



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**



FACULTAD DE INGENIERÍA

FUNDAMENTOS DE PROGRAMACIÓN

PRESENTA: ARVIZU GONZÁLEZ BRANDON ADRIÁN

CUELLAR ORBEZO JUAN MANUEL

REYES MACIAS FERNANDO ABDEL

SUAREZ RODRIGUEZ KEVIN AXEL

PROFESORA: KARINA GARCÍA MORALES

PROYECTO FINAL

FECHA DE ENTREGA: 17/05/24

SEMESTRE 2024-2

Análisis

Problema a solucionar

Realizar un sistema generador de tickets de compras de muebles, se utilizarán 3 departamentos con mínimo 3 artículos a elegir (para 5 tickets).

Datos de entrada

- Nombre del cliente
- Dirección del cliente
- Números que indican el tipo de muebles y la cantidad que busca adquirir el usuario.
- Datos tarjeta para realizar pago.

Datos auxiliares

- Un arreglo que contenga variables las cuales van a ser los muebles con su precio y una pequeña descripción del mismo.
- Uso de funciones para calcular los precios, cantidad de productos y dar al usuario la capacidad de agregar o quitar artículos de su carrito.

Proceso

Para solucionar el problema que se nos presenta requerimos hacer un arreglo en forma de matriz que contenga los artículos a vender, se escogieron 5 áreas a escoger y cada una de estas contará con 6 artículos de los cuales cada uno de ellos tendrá su nombre, precio y una breve descripción.

El arreglo nos servirá para facilitar la codificación, misma en la que se le pedirán sus datos al cliente (nombre y dirección) y se le dejará elegir los productos a comprar en cada uno de los 5 departamentos disponibles.

El código también le dará la opción al usuario de agregar a su carrito la cantidad de artículos que él decida y, en caso de que así lo requiera, también podrá quitar productos por si se arrepiente y así hasta que esté satisfecho con su elección.

Cada vez que nuestro usuario modifique su carrito, aparecerá en pantalla la cantidad de artículos que tiene y el precio de estos.

Finalmente, se le pedirán los datos de su tarjeta para realizar el pago y por último el programa generará automáticamente su ticket.

Datos de salida

· Un ticket que contiene los siguientes datos:

1. Número de ticket.
2. Nombre del cliente.
3. Dirección del cliente.
4. Cantidad total de productos.
5. Precio Total.

Pseudocódigo

Proceso Tienda_en_linea_IKEA

Definir ARREGLOMENU Como Cadena

Definir varR Como Entero

Definir arregloPrecios Como Real

Dimensionar matriz(5,7)

Definir OPCION Como Entero

Definir Tarjeta Como Real

Definir VigMes, VIGANO Como Entero

Definir ccv Como Entero

Definir TotalProd, TotalPrecio Como Real

Definir Nombre, DIRECCION Como Cadena

Definir Carrito Como Real

Escribir 'Hola! Bienvenido a la tienda en línea IKEA'

Escribir '¿Cuál es tu nombre?'

Leer Nombre

Escribir 'Excelente, ¿Podrías compartimos tu dirección?'

Leer DIRECCION

Escribir 'Muy bien, antes de continuar revisemos tus datos'

Escribir 'NOMBRE:', Nombre, ' ', 'DIRECCIÓN:', DIRECCION, ', ¿tus datos son correctos?'

Escribir '1=si 2=no'

Leer varR

Mientras varR>1 Hacer

Escribir 'vuelve a poner tus datos separados por comas (,).'

Leer Nombre, DIRECCION

Escribir 'Muy bien, antes de continuar revisemos tus datos'

Escribir 'NOMBRE:', Nombre, ' ', 'DIRECCIÓN:', DIRECCION, ', ¿tus datos son correctos?'

