\_Feira, Total\_Clientes\_Sairam\_Feira);

}

else if(Escolha\_Menu==3) //Isto é, Se o Menu Actual For o 3 (Mostrar Eventos em Tempo Real), Sempre Que Houver um Evento, Mostra no Ecra

{

Mostra\_No\_Monitor\_Eventos\_Tempo\_Real(Estado, ID\_Cliente, Hora\_Actual, Tipo\_Cliente, Idade);

}

else if(Escolha\_Menu==4) //Isto é, Se o Menu Actual For o 4 (Mostrar Zonas em Tempo Real), Sempre Que Houver uma Alteracao nas Zonas, Vai Ser Mostrada no Ecra

{

Mostra\_Zonas\_Tempo\_Real(Tamanho\_Maximo\_Bar, Tamanho\_Maximo\_Carroussel, Tamanho\_Maximo\_Feira, Total\_Clientes\_Acutalmente\_No\_Bar, Total\_Clientes\_Acutalmente\_No\_Carroussel, Total\_Clientes\_Acutalmente\_Na\_Feira, Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar, Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa, Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel, Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras);

}

}

//Funcao Que Poe o Valor das Variaveis Globais de Estatisticas no Seu Estado Original (=0) Para Permitir Varias Simulacoes Sem Ter de Correr a Aplicacao de Novo.

void Funcao\_Limpa\_Variaveis\_Estatisticas()

{

Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido=0;

Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa=0;

Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel=0;

Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira=0;

Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar=0;

Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa=0;

Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel=0;

Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras=0;

Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira=0;

Tempo\_Espera\_Filas\_Bilheteiras\_Antes\_Desistir=0;

Tempo\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido\_Bar=0;

Tempo\_Espera\_Fila\_Bar\_Antes\_Desistir=0;

Tempo\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel=0;

Tempo\_Espera\_Fila\_Carroussel\_Antes\_Desistir=0;

Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa=0;

Tempo\_Espera\_Fila\_MontanhaRussa\_Antes\_Desistir=0;

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar=0;

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa=0;

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel=0;

Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras=0;

Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar=0;

Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa=0;

Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel=0;

Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel=0;

Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras=0;

Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar=0;

Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa=0;

Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel=0;

Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras=0;

Total\_Clientes\_Criancas\_Atendidos\_Bar=0;

Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa=0;

Total\_Clientes\_Criancas\_Entraram\_Carroussel=0;

Total\_Clientes\_Criancas\_Entraram\_Feira=0;

Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar=0;

Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa=0;

Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel=0;

Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras=0;

Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar=0;

Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa=0;

Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel=0;

Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras=0;

Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar=0;

Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa=0;

Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel=0;

Total\_Clientes\_Publico\_Geral\_Entraram\_Feira=0;

Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar=0;

Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa=0;

Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel=0;

Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras=0;

Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar=0;

Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa=0;

Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel=0;

Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras=0;

Total\_Clientes\_Acutalmente\_No\_Bar=0;

Total\_Clientes\_Acutalmente\_No\_Carroussel=0;

Total\_Clientes\_Acutalmente\_Na\_Feira=0;

Soma\_Tempos\_Estadia\_Feira=0;

Total\_Clientes\_Sairam\_Feira=0;

}

//Teste Funcao da Thread Que Recebe Mensagens.

void Activa\_Ciclo\_Recebe\_Mensagens()

{

while(Simulacao\_Em\_Curso!=false)//==true)

{

Interpreta\_Mensagem();

}

}

//Menu Principal:

void Menu\_Principal()

{

//Escreve\_Cabecalho(Escolha\_Menu);

if(Escolha\_Menu!=3)

{

printf("Opcao="); //printf("$");

}

scanf("%d",&Escolha\_Menu);

switch(Escolha\_Menu)

{

case 1: //Iniciar Simulacao

if(Simulacao\_Em\_Curso==false) //Se Nao Houver Nenhuma Simulacao Activa, Vai Comecar Uma Simulacao Nova

{

Funcao\_Limpa\_Variaveis\_Estatisticas();

Prepara\_Ficheiro\_Eventos(); //Funcao Que Prepara o Ficheiro do Relatorio de Eventos (Limpa/Cria Ficheiro "Relatorio\_Eventos.c" e Poe Titulo)

Escreve\_Cabecalho(Escolha\_Menu);

Simulacao\_Em\_Curso=true;

Coloca\_Linhas\_Em\_Branco(24);

if(pthread\_create(&(Thread\_Recebe\_Mensagens), NULL ,(void \*)&Activa\_Ciclo\_Recebe\_Mensagens, NULL) != 0)

{

printf("Erro na criacao da tarefa\n");

exit(1);

}

}

else //Se Houver uma Simulacao Activa Devolve Erro, Pois So Podemos Executar Esta Opcao Se Nao Estiver Nenhuma Simulacao Activa

{

printf("Erro: Simulacao Activa Neste Momento \n");

}

break;

//---------------------------------------------------------------------------------------

case 2: //Estatisticas Tempo Real

if(Simulacao\_Em\_Curso==true)

{

//Funcao Para Mostrar as Estatisticas da Feira no Monitor:

Mostra\_Estatisticas\_Monitor(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar, Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido, Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa, Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel, Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar, Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Atendidos\_Bar, Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Feira, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar, Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar, Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Feira, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras, Soma\_Tempos\_Estadia\_Feira, Total\_Clientes\_Sairam\_Feira);

}

else

{

printf("Erro: Nenhuma Simulacao Activa Neste Momento \n");

}

break;

//---------------------------------------------------------------------------------------

case 3: //Eventos em Tempo Real

if(Simulacao\_Em\_Curso==true)

{

Escreve\_Cabecalho(Escolha\_Menu);

}

else

{

printf("Erro: Nenhuma Simulacao Activa Neste Momento \n");

}

break;

//---------------------------------------------------------------------------------------

case 4: //Estado Zonas em Tempo Real

if(Simulacao\_Em\_Curso==true)

{

Mostra\_Zonas\_Tempo\_Real(Tamanho\_Maximo\_Bar, Tamanho\_Maximo\_Carroussel, Tamanho\_Maximo\_Feira, Total\_Clientes\_Acutalmente\_No\_Bar, Total\_Clientes\_Acutalmente\_No\_Carroussel, Total\_Clientes\_Acutalmente\_Na\_Feira, Total\_Clientes\_Acutalmente\_Na\_Fila\_Bar, Total\_Clientes\_Acutalmente\_Na\_Fila\_MontanhaRussa, Total\_Clientes\_Acutalmente\_Na\_Fila\_Carroussel, Total\_Clientes\_Acutalmente\_Nas\_Filas\_Bilheteiras);

}

else

{

printf("Erro: Nenhuma Simulacao Activa Neste Momento \n");

}

break;

//---------------------------------------------------------------------------------------

case 5: //Sobre

Mostra\_Informacao\_Grupo();

break;

//---------------------------------------------------------------------------------------

case 6: //Ajuda

Mostra\_Funcionamento();

break;

//---------------------------------------------------------------------------------------

case 7: //Fechar

if(Simulacao\_Em\_Curso==true)

{

//Escreve Estatisticas (do Momento em Que a Simulacao Foi Interrompida) no Ficheiro

Escreve\_Estatisticas(Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Bar, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Tempo\_Maximo\_Espera\_Antes\_Ser\_Atendido\_Bar, Tempo\_Maximo\_Espera\_Antes\_Andar\_MontanhaRussa, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Carroussel, Tempo\_Maximo\_Espera\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Fila\_Bar\_Antes\_Ser\_Atendido, Soma\_Tempos\_Espera\_Fila\_MontanhaRussa\_Antes\_Andar\_MontanhaRussa, Soma\_Tempos\_Espera\_Fila\_Carroussel\_Antes\_Entrar\_Carroussel, Soma\_Tempos\_Espera\_Filas\_Bilheteiras\_Antes\_Entrar\_Feira, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Bar, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_MontanhaRussa, Soma\_Tempos\_Espera\_Antes\_Desistir\_Fila\_Carroussel, Soma\_Tempos\_Espera\_Antes\_Desistir\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Mobilidade\_Reduzida\_Atendidos\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Entraram\_Feira, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Bar, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Mobilidade\_Reduzida\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Entraram\_Fila\_Bar, Total\_Clientes\_Criancas\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Criancas\_Atendidos\_Bar, Total\_Clientes\_Criancas\_Andaram\_MontanhaRussa, Total\_Clientes\_Criancas\_Entraram\_Carroussel, Total\_Clientes\_Criancas\_Entraram\_Feira, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Bar, Total\_Clientes\_Criancas\_Desistiram\_Fila\_MontanhaRussa, Total\_Clientes\_Criancas\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Criancas\_Desistiram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Filas\_Bilheteiras, Total\_Clientes\_Publico\_Geral\_Atendidos\_Bar, Total\_Clientes\_Publico\_Geral\_Andaram\_MontanhaRussa, Total\_Clientes\_Publico\_Geral\_Entraram\_Carroussel, Total\_Clientes\_Publico\_Geral\_Entraram\_Feira, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Bar, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_MontanhaRusssa, Total\_Clientes\_Publico\_Geral\_Desistiram\_Fila\_Carroussel, Total\_Clientes\_Publico\_Geral\_Desistiram\_Filas\_Bilheteiras, Soma\_Tempos\_Estadia\_Feira, Total\_Clientes\_Sairam\_Feira);

