#include <stdio.h>

#include <stdlib.h>

#include <windows.h>

#include <time.h>

#include <string.h>

#include "TADColaDin.h"

#include "rep.h"

/\*

SIMULACION 01 - SUPERMERCADO

AUTORES: SANCHEZ TIRADO CITLALLI YASMIN

JIMENEZ DELGADO LUIS DIEGO

AARON GAMALIEL

VERSION: 1.0

DESCRIPCIÓN: Simulación de la atención de clientes en un supermercado

con la implementación del TAD Cola en C, (estática, estática circular y dinámica)

OBSERVACIONES:

COMPILACIÓN PARA GENERAR EL CÓDIGO OBJETO: gcc -c TADColaDin.c

gcc -c presentacionWin.o

COMPILACION DEL ARCHIVO .C PRINCIPAL gcc -o super super.c TADColaDin.o presentacionWin.o

\*/

#include <stdio.h>

#include <stdlib.h>

#include <windows.h>

#include <time.h>

#include <string.h>

#include "presentacion.h"

#include "TADColaDin.h"

/\*

#include "TADColaEst.h"

#include "TADColaEstCir.h"

\*/

//Declaracion de constantes

#define ancho 90 // Definimos el ancho del espacio que ocuparemos en la consola

#define repaintTime 20 //Tiempo en el que se volvera a pintar

#define tiempoEspaciado 200 //Tiempo de espaciado

#define espacioClientes 5 //Espacio que tendra cada cliente al momento de dibujarlo

//Funcion para ocultar el cursor(parpadeo).

void hidecursor**(**void**){**

HANDLE consoleHandle **=** GetStdHandle**(**STD\_OUTPUT\_HANDLE**);**

CONSOLE\_CURSOR\_INFO info**;**

info**.**dwSize **=** 100**;**

info**.**bVisible **=** FALSE**;**

SetConsoleCursorInfo**(**consoleHandle**,** **&**info**);**

**}**

/\* Nombre de la funcion: dibujaCaja

Descripción: En esta función dibuja una caja del supermecado, ademas

agrega una descripcion del estado de la caja

Recibe: 2 enteros(x, y) que proporcionan las coordenadas en donde se

comenzara a dibujar la caja; un entero(NumCaja)que guarda el numero de la

caja que se dibujara; un caracter(estado), el cual describe el estado de la caja

(si esta abierta o cerrada); y finalmente un entero(velocidad), que alamacena la

velocidad en que la cajera atendera los clientes

Devuelve: NULL \*/

void dibujaCaja**(**int x**,** int y**,** int NumCaja**,** char estado**,** int velocidad**){**

MoverCursor**(**x**,**y**);** //Se posiciona el cursor a donde se planea dibujar la caja

printf**(**"\*\*\*\t"**);** //Se dibuja la parte superior de la caja

**if(**estado **==** 'A'**){** //Verificamos si la caja esta Atendiendo

MoverCursor**(**x**+**7**,** y**);**

printf**(**" "**);** //Espacio por donde pasaran los clientes

MoverCursor**(**x**+**7**,** y**);**

printf**(**"Atendiendo"**);** //Imprimimos el estado de la caja

**}**

**else**

**if(**estado **==** 'F'**){** //Si la caja esta cerrada (estado = 'F')

MoverCursor**(**x**+**7**,** y**);**

printf**(**" "**);** //Espacio por donde pasan los clientes

MoverCursor**(**x**+**7**,** y**);**

printf**(**"CAJA CERRADA\n"**);** //Imprimimos el estado de la caja

**}**

**else{** //Si la caja esta abierta, pero no hay clientes

MoverCursor**(**x**+**7**,** y**);**

printf**(**" "**);** //Espacio para los clientes

MoverCursor**(**x**+**7**,** y**);**

printf**(**"Vacio"**);**//Imprimir estado de la caja

**}**

MoverCursor**(**x**+**2**,**y**+**1**);**//Movemos el cursor

printf**(**"\*\t Caja %d"**,** NumCaja**+**1**);**

MoverCursor**(**x**,**y**+**2**);**

printf**(**"\*\*\*\t"**);**

printf**(**"Atendiendo 1 cliente/%d seg\n"**,** velocidad**);**

**}**

/\* Nombre de la funcion: Cajeras

Descripción: Dibuja todas las cajas que se indiquen, además imprime en pantalla: El estado

de los clientes, el numero de clientes que han sido atendidos, el nombre del supermercado y el

estado del super(abierto o cerrado).