Escribir 's/n'

FinMientras

Repetir

Escribir 'Ahora escoge qué departamento quieres visitar:'

Escribir '1 Sala', ' ', ' ', '2 Baño', ' ', ' ', '3 Cuarto', ' ', ' ', '4 Cocina', ' ', ' ', '5 Patio', ' ', ' ', '6 Salir'

Leer OPCION

Según OPCION Hacer

1:

agregar al Carrito' Escribir 'Por favor elige los productos que gustes'

Escribir 'Opciones Sala'

Leer ProductoSala

2:

agregar al Carrito' Escribir 'Por favor elige los productos que gustes'

Escribir 'Opciones Baño'

Leer PRODUCTOBANO

3:

agregar al Carrito' Escribir 'Por favor elige los productos que gustes

Escribir 'Por favor elige los productos que gustes

Escribir 'Opciones Cuarto'

Leer ProductoCuarto

4:

agregar al Carrito' Escribir 'Por favor elige los productos que gustes

Escribir 'Por favor elige los productos que gustes

Escribir 'Opciones Cocina'

Leer ProductoCocina

5:

agregar al Carrito' Escribir 'Por favor elige los productos que gustes

Escribir 'Por favor elige los productos que gustes

Escribir 'Opciones Patio'

Leer ProductoPatio

6:

FinSegún

Escribir '¿Te gustaría agregar algo más a tu carrito? 1=sí, 2=no'

Leer DECISION

Hasta Que DESICION = 2

Escribir 'El total de tus productos es de:', TotalProd

Escribir 'El precio total es de:', TotalPrecio

Escribir 'Muy bien ahora solo ingresa tus datos de pago y te generaremos tu ticket.'

Leer Tarjeta

Leer VigMes, VIGANO

Leer ccv

Escribir 'Muchas gracias por tu compra'

FinProceso

Pruebas de escritorio

PROGRAMA

USUARIO

PRUEBA 1

Hola! Bienvenido a la tienda en línea de IKEA

Cual es tu nombre? **Rodrigo**

Excelente Rodrigo, Podrías compartirnos tu dirección? **232 Benito Juarez**

De acuerdo, entonces tu domicilio es 232 Benito Juarez ahí se hará la entrega de tus productos.

Antes de continuar revisemos tus datos :)

Rodrigo

232 Benito Juarez

Estás de acuerdo? [s/n] **s**

A continuación te presentamos los diferentes departamentos que tenemos en línea.

- 1) SALA
- 2) BAÑO
- 3) CUARTO
- 4) COCINA
- 5) PATIO
- 6) SALIR

Qué departamento deseas visitar? **5**

SALA

A Asador ARHOLMA \$ 1649.38

Asador portátil de acero inoxidable.

B Banco de jardín BONDHOLMEN \$ 2938.51

Banco de madera para exteriores.

C Cajonera CHAPLIN \$ 548.79

Caja de almacenamiento resistente a la intemperie.

D Dosel DYNING \$ 873.49

Dosel para protección solar con estructura de aluminio.

E Estera EKERÖN \$ 724.16

Estera de plástico tejido para proteger el suelo.

F Farol FÖRMEDLA \$ 210.73

Farol de metal para iluminar el patio por la noche.

G Gazebo GÅLÖ \$ 999.99

Gazebo plegable con cortinas para crear sombra.

Deseas agregar(a) o eliminar (e) un producto? **a**

Qué producto quisieras agregar a tu carrito? **A**

De acuerdo, y cuántos productos A quieres agregar? **3**

Agregaste 3 productos A a tu carrito

PATIO

3 productos en el depto:

3pz Asador ARHOLMA - \$4948.14

Deseas continuar (c) editando tu carrito o salir(s)? [c/s] **s**

A continuación te presentamos los diferentes departamentos que tenemos en línea.

- 1) SALA
- 2) BAÑO
- 3) CUARTO
- 4) COCINA
- 5) PATIO
- 6) SALIR

Qué departamento deseas visitar? **6**

Bienvenido al área de checkout!