}

//Fechar Socket

close(newsockfd);

close(sockfd);

exit(0);

break;

//---------------------------------------------------------------------------------------

default: printf("Erro: Opcao Inserida Nao Definida. \n");

break;

}

}

int main(int argc, char \*argv[])

{

Ler\_Ficheiro\_Config2();

//Faz Ligacao à Socket Quando Simulador é Iniciado. Simulador Comeca Sempre 1º

int servlen;

struct sockaddr\_un serv\_addr;

if ((sockfd= socket(AF\_UNIX, SOCK\_STREAM, 0)) < 0)

{

perror("Erro ao criar socket cliente\n");

}

bzero((char \*) &serv\_addr, sizeof(serv\_addr));

serv\_addr.sun\_family = AF\_UNIX;

strcpy(serv\_addr.sun\_path, UNIXSTR\_PATH);

servlen = strlen(serv\_addr.sun\_path) + sizeof(serv\_addr.sun\_family);

if (connect(sockfd, (struct sockaddr \*) &serv\_addr, servlen) < 0)

{

perror("Erro ao fazer connect\n");

}

system("clear");

Escreve\_Cabecalho(0);

Coloca\_Linhas\_Em\_Branco(24);

while(Escolha\_Menu!=7)

{

Menu\_Principal();

}

return 0;

}

## Simulador

#include "unix.h"

//Variaveis Que Vao Guardar os Valores dos Parametros Lidos no Ficheiro Config

int Tempo\_Medio\_Chegadas=0;

int Tempo\_Simulacao=0;

int Tamanho\_Maximo\_Bar=0;

int Tamanho\_Maximo\_Carroussel=0;

int Tamanho\_Maximo\_Feira=0;

int Tempo\_Maximo\_Espera\_Filas=0;

int Percentagem\_Ser\_Cliente\_Crianca=0;

int Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida=0;

int Numero\_Clientes\_A\_Criar=0;

int Tempo\_Medio\_Fica\_Zonas=0;

int Percentagem\_Ficar\_Bar=0;

int Tempo\_Medio\_Estadia\_Cliente\_Na\_Feira=0;

int newsockfd;

int sockfd;

int Fecha\_Feira=0; //Quando Todos os Cliente Forem Embora, Acaba a Thread do Gestor do Carroussel

int Valor\_Semaforo\_MontanhaRussa;

int Aux\_MontanhaRussa=2;

int Pessoas\_No\_Semaforo\_MontanhaRussa=0;

//Funcao Que Recebe o Valor Inserido (Tempo Medio de Qualquer Coisa) no Ficheiro Config e Devolve Valor Random Baseado no Valor Lido

int Calcula\_Tempos\_Medios\_Random(int Valor\_Lido\_Config)

{

srand(time(NULL));

int Desvio;

int Random;

int Valor\_Lido\_Mais\_Um; //Para Ser, Por Exemplo, de 0 a 10 e nao de 0 a 9

if(Valor\_Lido\_Config>1)

{

Desvio=Valor\_Lido\_Config/2; //Se For Impar Arredondar Para Tras. Isto é, 1.5->1

Valor\_Lido\_Mais\_Um=Valor\_Lido\_Config+1;

Random=(rand() %Valor\_Lido\_Mais\_Um)+Desvio;

}

else if(Valor\_Lido\_Config==1) //Caso Especial Se For 1. Temos de Deixar o Random a 1

{

Random=1;

}

/\*

Exemplo 1: Para Valor 10)

0 a 10 -> +5(=Desvio) -> 5 a 15 <=> Valor\_Lido\_Menos\_Desvio a Valor\_Lido\_Mais\_Desvio

Exemplo 2: Para Valor 1)

0 a 1 -> +0(=Desvio) -> 0 a 1 <=> Existe Hipotese de Nada Acontecer. Ou Seja, Se Valor\_Lido\_Config For 1, o Random Tem de Ser 1.

\*/

return Random;

}

//Funcao Que Define Idade do Cliente

int Define\_Idade\_Cliente(int Tipo\_Cliente)

{

srand(time(NULL));

//Para Entrar Na Feira Tem de Ser >=2 e <=80

//Crianca: Dos 2 aos 17

//PG: Dos 18 aos 80

//MR: Qualquer Idade

int Random;

if(Tipo\_Cliente==2) //Se For Crianca (>=2 e <18)

{

Random=(rand() %16) + 2; //Cria Numero Random de 2 a 17

}

else if(Tipo\_Cliente==3) //Se For Publico Geral (>=18)

{

Random=(rand() %63) + 18; //Cria Numero Random de 18 a 80

}

else if(Tipo\_Cliente==1) //Se For Mobilidade Reduzida (Pode Ser Qualquer Idade)

{

Random=(rand() %73) + 8; //Cria Numero Random de 8 a 80

}

return Random;

}

int Fica\_Ou\_Nao\_Bar(int Percentagem\_Ficar\_Bar)

{

srand(time(NULL));

int Random=(rand() %100)+1; //Cria Random de 1 a 100

int Fica;

if(Random<=Percentagem\_Ficar\_Bar)

{

Fica=1;

}

else

{

Fica=0;

}

return Fica;

}

//Funcao Que Define Tipo dos Clientes Baseado nas Percentagens Lidas do Ficheiro Config

int Define\_Tipo\_Cliente\_Random(int Percentagem\_Ser\_Cliente\_Crianca, int Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida)

{

srand(time(NULL));

int Tipo\_Cliente=0;

int Random=(rand() %100)+1; //Cria Numero Random de 1 a 100

/\* Exemplo Funcionamento:

Prob Cri=5%->1,2,3,4,5

Prob MR=10%->6,7,8,9,10,11,12,13,14,15

Random<=Prob Cri Ou seja, Random<=5

((Random>Prob Cri) && (Random<=(Prob Cri+Prob MR))) Ou Seja, Random>=6 e <=15 \*/

//Ser Crianca

if(Random<=Percentagem\_Ser\_Cliente\_Crianca)

{

Tipo\_Cliente=2;

}

//Ser Mobilidade Reduzida

else if((Random>Percentagem\_Ser\_Cliente\_Crianca) && (Random<=(Percentagem\_Ser\_Cliente\_Crianca+Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida)))

{

Tipo\_Cliente=1;

}

else //Publico Geral

{

Tipo\_Cliente=3;

}

return Tipo\_Cliente;

}

//Funcao Que Escreve Mensagem Para Enviar Pela Socket

void Escreve\_Mensagem(int newsockfd, int ID\_Cliente, int Tipo\_Cliente, int Estado, int Hora\_Actual, int Idade, int Hora\_Entrada\_Fila\_Bar, int Hora\_Desistencia\_Fila\_Bar, int Hora\_Atendimento\_Bar, int Hora\_Entrada\_Fila\_Carroussel, int Hora\_Desistencia\_Fila\_Carroussel, int Hora\_Entrada\_Carroussel, int Hora\_Entrada\_Fila\_MontanhaRussa, int Hora\_Desistencia\_Fila\_MontanhaRussa, int Hora\_Andou\_MontanhaRussa, int Hora\_Entrada\_Filas\_Bilheteiras, int Hora\_Desistencia\_Filas\_Bilheteiras, int Hora\_Entrada\_Feira, int Hora\_Saida\_Feira)

{

char Buffer[MAX\_LINE];

int Tamanho\_Mensagem;

sprintf(Buffer,"%d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d", ID\_Cliente, Tipo\_Cliente, Estado,Hora\_Actual,Idade,Hora\_Entrada\_Fila\_Bar, Hora\_Desistencia\_Fila\_Bar, Hora\_Atendimento\_Bar, Hora\_Entrada\_Fila\_Carroussel, Hora\_Desistencia\_Fila\_Carroussel, Hora\_Entrada\_Carroussel, Hora\_Entrada\_Fila\_MontanhaRussa, Hora\_Desistencia\_Fila\_MontanhaRussa, Hora\_Andou\_MontanhaRussa, Hora\_Entrada\_Filas\_Bilheteiras, Hora\_Desistencia\_Filas\_Bilheteiras, Hora\_Entrada\_Feira,Hora\_Saida\_Feira); // Writes in simulator socket buffer a message to monitor

