Operación ReservarMatriz.

Entradas: Recibe dos enteros llamados filas y columnas, estos datos fueron inicializados por una constante llamada CAP Y después su valor fueron cambiados por el usuario en el Main.

Salida: Esta función (Si no hay un error al reservar) retorna una matriz dinámica con las dimensiones especificadas en las entradas de la función.

Conceptos usados: Se usa structs, en este caso de tipo Local, ya que al retornar una matriz de tipo Local, la función tiene que ser del mismo tipo. También se usa malloc para reservar el tamaño de la matriz.

Operación llenarLocal.

Entradas: Esta función no recibe ningún parámetro.

Salida: Retorna una variable temporal de tipo Local, esta variable tiene un char que corresponde al nombre del local creado, tiene cuatro datos enteros: el piso del local, el lugar del local, una variable de nombre numero que corresponde a un número aleatorio (Entre 1.000 y 10.000) de nombre idLocal, y localTemp que es las ventas totales hechas por el local.

Conceptos usados: rand()% con ayuda de la librería time.h para crear el numero aleatorio.

Operación guardarNuevoUsuario.

Entradas: recibe un puntero que apunta a otro puntero de tipo Local con nombre pMatriz, un entero que representa las filas de la matriz, otro que representa las columnas de la matriz, el piso y el lugar definido por los usuarios y un puntero de tipo Local llamado pLocal que recibe la dirección de memoria de una variable temporal de tipo Local creada en la operación llenarMatriz.

Salida: Le da valor al puntero pLocal, en concreto define la matriz con las filas correspondiendo a los pisos y las columnas correspondiendo a los lugares de la matriz.

Conceptos usados: Paso de parámetros y con punteros para el uso de pMatriz y pLocal.

Operación llenarMatriz.

Entradas: pMatriz, las filas y columnas de pMatriz.

Salida: Crea la variable temporal de tipo Local usada para el puntero de guardarNuevoUsuario junto con el resto de datos faltantes.

Conceptos usados: Paso de parámetros con punteros.

Operación mostrarLocal.

Entradas: recibe un puntero que apunta a otro puntero de tipo Local con nombre pMatriz, un entero que representa las filas de la matriz, otro que representa las columnas de la matriz, el piso y el lugar definido por los usuarios. Todo esto aportado por la operación verificacion

Salida: Después de comprobar si los parámetros dados corresponden con rango predefinido, se imprime el nombre del local, el piso, el lugar, el id, y las ventas totales hechas por la tienda.

Conceptos usados: Paso de parámetros con punteros y el && para el condicional.

Operación verificación.

Entradas: pMatriz, las filas y columnas de pMatriz.

Salida: pide el piso y lugar del local, y le pasa estos datos a mostrarLocal para que este pueda funcionar.

Conceptos usados: paso de parámetros entre funciones.

Operación mostrarAnimo.

Entrada: No recibe parámetros.

Salida: crea una variable animoUsuario de tipo estadoDeAnimo\_e y retorna lo asignado por el usuario para la variable animoUsuario.

Conceptos usados: structs para crear el tipo estadoDeAnimo\_e que contiene enums del 1 al 4 con las constantes: MAL, NORMAL, BIEN y SUPER.

Operación enums

Entrada: No recibe parámetros.

Salida: crea una variable llamada opinión que recibe un valor número por el usuario, y dependiendo el número la operación imprime 4 casos distintos.

Conceptos usados: el switch con el parámetro opinión.

Operación recursión.

Entradas recibe una variable llamada numero de tipo long, esta variable está definida en la función promoción.

Salida: en caso de que numero sea mayor a 2 retorna el residuo del numero dividido 2 + el valor de numero en la entrada de la función por 10 dividido entre dos para conseguir el binario del numero en base 2.

Conceptos usados: recursión para volver a tener el valor de numero en la entrada de la función.

Operación promoción.

Entrada: No recibe parámetros.

Salidas: primero crea un número aleatorio con el método ya explicado pero con diferentes rangos (entre 100 y 500), imprime un mensaje de felicitaciones y te explica que consiste la promoción, en caso de que cumplas con lo pedido la función retorna 0, de lo contrario retorna 1.

Conceptos usados: rand()% y simples condicionales.

Operación ingresaVenta.

Entradas: pMatriz, las filas y columnas de pMatriz.

Salidas: se crea una condición de 250.000, primero se ingresa un nombre temporal correspondiente al de un local ya creado, cuando lo encuentra le pide registrar la compra realizada. Si esta esta es igual o mayor a la condición se va a la operación promoción, de lo contrario simplemente se le adiciona el valor de la venta al local.

Conceptos usados: For anidados y strcmp para comparar el nombre temporal, con el nombre del local.

Operación eliminarLocal.

Entradas: pMatriz, las filas y columnas de pMatriz.

Salida: primero se ingresa un nombre temporal correspondiente al de un local ya creado, cuando lo encuentra esta función vuelve a inicializar los datos del local daño a entender que el lugar está libre.

Conceptos usados: For anidados, strcmp para comparar el nombre temporal, con el nombre del local y strcpy como volver a poner vacío el nombre del local.

Operación MostrarMenu.

Entradas: No recibe parámetros.

Salida: Imprime los diferentes opciones que tiene el programa y usando una variable opción que digita el usuario, la cual es retornada.

Conceptos usados: printf y scanf.

Operación Main.

Entradas: No recibe parámetros

Salida: Básicamente se inicializa to lo que se va a usar y se mandan los parámetros correspondientes a las operaciones y funciones que se van a realizar, esto con ayuda del menú de mostrarMenu.

Conceptos usados, paso de parámetros con punteros, el uso de un switch para escoger cada caso que necesita el usuario con la variable opinión y el uso de un do while para dale aspecto de menú desde el cmd.