Revisemos tu ticket :)

PATIO

3 productos en el depto:

3pz Asador ARHOLMA - \$4948.14

TOTAL de productos en tu ticket = 3

TOTAL en tu ticket = \$4948.14

*** PAGO ***

Por favor introduce tu número de tarjeta a 16 dígitos

1111222211112222

Por favor introduce la vigencia de tu tarjeta [mm/aa] separados por comas

11,11

Por favor introduce el Código de seguridad de tu tarjeta [CCV] a tres dígitos

0004

*** TICKET DE COMPRA ***

Ticket No. 1

Generado el Fri May 18 19:36:00 2024

Cliente Rodrigo

Dirección 232 Benito Juearez

PATIO

3 productos en el depto:

3pz Asador ARHOLMA - \$4948.14 TOTAL de
productos en tu ticket = 3

TOTAL en tu ticket = \$4948.14

PRUEBA 2

Hola! Bienvenido a la tienda en línea de IKEA

Cual es tu nombre? Luis Miguel

Excelente Luis Miguel, Podrías compartirnos tu dirección? Valle de mi corazón 601

De acuerdo, entonces tu domicilio es Valle de mi corazón 601 ahí se hará la
entrega de tus productos.

Antes de continuar revisemos tus datos :)

Luis Miguel

Valle de mi corazón 601

Estás de acuerdo? [s/n] n

Cual es tu nombre? **Luis Miguel Angel**

Excelente Luis Miguel Angel, Podrías compartirnos tu dirección? **Valle del alma 601**

De acuerdo, entonces tu domicilio es Valle del alma 601 ahí se hará la entrega de tus productos.

Antes de continuar revisemos tus datos :)

Luis Miguel Angel

Valle del alma 601

A continuación te presentamos los diferentes departamentos que tenemos en línea.

- 1) SALA
- 2) BAÑO
- 3) CUARTO
- 4) COCINA
- 5) PATIO
- 6) SALIR

Qué departamento deseas visitar? **1**

*** SALA ***

A Alfombra AGAM \$ 857.32

Suave y resistente alfombra de pelo corto.

B Butaca BILLY \$ 1623.47

Cómoda butaca tapizada con reposabrazos.

C Cojín CILLA \$ 459.63

Cojín decorativo con diseño moderno y elegante.

D Diván DAGSTORP \$ 2098.75

Diván acolchado con chaise longue reversible.

E Estantería EKET \$ 376.89

Estantería modular de diseño versátil y funcional.

F Felpudo FJÄLLA \$ 764.91

Felpudo de fibra de coco duradero y resistente.

G Gabinete GRÖNLID \$ 1332.58

Gabinete con puertas de vidrio y almacenamiento oculto.

Deseas agregar(a) o eliminar (e) un producto? **a**

Qué producto quisieras agregar a tu carrito? **f**

De acuerdo, y cuántos productos A quieres agregar? **4**

Agregaste 4 productos F a tu carrito

SALA

4 productos en el depto:

4pz Felpudo FJÄLLA - \$3059.64

Deseas continuar (c) editando tu carrito o salir(s)? [c/s] **c**

Deseas agregar(a) o eliminar (e) un producto? **e**

Qué producto quisieras eliminar de tu carrito? **f**

De acuerdo, y cuántos productos F quieres eliminar? 1

Eliminaste 1 productos F a tu carrito

SALA

3 productos en el depto:

3pz Felpudo FJÄLLA - \$2294.73

Deseas continuar (c) editando tu carrito o salir(s)? [c/s] s

A continuación te presentamos los diferentes departamentos que tenemos en línea.

1) SALA

2) BAÑO

3) CUARTO

4) COCINA

5) PATIO

6) SALIR

Qué departamento deseas visitar? 2

*** BAÑO ***

A Alfombra ATOFTBO \$ 987.65

Alfombra de baño suave y absorbente.

B Botiquín BROGRUND \$ 624.39

Botiquín de espejo con estantes ajustables.

C Cortina CRUNDTAL \$ 731.24

Cortina de ducha de tela resistente al agua.

D Dispensador DRAGAN \$ 847.50

Dispensador de jabón de bambú con diseño elegante.

E Espejo ENHET \$ 562.78

Espejo de baño con estante para almacenamiento.

F Felpudo FLODALEN \$ 145.36

Felpudo de baño de microfibra suave y absorbente.