Tamanho\_Mensagem=strlen(Buffer)+1; // Add one character more because of \n

if(write(newsockfd,Buffer,Tamanho\_Mensagem)!=Tamanho\_Mensagem)

perror("Erro ao escrever no cliente\n");

}

//Funcao Para Ler do Ficheiro Config:

void Ler\_Ficheiro\_Config()

{

//Declaracao e Inicializacao de Variaveis Auxiliares da Funcao

char Le\_Linha[100];

int Valor\_Variavel\_Lida;

char Nome\_Variavel\_Lida[100];

char Var1[]="T\_Med\_Cheg"; //"Tempo\_Medio\_Chegadas";

char Var2[]="T\_Sim"; //"Tempo\_Simulacao";

char Var3[]="Tam\_Max\_Bar"; //"Tamanho\_Maximo\_Bar";

char Var4[]="Tam\_Max\_Carroussel"; //"Tamanho\_Maximo\_Carroussel";

char Var5[]="Tam\_Max\_Feira"; //"Tamanho\_Maximo\_Feira";

char Var6[]="T\_Max\_Esp\_Filas"; //"Tempo\_Maximo\_Espera\_Filas";

char Var7[]="P\_Ser\_Cli\_Cri"; //"Percentagem\_Ser\_Cliente\_Crianca";

char Var8[]="P\_Ser\_Cli\_MR"; //"Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida";

char Var9[]="N\_Cli\_Criar"; //"Numero\_Clientes\_A\_Criar";

char Var10[]="T\_Med\_Fica\_Zona";//"Tempo\_Medio\_Fica\_Zonas";

char Var11[]="P\_Fica\_Bar";//"Percentagem\_Ficar\_Bar";

char Var12[]="T\_Med\_Fica\_Feira";//"Tempo\_Medio\_Estadia\_Cliente\_Na\_Feira";

int Erro=0; //Se Uma Das Restricoes For Violada, a Variavel Passa Para 1

FILE \*fp;

fp=fopen("config.conf","r"); //Vai Abrir o Ficheiro Config.conf

if(fp!=NULL)

{

//Ciclo de Leitura:

while(fgets(Le\_Linha,100,fp)!=NULL)

{

/\* %s= Nome da Variavel | %i= Valor da Variavel \*/

sscanf(Le\_Linha, "%s = %i", Nome\_Variavel\_Lida, &Valor\_Variavel\_Lida);

/\* Strcmp Tem Valor 0 se [Elemento1]=[Elemento2] \*/

if(strcmp(Nome\_Variavel\_Lida,Var1)==0)

{

Tempo\_Medio\_Chegadas=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var2)==0)

{

Tempo\_Simulacao=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var3)==0)

{

Tamanho\_Maximo\_Bar=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var4)==0)

{

Tamanho\_Maximo\_Carroussel=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var5)==0)

{

Tamanho\_Maximo\_Feira=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var6)==0)

{

Tempo\_Maximo\_Espera\_Filas=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var7)==0)

{

Percentagem\_Ser\_Cliente\_Crianca=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var8)==0)

{

Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var9)==0)

{

Numero\_Clientes\_A\_Criar=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var10)==0)

{

Tempo\_Medio\_Fica\_Zonas=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var11)==0)

{

Percentagem\_Ficar\_Bar=Valor\_Variavel\_Lida;

}

else if(strcmp(Nome\_Variavel\_Lida,Var12)==0)

{

Tempo\_Medio\_Estadia\_Cliente\_Na\_Feira=Valor\_Variavel\_Lida;

}

}

fclose(fp);

}

else

{

printf("Erro Abertura Ficheiro");

}

//------------------------------------------------------------------------------------------//

//-------------------------- Restricoes: --------------------------------------------------//

//------------------------------------------------------------------------------------------//

if(Tempo\_Medio\_Chegadas<=0)

{

printf("Erro: Tempo\_Medio\_Chegadas Tem de Ser >0 \n");

Erro=1;

};

if(Tempo\_Simulacao<2)

{

printf("Erro: Tempo\_Simulacao Tem de Ser >2 \n");

Erro=1;

};

if(Tamanho\_Maximo\_Bar<=0)

{

printf("Erro: Tamanho\_Maximo\_Bar Tem de Ser >0 \n");

Erro=1;

};

if(Tamanho\_Maximo\_Carroussel<=0)

{

printf("Erro: Tamanho\_Maximo\_Carroussel Tem de Ser >0 \n");

Erro=1;

};

if(Tamanho\_Maximo\_Feira<=0)

{

printf("Erro: Tamanho\_Maximo\_Feira Tem de Ser >0 \n");

Erro=1;

};

if(Tempo\_Maximo\_Espera\_Filas<0)

{

printf("Erro: Tempo\_Maximo\_Espera\_Filas Tem de Ser >=0 \n");

Erro=1;

};

if(Percentagem\_Ser\_Cliente\_Crianca<0 || Percentagem\_Ser\_Cliente\_Crianca>100)

{

printf("Erro: Percentagem\_Ser\_Cliente\_Crianca Tem de Ser >0%% e <=100%% \n");

Erro=1;

};

if(Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida<0 || Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida>100)

{

printf("Erro: Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida Tem de Ser >0%% e <=100%% \n");

Erro=1;

};

if(Numero\_Clientes\_A\_Criar<=0 || Numero\_Clientes\_A\_Criar>9999) //10000->Limite Maximo Inserido das Struct e das Threads Clientes

{

printf("Erro: Numero\_Clientes\_A\_Criar Tem de Ser >0 e <10000 \n");

Erro=1;

};

if(Tempo\_Medio\_Fica\_Zonas<=0)

{

printf("Erro: Tempo\_Medio\_Fica\_Zonas Tem de Ser >0 \n");

Erro=1;

};

if(Percentagem\_Ficar\_Bar<0 || Percentagem\_Ficar\_Bar>100)

{

printf("Erro: Percentagem\_Ficar\_Bar Tem de Ser >0%% e <=100%% \n");

Erro=1;

};

if(Tempo\_Medio\_Estadia\_Cliente\_Na\_Feira<=0)

{

printf("Erro: Tempo\_Medio\_Estadia\_Cliente\_Na\_Feira Tem de Ser >0 \n");

Erro=1;

};

//Ultima Restricao: Se a Soma das Percentagens do Tipo de Clientes For >100

if(Percentagem\_Ser\_Cliente\_Crianca+Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida>100)

{

printf("Erro: Soma Percentagens Tipo Clientes Nao Pode Ser >100%% \n");

Erro=1;

};

//Se Alguma das Restricoes Anteriores Acontecer, Programa Termina

if(Erro==1)

{

exit(0);

};

}

//Estrutura de Dados dos Clientes

typedef struct{

int Estado\_Seguinte;

int ID\_Cliente;

int Tipo\_Cliente;

int Tempo\_Espera\_Maximo\_Filas;

int Tempo\_Estadia\_Feira;

int Hora\_Entrada\_Fila\_Bar;

int Hora\_Desistencia\_Fila\_Bar;

int Hora\_Atendimento\_Bar;

int Hora\_Entrada\_Fila\_Carroussel;

int Hora\_Desistencia\_Fila\_Carroussel;

int Hora\_Entrada\_Carroussel;

int Hora\_Entrada\_Fila\_MontanhaRussa;

int Hora\_Desistencia\_Fila\_MontanhaRussa;

int Hora\_Andou\_MontanhaRussa;

int Hora\_Entrada\_Filas\_Bilheteiras;

int Hora\_Desistencia\_Filas\_Bilheteiras;

int Hora\_Entrada\_Feira;

int Hora\_Saida\_Feira;

int Lugar\_Fila\_Bilheteiras;

int Lugar\_Fila\_Bar;

int Lugar\_Fila\_Carroussel;

int Lugar\_Fila\_MontanhaRussa;

int Idade;

int Hora\_Vai\_Desistir\_Fila\_Bilheteiras;

int Hora\_Vai\_Desistir\_Fila\_Bar;

int Hora\_Vai\_Desistir\_Fila\_Carroussel;

int Hora\_Vai\_Desistir\_Fila\_MontanhaRussa;

int Tempo\_Fica\_Zona;

} Struct\_Clientes;

int Clientes\_Criados=0;

int Tempo\_Simulacao\_Actual=0; //Guarda o Instante Actual da Simulacao. Comeca a 0. Vai Incrementando Ate Acabar a Simulacao

Struct\_Clientes Cliente[10000]; //[Numero\_Clientes\_A\_Criar]; ->Numero de Structs de Clientes Criadas