Recibe: La cantidad de cajas que se van a dibujar; el espacio que tendra la fila de cada cajera;

el estado de cada una de las cajas, el nombre del supermercado, la velocidad de llegada de los

clientes, la velocidad en que cada cajera atendera a los clientes, la cantidad de clientes que han

llegado y finalmente la cantidad de clientes atendidos.

Devuelve: un entero que representa una posicion abajo de lo que dibujamos \*/

int Cajeras**(**int cant**,** int largo**,** char estadoCajeras**[],** char marketName**[],** int llegadaClientes**,** int tiempoAtencion**[],** int cantClientesLlegados**,** int cantClientesAtendidos**){**

int i**;** //Indice

MoverCursor**(**0**,**0**);** //Movemos el cursor al origen

printf**(**" "**);** //Limpiamos este espacio, ya que aqui imprimiremos

MoverCursor**(**0**,**0**);**//Volvemos al origen

//Imprimimos el estado de los clientes, la cantidad de clientes que han llegado, la cantidad de clientes antendidos y el nombre del supermecado

printf**(**"%s \tRecibiendo 1 cliente/%d seg\tHan llegado %d clientes\t\t%d clientes han sido atendidos\t"**,** marketName**,** llegadaClientes**,** cantClientesLlegados**,** cantClientesAtendidos**);**

**if(**cantClientesAtendidos**>**100**){**//Condicion para que cierre el supermercado

printf**(**"\t\tEL SUPERMERCADO CERRARA\n"**);**

**}**

//Dibujamos cada una de las cajas

**for(**i**=**0**;** i**<**cant**;** i**++){**

dibujaCaja**(**largo**+**1**,** i**\***4**+**1**,** i**,** estadoCajeras**[**i**],** tiempoAtencion**[**i**]);**

**}**

**return** i**\***4**+**5**;** //Regresamos un entero que representa una posicion abajo de lo que acabamos de dibujar

**}**

/\* Nombre de la funcion: pintar

Descripcion: Dibuja cada uno de los clientes del supermercado

Recibe: la fila en la que se dibujara, la posiscion, el cliente y un caracter que describe si

la fila se ha terminado.

Devuelve: NULL \*/

void pintar**(**int filaPintar**,** int Posicion**,** elemento Cliente**,** char acaboFila**){**

char temp**[**4**];** //Creamos un arreglo auxiliar de caracteres

itoa**(**Cliente**.**n**,**temp**,**10**);**// Convierte a cadena el entero que esta dentro de la estructura elemento

int x **=** ancho **-** espacioClientes**\***Posicion **-** 3**;** //cordenada en x donde se dibuja al cliente

int y **=** filaPintar **\*** 4 **+** 2**;** //cordenada en y donde se dibuja al cliente

**if(**acaboFila **==** 'S'**){** //Verificamos si ya no hay espacio en la fila

**for(**int i**=**0**;** i**<**x**;** i**++){** //Hace un ciclo para limpira la fila

MoverCursor**(**i**,**y**);** //Recorrer el cursor desde el inicio hasta el final de la fila