G Ganchos GISKEN \$ 295.81

Ganchos de ventosa para colgar objetos en la ducha.

Deseas agregar(a) o eliminar (e) un producto? **a**

Qué producto quisieras agregar a tu carrito? **e**

De acuerdo, y cuántos productos E quieres agregar? **1** Agregaste 1
productos E a tu carrito

BAÑO

1 productos en el depto:

1pz Espejo ENHET - \$562.78

Deseas continuar (c) editando tu carrito o salir(s)? [c/s] s

A continuación te presentamos los diferentes departamentos que tenemos en línea.

1) SALA

2) BAÑO

3) CUARTO

4) COCINA

5) PATIO

6) SALIR

Qué departamento deseas visitar? 6

Bienvenido al área de checkout!

Revisemos tu ticket :)

SALA

3 productos en el depto:

3pz Felpudo FJÄLLA - \$2294.73

BAÑO

1 productos en el depto:

1pz Espejo ENHET - \$562.78

TOTAL de productos en tu ticket = 4

TOTAL en tu ticket = \$2857.51

*** PAGO ***

Por favor introduce tu número de tarjeta a 16 dígitos

444555333222

El número de tarjeta debe tener 16 dígitos. Por favor, inténtalo de nuevo.

Por favor introduce tu número de tarjeta a 16 dígitos

4445553332221111

Por favor introduce la vigencia de tu tarjeta [mm/aa] separados por comas

55,11

Por favor introduce el Código de seguridad de tu tarjeta [CCV] a tres dígitos

000

*** TICKET DE COMPRA ***

Ticket No. 84

Generado el Fri May 23 20:45:59 2024

Cliente Luis Miguel Angel

Dirección Valle del alma 601

SALA

3 productos en el depto:

3pz Felpudo FJÄLLA - \$2294.73

BAÑO

1 productos en el depto:

1pz Espejo ENHET - \$562.78

TOTAL de productos en tu ticket = 4

TOTAL en tu ticket = \$2857.51

PRUEBA 3

Hola! Bienvenido a la tienda en línea de IKEA

Cual es tu nombre? **Migue**

Excelente Migue, Podrías compartirnos tu dirección? **no** De acuerdo, entonces tu domicilio es no ahí se hará la entrega de tus productos.

Antes de continuar revisemos tus datos :) Migue

no

Estás de acuerdo? [s/n] **s**

A continuación te presentamos los diferentes departamentos que tenemos en línea.

- 1) SALA
- 2) BAÑO
- 3) CUARTO
- 4) COCINA
- 5) PATIO
- 6) SALIR

Qué departamento deseas visitar? **6**

Bienvenido al área de checkout!

Revisemos tu ticket :)

TOTAL de productos en tu ticket = 0

TOTAL en tu ticket = \$0.00

*** PAGO ***

Por favor introduce tu número de tarjeta a 16 dígitos

0000000000000000

El número de tarjeta debe tener 16 dígitos. Por favor, inténtalo de nuevo.

Por favor introduce tu número de tarjeta a 16 dígitos

1111222233334444

Por favor introduce la vigencia de tu tarjeta [mm/aa] separados por comas

22,11

Por favor introduce el Código de seguridad de tu tarjeta [CCV] a tres dígitos

000

*** TICKET DE COMPRA ***

Ticket No. 3

Generado el Fri May 17 11:36:00 2024

Cliente Migue

Dirección no

TOTAL de productos en tu ticket = 0

TOTAL en tu ticket = \$0.00

Análisis y comentarios del código:

Para fines de orden y claridad en esta parte, hemos decidido dividir el código para dar detalle de cada parte del programa para que destaquemos los detalles de lo que se realizó para que el programa **funcione y que no quede duda de cada paso realizado. A continuación, empezamos:**

PARTE I

El comprador podrá por medio del sistema:

- a) Agregar productos
- b) Quitar productos
- c) Se debe actualizar el número de productos (ejemplo: 2 salas, 1 cocina integral, etc.)
- d) El total de su compra (acumulado de su gasto)
- e) Total de productos comprados.