//Declaracao Trincos:

pthread\_mutex\_t Trinco\_MontanhaRussa;

pthread\_mutex\_t Trinco\_Comunicacao\_Socket; //Trinco Que Impede Atribuir Instrucoes a Varios Clientes ao Mesmo Tempo

//Declaracao Threads:

pthread\_t Thread\_Ciclo\_Tempo\_Simulacao;

pthread\_t Thread\_Cliente[10000]; //[Numero\_Clientes\_A\_Criar]; ->Numero de Tarefas Criadas

pthread\_t Thread\_Gestor\_Carroussel; //Definicao da Thread do Gestor do Carroussel

pthread\_t Thread\_Gestor\_MontanhaRussa;

// Declaracao Semaforos:

//Semaforos Feira:

sem\_t Semaforo\_Lotacao\_Feira;

//Semaforos Bar:

sem\_t Semaforo\_Lotacao\_Bar;

//Semaforos Carroussel:

//sem\_t Semaforo\_Lotacao\_Carrousel; Retirado. Thread do Gestor Trata Disto Agora

//sem\_t Semaforo\_Espera\_Cliente\_Chegue;

//sem\_t Semaforo\_Deixa\_Cliente\_Entrar;

sem\_t Espera\_Cliente;

sem\_t Espera\_Sinal\_Entrada;

sem\_t Lotacao\_Carroussel;

sem\_t Semaforo\_So\_1\_Cliente;

//Semaforos Montanha Russa:

sem\_t Semaforo\_Desce\_MontanhaRussa;

int Clientes\_A\_Andar\_MontanhaRussa=0;

//Prioridades Fila Bilheteiras:

int Clientes\_MR\_Fila\_Bilheteiras=0;

int Resto\_Clientes\_Fila\_Bilheteiras=0;

//Prioridades Fila Bar:

int Clientes\_MR\_Fila\_Bar=0;

int Clientes\_PG\_Fila\_Bar=0;

int Clientes\_Cri\_Fila\_Bar=0;

//Prioridades Fila Montanha Russa:

//- Nao Tem, Pois Nao Importa o Tipo de Cliente. Apenas Tem de Descer aos Pares

int Clientes\_Fila\_MontanhaRussa=0;

//Prioridades Fila Carroussel:

int Clientes\_Cri\_Fila\_Carroussel=0;

int Clientes\_MR\_Fila\_Carroussel=0;

int Clientes\_PG\_Fila\_Carroussel=0;

int Clientes\_Cri\_Pequenas\_Fila\_Carroussel=0;

////////////////////////////////////// BAR //////////////////////////////////////////////

//Aumenta o Numero de Clientes na Fila Dependendo do Seu Tipo

void Aumenta\_Tipo\_Cliente\_Fila\_Bilheteiras(int Tipo\_Cliente)

{

if(Tipo\_Cliente==1)

{

Clientes\_MR\_Fila\_Bilheteiras++;

}

else

{

Resto\_Clientes\_Fila\_Bilheteiras++;

}

}

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

void Diminui\_Tipo\_Cliente\_Fila\_Bilheteiras(int Tipo\_Cliente)

{

if(Tipo\_Cliente==1)

{

Clientes\_MR\_Fila\_Bilheteiras--;

}

else

{

Resto\_Clientes\_Fila\_Bilheteiras--;

}

}

//Quando o Cliente Chega à Fila das Bilheteiras É Verificado o Tipo do Cliente Que Chegou e Poe o Cliente no Seu Lugar da Fila

void Atribui\_Lugar\_Fila\_Bilheteiras\_A\_Cliente(int ID\_Cliente)

{

int i;

if(Cliente[ID\_Cliente].Tipo\_Cliente==1)//Se o Cliente Que Chegou For MR

{

Cliente[ID\_Cliente].Lugar\_Fila\_Bilheteiras=Clientes\_MR\_Fila\_Bilheteiras; //Isto é, Vai Para o Lugar Do Ultimo Cliente MR (Numero de Clientes MR na Fila <=> Ao Lugar Onde Ele Vai Ficar Se Entrar na Fila)

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Tipo\_Cliente!=1 && Cliente[i].Lugar\_Fila\_Bilheteiras>0) //Se For Cri ou PG Vai 1 Lugar Para Tras && Esta na Fila das Bilheteiras (Se o Lugar For 0->Nao Esta na Fila)

{

Cliente[i].Lugar\_Fila\_Bilheteiras++;

}

}

//Como MR Tem Prioridade, Fica Atras dos MR Existentes e a Frente dos PG e Cri Existentes. Todos os Outros de PG e Cri, Vao 1 Lugar Para Tras. I.E. Lugar[Cliente PG ou Cri]+1;

}

else //Se o Cliente Que Chegou Nao For MR

{

Cliente[ID\_Cliente].Lugar\_Fila\_Bilheteiras=Clientes\_MR\_Fila\_Bilheteiras+Resto\_Clientes\_Fila\_Bilheteiras;

}

}

//Funcao Que Faz o Cliente Recebido (1º da Fila) Sair da Fila e Entrar na Zona. E Todos Avancam 1 Posicao

void Avanca\_Um\_Lugar\_Fila\_Bilheteiras()//int ID\_Cliente)

{

int i;

//Cliente[ID\_Cliente].Lugar\_Fila\_Bilheteiras=0; Nao Precisa Desta Linha Porque o 1º da Fila Esta Incluido no Ciclo Em Baixo

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Lugar\_Fila\_Bilheteiras>0) //Se Esta na Fila das Bilheteiras)

{

Cliente[i].Lugar\_Fila\_Bilheteiras--;

}

}

}

//Funcao Que Faz o Cliente Recebido Desistir da Fila e Todos Atras Dele Avancam 1 Posicao

void Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_Bilheteiras(int ID\_Cliente)

{

int i;

Cliente[ID\_Cliente].Lugar\_Fila\_Bilheteiras=0; //Cliente Que Desiste Sai da Fila (Poiscao=0)

for(i=0;i<Clientes\_Criados;i++) //Todos os Clientes Atras Dele Avancam 1 Posicao

{

if(Cliente[i].Lugar\_Fila\_Bilheteiras>0) //Se Esta na Fila das Bilheteiras)

{

Cliente[i].Lugar\_Fila\_Bilheteiras--;

}

}

}

//Aumenta o Numero de Clientes na Fila Dependendo do Seu Tipo

void Aumenta\_Tipo\_Cliente\_Fila\_Bar(int Tipo\_Cliente)

{

if(Tipo\_Cliente==1)

{

Clientes\_MR\_Fila\_Bar++;

}

else if(Tipo\_Cliente==3)

{

Clientes\_PG\_Fila\_Bar++;

}

else if(Tipo\_Cliente==2)

{

Clientes\_Cri\_Fila\_Bar++;

}

}

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

void Diminui\_Tipo\_Cliente\_Fila\_Bar(int Tipo\_Cliente)

{

if(Tipo\_Cliente==1)

{

Clientes\_MR\_Fila\_Bar--;

}

else if(Tipo\_Cliente==3)

{

Clientes\_PG\_Fila\_Bar--;

}

else if(Tipo\_Cliente==2)

{

Clientes\_Cri\_Fila\_Bar--;

}

}

//Quando o Cliente Chega à Fila do Bar É Verificado o Tipo do Cliente Que Chegou e Poe o Cliente no Seu Lugar da Fila

void Atribui\_Lugar\_Fila\_Bar\_A\_Cliente(int ID\_Cliente)

{

int i;

if(Cliente[ID\_Cliente].Tipo\_Cliente==1)//Se o Cliente Que Chegou For MR

{

Cliente[ID\_Cliente].Lugar\_Fila\_Bar=Clientes\_MR\_Fila\_Bar; //Isto é, Vai Para o Lugar Do Ultimo Cliente MR (Numero de Clientes MR na Fila <=> Ao Lugar Onde Ele Vai Ficar Se Entrar na Fila)

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Tipo\_Cliente!=1 && Cliente[i].Lugar\_Fila\_Bar>0) //Se For Cri ou PG Vai 1 Lugar Para Tras && Esta na Fila das Bar (Se o Lugar For 0->Nao Esta na Fila)

{

Cliente[i].Lugar\_Fila\_Bar++;

}

}

//Como MR Tem Prioridade, Fica Atras dos MR Existentes e a Frente dos PG e Cri Existentes. Todos os Outros de PG e Cri, Vao 1 Lugar Para Tras. I.E. Lugar[Cliente PG ou Cri]+1;

}

else if(Cliente[ID\_Cliente].Tipo\_Cliente==3)//Se o Cliente Que Chegou For PG

{

Cliente[ID\_Cliente].Lugar\_Fila\_Bar=Clientes\_MR\_Fila\_Bar+Clientes\_PG\_Fila\_Bar; //Isto é, Vai Para o Lugar Do Ultimo Cliente PG (Numero de Clientes MR+PG na Fila <=> Ao Lugar Onde Ele Vai Ficar Se Entrar na Fila)