printf**(**" "**);** //Imprime un espacio vacio

**}**

**}**

**else{**

MoverCursor**(**x**,**y**);** //Mueve el cursor al final de la fila

printf**(**" "**);** //Imprime un espacio de 5, entre clientes

MoverCursor**(**x**,**y**);**//mueve el cursor nuevamente

fflush**(**stdout**);** //limpiamos el buffer

printf**(**"#%s"**,** temp**);** //Imprimos al cliente

**}**

**return;**

**}**

//Funcion main

int main**(**void**){**

unsigned int tiempo**;** //Contador del tiempo

unsigned int cliente**;** //Variable auxiliar que lleva el contador de clientes

unsigned int atendidos**;** //Almacena el numero de clientes atendidos

unsigned int minClientes**;** //Minimo de clientes para cerrar el supermercado

unsigned int fila**;** //Almacena la fila sobre la cual se trabajara

unsigned int columna**;** //Almacena la columna sobre la cual se trabajara

unsigned int alto**;** //Alto del area en que se dibujara la simulacion

unsigned int cantCajeras**;** //Cantidad de cajeras que habra en el supermercado

unsigned int filaElegida**;** // Fila sobre la que se esta trabajando

unsigned int llegadaClientes**;** //Tiempo de llegada de los clientes

elemento aux**;** //Elemento auxiliar para agregarlo a la cola

char marketName**[**20**];** //Nombre del supermercado

char fin **=** 'N'**;** //Caracter que describe si el supermercado cerrara

//Pedimos al usuario que ingrese el nombre del supermercado

printf**(**"Introduce el nombre del supermercado: "**);**

scanf**(**"%s"**,** **&**marketName**);**

//Pedimos al usuario que ingrese la cantidad de cajeras con las que trajara el supermercado

printf**(**"Introduce la cantidad de cajeras (1-10): "**);**

scanf**(**"%d"**,** **&**cantCajeras**);**

//Validamos que ingrese un valor dentro del rango de numero de cajeras 0<n<10

**if(**cantCajeras**>**10 **||** cantCajeras **<** 1**){**

printf**(**"Cantidad de cajeras no valida\n"**);**

exit**(**0**);** //Cerramos el programa

**}**

//Pedimos al usuario que ingrese la minima cantidad de clientes para cerrar el supermercado

printf**(**"Introduce la minima cantidad de clientes para cerrar el supermercado: "**);**

scanf**(**"%d"**,** **&**minClientes**);**

char estadoCajeras**[**cantCajeras**];** // Arreglo para almacenear el estado de las cajeras

int tiempoAtencion**[**cantCajeras**];** //Arreglo para almacenar la velocidad a la que atenderan las cajeras

cola filaCajera**[**cantCajeras**];** // Creamos las filas(colas) donde se almacenaran a los clientes

**for(**int i**=**0**;** i**<**cantCajeras**;** i**++)**

Initialize**(&**filaCajera**[**i**]);**//Inicializamos todas las filas

//Pedimos al usuario que ingrese la velocidad de atencion de cada una de las cajeras en milisengundos

**for(**int i**=**0**;** i**<**cantCajeras**;** i**++){**

printf**(**"Tiempo de atencion de la caja #%d (multiplos de 10): "**,** i**+**1**);**

scanf**(**"%d"**,** **&**tiempoAtencion**[**i**]);**

**}**

//Pedimos al usuario que ingrese la velocidad de llegada de los clientes en milisengundos

printf**(**"Introduce la velocidad de llegada de Clientes: "**);**

scanf**(**"%d"**,** **&**llegadaClientes**);**

/\*Llamamos a la funcion cajeras, la cual dibujara la simulacion del supermercado, y nos retornara un valor entero que

representa una posicion en el eje de las y abajo del dibujo de la simulacion\*/

alto **=** Cajeras**(**cantCajeras**,** ancho**,** estadoCajeras**,** marketName**,** llegadaClientes**,** tiempoAtencion**,** cliente**,** atendidos**);**

tiempo **=** 0**;** cliente **=** 0**;** atendidos **=** 0**;** //Inicializamos las variables

system**(**"cls"**);** //Limpiamos la pantalla

hidecursor**();** //Ocultamos el cursor para que no parpadee

srand**(**time**(NULL));** //Damos la semilla a rand, tomando la hora actual, para generar numeros aleatorios