a) Agregar productos

La función `addprod()` permite al usuario agregar productos a su carrito. Esta función se llama dentro del bucle principal donde se presentan los departamentos y se solicita al usuario que seleccione productos.

b) Quitar productos

El código proporcionado no incluye la funcionalidad para quitar productos. De igual forma se puede implementar en un futuro.

c) Actualizar el número de productos

La actualización del número de productos se realiza dentro de la función `addprod()`, donde se solicita al usuario ingresar la cantidad de productos que desea agregar al carrito.

d) El total de su compra (acumulado de su gasto)

El cálculo del total de la compra se realiza al final del programa, donde se muestra sobre la matriz `ticket` y se multiplica la cantidad de cada producto por su precio correspondiente. El resultado se acumula en la variable `total_cost`.

e) Total de productos comprados

El total de productos comprados se calcula y se almacena en `ticket[0][4]`. Se actualiza cada vez que se agrega un producto al carrito.

PARTE II

El sistema mostrará el ticket con lo siguiente:

- a) Número de ticket
- b) Los artículos comprados y total por artículo
- c) El total gastado conforme va comprando (el acumulado de su gasto)
- d) El total de productos comprados

a) Número de ticket

El número de ticket se almacena en la matriz ticket en la primera columna para cada departamento. Por ejemplo, ticket[1][0] contiene el número de ticket para el departamento de la sala. Cuando se completa la compra, se calcula el total de productos comprados y se almacena en ticket[0][4]. Esta matriz se utiliza para imprimir el número de ticket en el área de checkout.

b) Los artículos comprados y total por artículo

La función checkticket(int n) imprime los artículos comprados y el total por artículo. Recibe como parámetro el número del departamento y recorre la matriz ticket para imprimir los artículos comprados, la cantidad y el total por artículo.

c) El total gastado conforme va comprando (el acumulado de su gasto)

El total gastado se calcula conforme se compra y se muestra al final del programa en el área de checkout. Se acumula en la variable total_cost, que se calcula sumando el precio de cada artículo comprado multiplicado por la cantidad.

d) El total de productos comprados

El total de productos comprados se calcula sumando la cantidad de productos de cada departamento y se almacena en ticket[0][4]. Este valor se utiliza para imprimir el total de productos comprados en el área de checkout.

Conclusión

El sistema generador de tickets de compra fue un proyecto bastante desafiante, ya que ponía a prueba todos los conocimientos adquiridos en la materia a lo largo del semestre. Sin embargo, al final logramos hacer que el programa funcione exitosamente y cumpliendo con todos los requerimientos iniciales. Este proyecto ha sido más que un simple trabajo para entregar, ha sido un viaje a través del vasto mundo del lenguaje C, donde hemos puesto en práctica y afianzado conceptos fundamentales.

Para no perdernos en el camino de la lógica del programa, empleamos los diagramas de flujo. Esto nos ayudó a representar el flujo de ejecución del programa, identificando claramente la secuencia de pasos y las decisiones a tomar en cada etapa.

Su uso contribuyó a la organización del código, la detección de posibles errores y, en última instancia, a la mejora de la legibilidad del programa.

Cabe destacar que, si bien el pseudocódigo del programa fue la columna vertebral de este proyecto, el diagrama de flujo ayuda al espectador a identificar mucho mejor y visualmente todo lo ocurrido en el programa y lo que realizamos.

Además de estos conceptos centrales, el proyecto nos brindó la oportunidad de poner en práctica otras habilidades esenciales para el desarrollo de software de calidad. La modularidad del código nos permitió dividir el programa en secciones funcionales e independientes, facilitando su comprensión y mantenimiento. También aprendimos a identificar nuestros propios errores, garantizando el funcionamiento correcto del sistema en escenarios inesperados. Realizamos constantemente pruebas de escritorio para detectar y corregir errores, y para identificar áreas de mejora. La entrada y salida de datos fueron aspectos cruciales para interactuar con el usuario y generar los tickets. Por último, la importancia de la documentación del programa se hizo evidente, ya que nos permitió registrar las decisiones tomadas, las funcionalidades implementadas y las consideraciones a tener en cuenta.