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Tipo\_Cliente==2 && Cliente[i].Lugar\_Fila\_Bar>0) //Se For Cri Vai 1 Lugar Para Tras && Esta na Fila das Bar (Se o Lugar For 0->Nao Esta na Fila)

{

Cliente[i].Lugar\_Fila\_Bar++;

}

}

}

else if(Cliente[ID\_Cliente].Tipo\_Cliente==2)//Se o Cliente Que Chegou For Cri

{

Cliente[ID\_Cliente].Lugar\_Fila\_Bar=Clientes\_MR\_Fila\_Bar+Clientes\_PG\_Fila\_Bar+Clientes\_Cri\_Fila\_Bar; //Vai Para o Ultimo Lugar da Fila Composta Pela Soma de Todas as Outras "Sub Filas MR, PG e Cri"

}

}

//Funcao Que Faz o Cliente Recebido (1º da Fila) Sair da Fila e Entrar na Zona. E Todos Avancam 1 Posicao

void Avanca\_Um\_Lugar\_Fila\_Bar()//int ID\_Cliente)

{

int i;

//Cliente[ID\_Cliente].Lugar\_Fila\_Bar=0; Nao Precisa Desta Linha Porque o 1º da Fila Esta Incluido no Ciclo Em Baixo

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Lugar\_Fila\_Bar>0) //Se Esta na Fila da Bar)

{

Cliente[i].Lugar\_Fila\_Bar--;

}

}

}

//Funcao Que Faz o Cliente Recebido Desistir da Fila e Todos Atras Dele Avancam 1 Posicao

void Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_Bar(int ID\_Cliente)

{

int i;

Cliente[ID\_Cliente].Lugar\_Fila\_Bar=0; //Cliente Que Desiste Sai da Fila (Poiscao=0)

for(i=0;i<Clientes\_Criados;i++) //Todos os Clientes Atras Dele Avancam 1 Posicao

{

if(Cliente[i].Lugar\_Fila\_Bar>0) //Se Esta na Fila do Bar)

{

Cliente[i].Lugar\_Fila\_Bar--;

}

}

}

////////////////////////// Montanha Russa //////////////////////////////////////////////

//Quando o Cliente Chega à Fila da Montanha Russa Poe o Cliente no Seu Lugar da Fila

void Atribui\_Lugar\_Fila\_MontanhaRussa\_A\_Cliente(int ID\_Cliente)

{

Cliente[ID\_Cliente].Lugar\_Fila\_MontanhaRussa=Clientes\_Fila\_MontanhaRussa; //Vai Para o Ultimo Lugar da Fila

}

//Quando 2 Pessoas Andam na Montanha Russa, Todos Vao Avancar 2 Posicoes (Lugar\_Fila-2)

void Avanca\_Dois\_Lugares\_Fila\_MontanhaRussa()//int ID\_Cliente)

{

int i;

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Lugar\_Fila\_MontanhaRussa==1) //Se o Cliente Estiver na Fila da Montanha Russa e For o 1º (Regra Especial Pois 1-2 =-1)

{

Cliente[i].Lugar\_Fila\_MontanhaRussa-1; //O 1º da Fila So Passa 1 Lugar Para a Frente!

}

if(Cliente[i].Lugar\_Fila\_MontanhaRussa>1) //Se o Cliente Estiver na Fila da Montanha Russa e Nao é o 1º da Fila

{

Cliente[i].Lugar\_Fila\_MontanhaRussa-2; //Passa 2 Lugares Para a Frente Pois os 2 Primeiros Andaram na Montanha Russa

}

}

//A Parte Importante Desta Funcao É o Facto de Que Sempre Que 2 Clientes Andam na Montanha Russa, TODOS Andam 2 Posicoes. O 3º Lugar Ate ao Fim da Fila Andam 2 Casas. O 2º Lugar da Fila Tambem Anda 2 Casas (2-2=0. 0<=> Nao Esta na Fila/Saiu da Fila.

//No Entanto o 1º Lugar Apenas Anda 1 Casa. Senao Ficaria (1-2=-1)

}

//Funcao Que Faz o Cliente Recebido Desistir da Fila e Todos Atras Dele Avancam 1 Posicao

void Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_MontanhaRussa(int ID\_Cliente)

{

int i;

Cliente[ID\_Cliente].Lugar\_Fila\_MontanhaRussa=0; //Cliente Que Desiste Sai da Fila (Poiscao=0)

for(i=0;i<Clientes\_Criados;i++) //Todos os Clientes Atras Dele Avancam 1 Posicao

{

if(Cliente[i].Lugar\_Fila\_MontanhaRussa>0) //Se Esta na Fila da Montanha Russa)

{

Cliente[i].Lugar\_Fila\_MontanhaRussa--;

}

}

}

/////////////////////////// Carroussel /////////////////////////////////////////////////

//Aumenta o Numero de Clientes na Fila Dependendo do Seu Tipo

void Aumenta\_Tipo\_Cliente\_Fila\_Carroussel(int Tipo\_Cliente, int Idade)

{

if(Tipo\_Cliente==2 && Idade<=10)

{

Clientes\_Cri\_Pequenas\_Fila\_Carroussel++;

}

else if(Tipo\_Cliente==2 && Idade>10)

{

Clientes\_Cri\_Fila\_Carroussel++;

}

else if(Tipo\_Cliente==1)

{

Clientes\_MR\_Fila\_Carroussel++;

}

else if(Tipo\_Cliente==3)

{

Clientes\_PG\_Fila\_Carroussel++;

}

}

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

void Diminui\_Tipo\_Cliente\_Fila\_Carroussel(int Tipo\_Cliente, int Idade)

{

if(Tipo\_Cliente==2 && Idade<=10)

{

Clientes\_Cri\_Pequenas\_Fila\_Carroussel--;

}

else if(Tipo\_Cliente==2 && Idade>10)

{

Clientes\_Cri\_Fila\_Carroussel--;

}

else if(Tipo\_Cliente==1)

{

Clientes\_MR\_Fila\_Carroussel--;

}

else if(Tipo\_Cliente==3)

{

Clientes\_PG\_Fila\_Carroussel--;

}

}

//Quando o Cliente Chega à Fila da Carroussel É Verificado o Tipo do Cliente Que Chegou e Poe o Cliente no Seu Lugar da Fila

void Atribui\_Lugar\_Fila\_Carroussel\_A\_Cliente(int ID\_Cliente)

{

int i;

if(Cliente[ID\_Cliente].Tipo\_Cliente==2 && Cliente[ID\_Cliente].Idade<=10)//Se o Cliente Que Chegou For Cri Mais Nova

{

Cliente[ID\_Cliente].Lugar\_Fila\_Carroussel=Clientes\_Cri\_Pequenas\_Fila\_Carroussel; //Isto é, Vai Para o Lugar Do Ultimo Cliente Cri Mais Nova (Numero de Clientes Cri Mais Nova na Fila <=> Ao Lugar Onde Ele Vai Ficar Se Entrar na Fila)

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Idade>10 && Cliente[i].Lugar\_Fila\_Carroussel>0) //Se For Criana Mais Velha, MR ou PG Vai 1 Lugar Para Tras && Esta na Fila do Carroussel (Se o Lugar For 0->Nao Esta na Fila). Aqui Verificamos Pela Idade! Cri Novas dos 2 aos 10

{

Cliente[i].Lugar\_Fila\_Carroussel++;

}

}

//Como Cri Mais Novas Tem Prioridade, Fica Atras das Cri Mais Novas Existentes e a Frente das Criancas Mais Velhas e dos MR e PG Existentes. Todos os Outros , Vao 1 Lugar Para Tras. I.E. Lugar[Cliente Cri Velha ou MR ou PG]+1;

}

if(Cliente[ID\_Cliente].Tipo\_Cliente==2 && Cliente[ID\_Cliente].Idade>10)//Se o Cliente Que Chegou For Cri Mais Velha

{

Cliente[ID\_Cliente].Lugar\_Fila\_Carroussel=Clientes\_Cri\_Pequenas\_Fila\_Carroussel+Clientes\_Cri\_Fila\_Carroussel; //Isto é, Vai Para o Lugar Do Ultimo Cliente Cri (Numero de Clientes Cri na Fila <=> Ao Lugar Onde Ele Vai Ficar Se Entrar na Fila)

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Tipo\_Cliente!=2 && Cliente[i].Lugar\_Fila\_Carroussel>0) //Se For MR ou PG Vai 1 Lugar Para Tras && Esta na Fila do Carroussel (Se o Lugar For 0->Nao Esta na Fila)