**while(**fin **==** 'N'**){** //Hacemos un ciclo que representara el tiempo en que esta funcionando el supermercado

tiempo**++;** //Aumentamos en 1 el tiempo cada que entre en el ciclo

fin **=** 'S'**;** //El supermecado cerrara

**for(**int i**=**0**;** i**<**cantCajeras**;** i**++){** //Comenzamos un ciclo desde 0 hasta el numero de cajeras

**if((!**Empty**(&**filaCajera**[**i**]))** **&&** estadoCajeras**[**i**]** **!=** 'F'**){** //Si la fila no esta vacia y la caja no esta cerrada

**if(**tiempo **%** tiempoAtencion**[**i**]** **==** 0**){** //Si ya se ha atendido un clinete

aux **=** Dequeue**(&**filaCajera**[**i**]);** //Se remueve el cliente de la fila

//Aqui deberia ir el mensaje de a quien antendio

atendidos**++;** //Aunmentamos en uno el contador de clientes antendidos

**}**

estadoCajeras**[**i**]** **=** 'A'**;** //Cambiamos el estado de la caja a Atendiendo

**}**

**else{**

//Aqui da el mensaje de que esta vacia la cola

estadoCajeras**[**i**]** **=** 'Z'**;** //Cambiamos el estado de la caja a vacia

**if(**atendidos**>**minClientes**)** //Si ya se cumplio con el minimo de clientes atendidos

estadoCajeras**[**i**]** **=** 'F'**;** //Cambiamos el estado de la caja a cerrada

**}**

//Dibujamos la simulacion actualizandola cada vez que entre al ciclo

Cajeras**(**cantCajeras**,**ancho**,** estadoCajeras**,** marketName**,** llegadaClientes**,** tiempoAtencion**,** cliente**,** atendidos**);**

**}**

// Cajeras(cantCajeras,ancho, estadoCajeras, marketName, llegadaClientes, tiempoAtencion);

//Aqui se dibujan los clientes

**for(**int i**=**0**;** i**<**cantCajeras**;** i**++){** //Inciamos un ciclo que vaya desde 0 hasta el numero de cajeras

**for(**int k**=**0**;** k**<=**Size**(&**filaCajera**[**i**]);** k**++){** //Ciclo que vaya desde cero hasta el tamano de la fila

// printf(" %d",k);

**if(**k **==** Size**(&**filaCajera**[**i**])){** //Si la fila ya termino

pintar**(**i**,**k**-**1**,**aux**,** 'S'**);** //Dibujamos los cliente

**}**

**else{**

aux **=** Element**(&**filaCajera**[**i**],**k**+**1**);** // Si la fila no ha terminado

pintar**(**i**,** k**,** aux**,** 'N'**);** //Dibujamos el cliente

**}**

**}**

**}**

//Cuando se ha antedido a un cliente y el supermecado sigue abierto

**if((**tiempo **%** llegadaClientes **==** 0**)** **&&** atendidos**<=**minClientes**){**

cliente**++;** //Aumnetamos la cantidada de clientes

aux**.**n **=** cliente**;** //Almacenamos al cliente dentro del elemento auxiliar para poder encolarlo

filaElegida **=** rand**()%**cantCajeras**;** //Asignmos la fila aleatoriamente al cliente

Queue**(&**filaCajera**[**filaElegida**],** aux**);** //Encolamos en la fila al cliente

**}**

Sleep**(**repaintTime**);** //Dormimos el programa con el tiempo base

// system("mode con lines=0 cols=0 >nul");

**for(**int a**=**0**;** a**<**cantCajeras**;** a**++){** //Verificamos que todas las cajas esten funcionando

**if(**estadoCajeras**[**a**]** **!=** 'F'**)**

fin **=** 'N'**;** //Si es el caso continuamos con la simulacion

//MoverCursor(0, alto+5); //Movemos el cursor

//printf("%c\n", fin);

**}**

**}**

MoverCursor**(**0**,**alto**+**1**);**//Movemos el cursor al final del dibujo de la simulacion

**return** 0**;**

**}**