{

Cliente[i].Lugar\_Fila\_Carroussel++;

}

}

//Fica Atras das Criancas Todas Existentes e a Frente dos MR e PG Existentes. Todos os Outros de MR e PG, Vao 1 Lugar Para Tras. I.E. Lugar[Cliente MR ou PG]+1;

}

else if(Cliente[ID\_Cliente].Tipo\_Cliente==1)//Se o Cliente Que Chegou For MR

{

Cliente[ID\_Cliente].Lugar\_Fila\_Carroussel=Clientes\_Cri\_Pequenas\_Fila\_Carroussel+Clientes\_Cri\_Fila\_Carroussel+Clientes\_MR\_Fila\_Carroussel; //Isto é, Vai Para o Lugar Do Ultimo Cliente MR (Numero de Clientes Cri+MR na Fila <=> Ao Lugar Onde Ele Vai Ficar Se Entrar na Fila)

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Tipo\_Cliente==3 && Cliente[i].Lugar\_Fila\_Carroussel>0) //Se For PG Vai 1 Lugar Para Tras && Esta na Fila do Carroussel (Se o Lugar For 0->Nao Esta na Fila)

{

Cliente[i].Lugar\_Fila\_Carroussel++;

}

}

}

else if(Cliente[ID\_Cliente].Tipo\_Cliente==3)//Se o Cliente Que Chegou For PG

{

Cliente[ID\_Cliente].Lugar\_Fila\_Carroussel=Clientes\_Cri\_Pequenas\_Fila\_Carroussel+Clientes\_Cri\_Fila\_Carroussel+Clientes\_MR\_Fila\_Carroussel+Clientes\_PG\_Fila\_Carroussel; //Vai Para o Ultimo Lugar da Fila Composta Pela Soma de Todas as Outras "Sub Filas Cri, MR e PG"

}

}

//Funcao Que Faz o Cliente Recebido (1º da Fila) Sair da Fila e Entrar na Zona. E Todos Avancam 1 Posicao

void Avanca\_Um\_Lugar\_Fila\_Carroussel()//int ID\_Cliente)

{

int i;

for(i=0;i<Clientes\_Criados;i++)

{

if(Cliente[i].Lugar\_Fila\_Carroussel>0) //Se Esta na Fila do Carroussel)

{

Cliente[i].Lugar\_Fila\_Carroussel--;

}

}

}

//Funcao Que Faz o Cliente Recebido Desistir da Fila e Todos Atras Dele Avancam 1 Posicao

void Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_Carroussel(int ID\_Cliente)

{

int i;

Cliente[ID\_Cliente].Lugar\_Fila\_Carroussel=0; //Cliente Que Desiste Sai da Fila (Poiscao=0)

for(i=0;i<Clientes\_Criados;i++) //Todos os Clientes Atras Dele Avancam 1 Posicao

{

if(Cliente[i].Lugar\_Fila\_Carroussel>0) //Se Esta na Fila do Carroussel)

{

Cliente[i].Lugar\_Fila\_Carroussel--;

}

}

}

//Funcao de Tratamento dos Lugares Carroussel

void Gestao\_Entradas\_Carroussel()

{

while(Fecha\_Feira!=1)

{

sem\_wait(&Espera\_Cliente); //Semaforo Inicializado a 0. Quando Recebe Post do Cliente, Continua

sem\_wait(&Lotacao\_Carroussel); //Se Tiver Lugar no carroussel

sem\_post(&Espera\_Sinal\_Entrada);

sem\_post(&Semaforo\_So\_1\_Cliente); //Ultimo Passo da Rotina da Comunicacao Gestor Carroussel-Cliente

}

}

//Cria Gestor de Lugares Carroussel

void Criacao\_Gestor\_Carroussel()

{

if(pthread\_create(&(Thread\_Gestor\_Carroussel), NULL ,(void \*)&Gestao\_Entradas\_Carroussel,NULL) != 0)

{

printf("Erro na Criacao da Tarefa do Gestor do Carroussel\n");

exit(1);

}

}

int Executa\_Proxima\_Zona(int Proxima\_Zona, int ID\_Cliente\_Recebido, int Tempo\_Fica\_Zona)

{

int Hora\_Actual=0;

//Sinais Para Indicar Que Desistiu e Nao Vai Haver Entradas nas Zonas Nem Saidas

int Desistencia\_Fila\_Bar=0;

int Desistencia\_Fila\_Carroussel=0;

int Desistencia\_Fila\_MontanhaRussa=0;

srand(time(NULL));

if(Proxima\_Zona==2)//Vai ao Bar

{

//Instrucoes Bar:

//------------------------------------------------------------------------------------------------------

//case 4: Entra Fila Bar

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

//Define a Hora de Desistencia:

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Bar=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar+Cliente[ID\_Cliente\_Recebido].Tempo\_Espera\_Maximo\_Filas;

//Aumenta o Numero de Clientes na Fila Dependendo do Seu Tipo

Aumenta\_Tipo\_Cliente\_Fila\_Bar(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 4 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Atribui\_Lugar\_Fila\_Bar\_A\_Cliente(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

while(Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_Bar!=1 && Desistencia\_Fila\_Bar!=1) //Enquanto o Cliente da Tarefa Nao For 1º da Fila, Espera

{

if(Tempo\_Simulacao\_Actual>=Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Bar)

{

//Faz Evento de Desistencia da Fila do Bar

//------------------------------------------------------------------------------------------------------

//case 7: Desiste Fila Bar

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

Diminui\_Tipo\_Cliente\_Fila\_Bar(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 7 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_Bar(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

//Se Cliente Desistir, Activa Sinal Para Indicar Que Nao Vai Haver Entrada na Zona em Questao nem Saida

Desistencia\_Fila\_Bar=1;

}

}

//----------------------------------------------------------------------------------------------------------------------------

if(Desistencia\_Fila\_Bar==0)

{

//----------------------------------------------------------------------------------------------------------------------------

//Invoca Funcao Random Se Fica ou Nao no Bar Depois de Ser Atendido

int Fica\_Bar=Fica\_Ou\_Nao\_Bar(Percentagem\_Ficar\_Bar);

if(Fica\_Bar==1)

{

//Procedimento É Atendido e Fica no Bar

//------------------------------------------------------------------------------------------------------

//case 6: Fica no Bar/Sai Fila Bar, é Atendido e Fica no Bar

sem\_wait(&Semaforo\_Lotacao\_Bar);

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

Diminui\_Tipo\_Cliente\_Fila\_Bar(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 6 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Avanca\_Um\_Lugar\_Fila\_Bar();//int ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

sleep(Tempo\_Fica\_Zona);

//------------------------------------------------------------------------------------------------------

//case 8: Sai Sala Bar

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

//=Tempo\_Simulacao\_Actual;

Hora\_Actual=Tempo\_Simulacao\_Actual;

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 8 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

sem\_post(&Semaforo\_Lotacao\_Bar);

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

}

else

{

//Procedimento É Atendido e Sai do Bar

//------------------------------------------------------------------------------------------------------

//case 5: Sai Fila Bar, é Atendido e Sai do Bar

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

Diminui\_Tipo\_Cliente\_Fila\_Bar(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 5 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

sleep(1);

Avanca\_Um\_Lugar\_Fila\_Bar();

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

}//Fim do If se Fica ou Nao no Bar

}//Fim do If se Aconteceu Desistencia

}

else if(Proxima\_Zona==3)//Vai ao Carroussel

{

//Instrucoes Carroussel:

//------------------------------------------------------------------------------------------------------

//case 9: Entra Fila Carroussel

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel; //porque a hora pode mudar enquanto está percorrendo as instruçoes

//Define a Hora de Desistencia:

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Carroussel=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel+Cliente[ID\_Cliente\_Recebido].Tempo\_Espera\_Maximo\_Filas;

//Aumenta o Numero de Clientes na Fila Dependendo do Seu Tipo

Aumenta\_Tipo\_Cliente\_Fila\_Carroussel(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, Cliente[ID\_Cliente\_Recebido].Idade);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 9 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Atribui\_Lugar\_Fila\_Carroussel\_A\_Cliente(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

while(Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_Carroussel!=1 && Desistencia\_Fila\_Carroussel!=1) //Enquanto o Cliente da Tarefa Nao For 1º da Fila, Espera

{

if(Tempo\_Simulacao\_Actual>=Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Carroussel)

{

//Faz Evento de Desistencia da Fila da Carroussel

//------------------------------------------------------------------------------------------------------

//case 11: Desiste Fila Carroussel

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel;

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

Diminui\_Tipo\_Cliente\_Fila\_Carroussel(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, Cliente[ID\_Cliente\_Recebido].Idade);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 11 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_Carroussel(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

//Se Cliente Desistir, Activa Sinal Para Indicar Que Nao Vai Haver Entrada na Zona em Questao nem Saida

Desistencia\_Fila\_Carroussel=1;

}

}

//----------------------------------------------------------------------------------------------------------------------------

if(Desistencia\_Fila\_Carroussel==0)

{

sem\_wait(&Semaforo\_So\_1\_Cliente); //Semaforo Inicializado a 1. Quando Cliente Chega ao Carroussel, Faz Wait. Gestor Trata de 1 Cliente de Cada Vez (Exclusao Mutua Com Semaforo Em Vez de Trincos)

sem\_post(&Espera\_Cliente); //Sinaliza Que Chegou 1 Cliente

sem\_wait(&Espera\_Sinal\_Entrada); //Semaforo Inicializacao a 0.

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel;

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

Diminui\_Tipo\_Cliente\_Fila\_Carroussel(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, Cliente[ID\_Cliente\_Recebido].Idade);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 10 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Avanca\_Um\_Lugar\_Fila\_Carroussel();//int ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

sleep(Tempo\_Fica\_Zona);

//------------------------------------------------------------------------------------------------------

//case 16: Sai Carroussel

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

//=Tempo\_Simulacao\_Actual;

Hora\_Actual=Tempo\_Simulacao\_Actual;

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 16 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

//Abre Vaga no Semaforo do Carroussel

sem\_post(&Lotacao\_Carroussel);

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

}//Fim do If se Houve Desistencia

}

else if(Proxima\_Zona==4)//Vai ao Carroussel

{

int Cliente\_Esta\_No\_Semaforo=0;

//Instrucoes Carroussel:

//------------------------------------------------------------------------------------------------------

//case 12: Entra Fila Montanha Russa

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa;

//Define a Hora de Desistencia:

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_MontanhaRussa=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa+Cliente[ID\_Cliente\_Recebido].Tempo\_Espera\_Maximo\_Filas;

//Aumenta o Numero de Clientes na Fila

Clientes\_Fila\_MontanhaRussa++;

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 12 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

sleep(1);

Atribui\_Lugar\_Fila\_MontanhaRussa\_A\_Cliente(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

if(Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_MontanhaRussa<3 && Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_MontanhaRussa>0) //Se For 1º ou 2º da Fila

{

Cliente\_Esta\_No\_Semaforo=1;

Pessoas\_No\_Semaforo\_MontanhaRussa++;

}

while((Pessoas\_No\_Semaforo\_MontanhaRussa!=2 || Cliente\_Esta\_No\_Semaforo!=1) && Desistencia\_Fila\_MontanhaRussa!=1) //Enquanto Nao Chegar o Momento de Desistir, Espera. Nao Pode Ter Condicao dos 2 Primeiros Nao Desistem, Porque se So Chegar 1, Vai Ficar Sempre La, Como Tem de Esperar Por Outro. Neste Caso, Se Estiver no Semaforo e Estiver +1 Pessoa Tambem, Acaba o Ciclo

{

if(Tempo\_Simulacao\_Actual>=Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_MontanhaRussa)

{

if(Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_MontanhaRussa<3 && Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_MontanhaRussa>0) //Se For 1º ou 2º da Fila

{

Cliente\_Esta\_No\_Semaforo=0;

Pessoas\_No\_Semaforo\_MontanhaRussa--;

}

//Faz Evento de Desistencia da Fila da Montanha Russa

//------------------------------------------------------------------------------------------------------

//case 14: Desiste Fila Montanha Russa

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa;

//Diminui o Numero de Clientes na Fila

Clientes\_Fila\_MontanhaRussa--;

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 14 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_MontanhaRussa(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

//Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

//Se Cliente Desistir, Activa Sinal Para Indicar Que Nao Vai Haver Entrada na Zona em Questao nem Saida

Desistencia\_Fila\_MontanhaRussa=1;

}

}

//----------------------------------------------------------------------------------------------------------------------------

if(Desistencia\_Fila\_MontanhaRussa==0)

{

//------------------------------------------------------------------------------------------------------

//case 13: Sai Fila Montanha Russa e Anda na Montanha Russa

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa;

//Diminui o Numero de Clientes na Fila

Clientes\_Fila\_MontanhaRussa--;

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 13 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

sleep(1);

pthread\_mutex\_lock(&Trinco\_MontanhaRussa);

Clientes\_A\_Andar\_MontanhaRussa++;

pthread\_mutex\_unlock(&Trinco\_MontanhaRussa);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

sleep(2);

Aux\_MontanhaRussa--; // Quando For 0 <=> 2 Clientes Que Estavam no Semaforo da Montanha Russa Ja Desceram, Semaforo Volta ao Valor Original e Clientes na Fila Avancam 2 Lugares

if(Aux\_MontanhaRussa==0)

{

Avanca\_Dois\_Lugares\_Fila\_MontanhaRussa();

Pessoas\_No\_Semaforo\_MontanhaRussa=0;

Aux\_MontanhaRussa=2;

}

sleep(2);

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

}

}//Fim do if Se Houve Desistencia

}

void Gestor\_Eventos\_Clientes(int ID\_Cliente\_Recebido) //(int Numero\_Cliente)

{

Cliente[ID\_Cliente\_Recebido].ID\_Cliente=ID\_Cliente\_Recebido;

Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente=Define\_Tipo\_Cliente\_Random(Percentagem\_Ser\_Cliente\_Crianca, Percentagem\_Ser\_Cliente\_Mobilidade\_Reduzida);

Cliente[ID\_Cliente\_Recebido].Tempo\_Espera\_Maximo\_Filas=Tempo\_Maximo\_Espera\_Filas; //Variavel Lida do Config. //Tirar em Principio da Mensagem

Cliente[ID\_Cliente\_Recebido].Tempo\_Estadia\_Feira=Calcula\_Tempos\_Medios\_Random(Tempo\_Medio\_Estadia\_Cliente\_Na\_Feira);//Tirar em Principio da Mensagem

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira=0;

Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_Bilheteiras=0;

Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_Bar=0;

Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_Carroussel=0;

Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_MontanhaRussa=0;

Cliente[ID\_Cliente\_Recebido].Idade=Define\_Idade\_Cliente(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

//Chega a Esta Hora e Desiste da Fila:

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Bilheteiras=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Bar=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Carroussel=0;

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_MontanhaRussa=0;

//Tempo Que Vai Ficar na Zona:

Cliente[ID\_Cliente\_Recebido].Tempo\_Fica\_Zona=Calcula\_Tempos\_Medios\_Random(Tempo\_Medio\_Fica\_Zonas);

//int Estado; - Inserido Manualmente

int Hora\_Actual=0; //Variavel Auxiliar (Podia Nao Ter, Mas Facilita Perceber o Que Estamos a Por na Mensagem)

//Variaveis Para a Decisao de Entrada de Zonas:

int Proxima\_Zona=0;

//Variavel Que Guarda Se Houve Desistencia ou Nao

int Desistencia\_Fila\_Bilheteiras=0;

//case 1: Entra Fila Bilheteiras da Feira

//usleep(10000);

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

//Define a Hora de Desistencia:

Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Bilheteiras=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras+Cliente[ID\_Cliente\_Recebido].Tempo\_Espera\_Maximo\_Filas;

//Aumenta o Numero de Clientes na Fila Dependendo do Seu Tipo

Aumenta\_Tipo\_Cliente\_Fila\_Bilheteiras(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 1 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Atribui\_Lugar\_Fila\_Bilheteiras\_A\_Cliente(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

//usleep(10000);

//sleep(2);

while(Cliente[ID\_Cliente\_Recebido].Lugar\_Fila\_Bilheteiras!=1 && Desistencia\_Fila\_Bilheteiras!=1) //Enquanto o Cliente da Tarefa Nao For 1º da Fila, Espera

{

if(Tempo\_Simulacao\_Actual>=Cliente[ID\_Cliente\_Recebido].Hora\_Vai\_Desistir\_Fila\_Bilheteiras)

{

//Faz Evento de Desistencia da Fila das Bilheteiras

//------------------------------------------------------------------------------------------------------

//case 3: Desiste Fila Bilheteiras da Feira

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

Diminui\_Tipo\_Cliente\_Fila\_Bilheteiras(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 3 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Desiste\_Fila\_E\_Avancam\_Um\_Lugar\_Fila\_Bilheteiras(Cliente[ID\_Cliente\_Recebido].ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

//Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

//Se Cliente Desistir, Activa Sinal Para Indicar Que Nao Vai Haver Entrada na Zona em Questao nem Saida

Desistencia\_Fila\_Bilheteiras=1;

}

}

//------------------------------------------------------------------------------------------------------

if(Desistencia\_Fila\_Bilheteiras==0)

{

//------------------------------------------------------------------------------------------------------

//case 2: Sai Fila Bilheteiras da Feira e Entra na Feira

sem\_wait(&Semaforo\_Lotacao\_Feira);

//usleep(10000);

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

//Diminui o Numero de Clientes na Fila Dependendo do Seu Tipo

Diminui\_Tipo\_Cliente\_Fila\_Bilheteiras(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente);

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 2 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

//sleep(2);

sleep(1);

Avanca\_Um\_Lugar\_Fila\_Bilheteiras();//int ID\_Cliente);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

//usleep(10000);

//sleep(2);

srand(time(NULL));

//Escolha da 1a Zona Para Onde o Cliente Vai: (Fica Fora do Ciclo While Porque na Funcao "Executa\_Proxima\_Zona" ja Tem Esta Funcao no Final da Saida da Zona (Seja Desistencia ou Saida Normal)+ Restricao da Montanha Russa se Tipo Cliente for MR

if(Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente==1)

{

Proxima\_Zona=(rand() %2)+2; //Cria Numero Random de 2 a 3

}

else

{

Proxima\_Zona=(rand() %3)+2; //Cria Numero Random de 2 a 4

}

//Escolhe as Zonas de Forma Random Atraves das Funcoes Auxiliares:

while(Tempo\_Simulacao\_Actual<Cliente[ID\_Cliente\_Recebido].Tempo\_Estadia\_Feira)

{

Proxima\_Zona=Executa\_Proxima\_Zona(Proxima\_Zona, ID\_Cliente\_Recebido, Cliente[ID\_Cliente\_Recebido].Tempo\_Fica\_Zona);

}

//------------------------------------------------------------------------------------------------------

//case 15: Sai Feira

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira=Tempo\_Simulacao\_Actual;

//Hora\_Actual=Tempo\_Simulacao\_Actual;

Hora\_Actual=Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira; //Nao va a Hora Mudar Entre Estas 2 Instrucoes e Haver Diferencas.

Escreve\_Mensagem(newsockfd, Cliente[ID\_Cliente\_Recebido].ID\_Cliente, Cliente[ID\_Cliente\_Recebido].Tipo\_Cliente, 15 , Hora\_Actual, Cliente[ID\_Cliente\_Recebido].Idade, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Atendimento\_Bar, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Carroussel, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Fila\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Andou\_MontanhaRussa, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Desistencia\_Filas\_Bilheteiras, Cliente[ID\_Cliente\_Recebido].Hora\_Entrada\_Feira, Cliente[ID\_Cliente\_Recebido].Hora\_Saida\_Feira);

sleep(1);

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

sleep(1);

sem\_post(&Semaforo\_Lotacao\_Feira);

}//Fim do If se Houve Desistencia

}

//Vai Criar Clientes Ate Alcancar o Tempo de Simulacao OU Criar Todos os Clientes Definidos

void Criacao\_Clientes()

{

srand(time(NULL));

long Clientes\_Criados2=0;

//Explicacao Para Este Long -> Google linuxquestions.org a question about passing arguments into a thread in c

while(Tempo\_Simulacao\_Actual<Tempo\_Simulacao && Clientes\_Criados<=Numero\_Clientes\_A\_Criar)

{

if(pthread\_create(&(Thread\_Cliente[Clientes\_Criados]), NULL ,(void \*)&Gestor\_Eventos\_Clientes,(void \*)Clientes\_Criados2) != 0)

{

printf("Erro na criacao da tarefa de criacao dos clientes\n");

exit(1);

}

sleep(Calcula\_Tempos\_Medios\_Random(Tempo\_Medio\_Chegadas));

Clientes\_Criados++; //Variavel Global

Clientes\_Criados2++; //Variavel Local ->Senao, Para Eliminar Esta Variavel, Talvez Por a Variavel Global Como Long!

}

}

//Relogio Funciona Ate Todos os Clientes Sairem

void Funcao\_Incrementa\_Tempo\_Actual()

{

while(Fecha\_Feira!=1)

{

Tempo\_Simulacao\_Actual=Tempo\_Simulacao\_Actual+1;

sleep(1);

}

}

//Quando Recebe Mensagem do Monitor Com Sinal Para Comecar a Simulacao

void Comeca\_Simulacao()

{

if(pthread\_create(&(Thread\_Ciclo\_Tempo\_Simulacao), NULL ,(void \*)&Funcao\_Incrementa\_Tempo\_Actual, NULL) != 0)

{

printf("Erro na criacao da tarefa do ciclo do tempo\n");

exit(1);

}

}

int main(int argc, char \*argv[])

{

//Le os Parametros do Ficheiro Config

Ler\_Ficheiro\_Config();

pthread\_mutex\_init(&Trinco\_Comunicacao\_Socket, NULL);

pthread\_mutex\_init(&Trinco\_MontanhaRussa, NULL);

//Iniciar Semaforos/Semaphores: (Sintaxe: [Semaforo,Flag se o Semaforo Deve Ser Partilhado Com Outras Threads,Valor Inicial do Semaforo]

sem\_init(&Semaforo\_Lotacao\_Feira,0,Tamanho\_Maximo\_Feira);

sem\_init(&Semaforo\_Lotacao\_Bar,0,Tamanho\_Maximo\_Bar);

sem\_init(&Semaforo\_Desce\_MontanhaRussa,0,2); //Iniciado a 2 Porque Quando Tiver 2 Clientes, Estes Vao Juntos na Montanha Russa

//Semaforos Para o Gestor do Carroussel:

sem\_init(&Espera\_Cliente,0,0);

sem\_init(&Espera\_Sinal\_Entrada,0,0);

sem\_init(&Lotacao\_Carroussel,0,Tamanho\_Maximo\_Carroussel);

sem\_init(&Semaforo\_So\_1\_Cliente,0,1);

int clilen, servlen,i;

//int sockfd, newsockfd, clilen, servlen,i;

struct sockaddr\_un cli\_addr, serv\_addr;

// Cria Socket

if ((sockfd = socket(AF\_UNIX,SOCK\_STREAM,0)) < 0) perror("Erro ao criar socket stream servidor\n");

unlink(UNIXSTR\_PATH);

bzero((char \*)&serv\_addr, sizeof(serv\_addr));

serv\_addr.sun\_family = AF\_UNIX;

strcpy(serv\_addr.sun\_path, UNIXSTR\_PATH);

servlen = strlen(serv\_addr.sun\_path) + sizeof(serv\_addr.sun\_family);

if (bind(sockfd, (struct sockaddr \*) &serv\_addr, servlen) < 0) perror("Erro ao atribuir nome socket servidor\n");

listen(sockfd, 1);

clilen = sizeof(cli\_addr);

newsockfd = accept(sockfd, (struct sockaddr \*) &cli\_addr, &clilen);

if (newsockfd < 0) perror("Erro ao criar ligacao dedicada - accept\n");

printf("Pressione Enter Para Comecar Simulacao \n");

char Tecla=0;

while (Tecla!='\n' && Tecla!='\r') {Tecla=getchar();}

//Comeca Simulacao

Criacao\_Gestor\_Carroussel();

Comeca\_Simulacao();

Criacao\_Clientes();

sleep(2);

int qwe;

for(qwe=0;qwe<Clientes\_Criados;qwe++)

{

pthread\_join(Thread\_Cliente[qwe],NULL);

}

printf("Acabou Join Threads dos Clientes\n");

sleep(2);

//Quando Termina Todas as Threads Criadas, a Feira Comeca a Fechar

Fecha\_Feira=1;

pthread\_join(Thread\_Ciclo\_Tempo\_Simulacao,NULL);

printf("Acabou Join Thread do Ciclo do Tempo de Simulacao\n");

sleep(2);

//Mensagem de Fim de Simulacao

pthread\_mutex\_lock(&Trinco\_Comunicacao\_Socket);

Escreve\_Mensagem(newsockfd,0,0,50,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0);

printf("Enviou Mensagem Fim Simulacao!\n");

pthread\_mutex\_unlock(&Trinco\_Comunicacao\_Socket);

sleep(2);

//Fechar Socket

close(newsockfd);

return 0;

}

## Unix

#include <stdio.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <sys/un.h>

#include <stdlib.h>

#include <string.h>

#include <pthread.h>

#include <unistd.h>

#include <math.h>

#include <semaphore.h>

#include <time.h>

#include <stdbool.h>

#define ADDRESS "champ\_socket"

#define UNIXSTR\_PATH "/tmp/socket.unix.stream1"

#define UNIXDG\_PATH "/tmp/socket.unix.datagram"

#define UNIXDG\_TMP "/tmp/socket.unix.dgXXXXXXX"

#define MAX\_LINE 512