En definitiva, este proyecto ha sido una experiencia de aprendizaje valiosa que nos ha permitido aplicar y fortalecer los conocimientos adquiridos en el curso de fundamentos de programación. Por último, hemos demostrado la capacidad de utilizar los conceptos aprendidos para resolver problemas prácticos y crear aplicaciones útiles.

Código

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
#include <time.h>
```

```
float precios[5][7] = {  
    {857.32, 1623.47, 459.63, 2098.75, 376.89, 764.91, 1332.58},  
    {987.65, 624.39, 731.24, 847.50, 562.78, 145.36, 295.81},  
    {521.99, 875.43, 1096.82, 679.25, 234.56, 1500.73, 358.97},  
    {983.24, 1206.58, 376.54, 802.91, 653.27, 1147.89, 2364.75},  
    {1649.38, 2938.51, 548.79, 873.49, 724.16, 210.73, 999.99}};
```

```
char SALt[][150] = {"Alfombra AGAM",  
    "Butaca BILLY",  
    "Cojín CILLA",
```

"Diván DAGSTORP",
"Estantería EKET",
"Felpudo FJÄLLA",
"Gabinete GRÖNLID",
"Suave y resistente alfombra de pelo corto.",
"Cómoda butaca tapizada con reposabrazos.",
"Cojín decorativo con diseño moderno y elegante.",
"Diván acolchado con chaise longue reversible.",
"Estantería modular de diseño versátil y funcional.",
"Felpudo de fibra de coco duradero y resistente.",
"Gabinete con puertas de vidrio y almacenamiento oculto."};

char BANT[][150] = {"Alfombra ATOFTBO",
"Botiquín BROGRUND",
"Cortina CRUNDTAL",
"Dispensador DRAGAN",
"Espejo ENHET",
"Felpudo FLODALEN",
"Ganchos GISKEN",
"Alfombra de baño suave y absorbente.",
"Botiquín de espejo con estantes ajustables.",
"Cortina de ducha de tela resistente al agua.",
"Dispensador de jabón de bambú con diseño elegante.",
"Espejo de baño con estante para almacenamiento.",
"Felpudo de baño de microfibra suave y absorbente.",
"Ganchos de ventosa para colgar objetos en la ducha."};

char CUAt[][150] = {
"Almohada ANGSVIDE",
"Banco BRIMNES",

"Colchón COLDBROOK",
"Duvet DAGGRUTA",
"Escritorio EKEDALEN",
"Funda FJADRAR",
"Galán GUBBARP",
"Almohada ergonómica para un mejor descanso.",
"Banco con almacenamiento para el pie de la cama.",
"Colchón de espuma viscoelástica para un sueño confortable.",
"Edredón ligero y cálido para todas las estaciones.",
"Escritorio extensible con cajón de almacenamiento.",
"Funda de almohada suave y duradera.",
"Galán de noche con perchas para ropa."};

char COCt[150] = {"Abrelatas ANTAGEN",
"Bandeja BÄRBAR",
"Cortador CHOSIGT",
"Dispensador DROPPAR",
"Escurridor EKORRE",
"Frutero FÖRNUFT",
"Grifo GRANSKÄR",
"Abridor de latas de diseño ergonómico.",
"Bandeja de plástico resistente para servir.",
"Cortador de verduras multifuncional.",
"Dispensador de condimentos con tapa dosificadora.",
"Escurridor de platos plegable para ahorrar espacio.",
"Frutero de alambre de acero para organizar frutas.",
"Grifo de cocina con ducha extraíble y giratoria."};

char PATt[150] = {"Asador ARHOLMA",
"Banco de jardín BONDHOLMEN",

"Cajonera CHAPLIN",
"Dosel DYNING",
"Estera EKERÖN",
"Farol FÖRMEDLA",
"Gazebo GÄLÖ",
"Asador portátil de acero inoxidable.",
"Banco de madera para exteriores.",
"Caja de almacenamiento resistente a la intemperie.",
"Dosel para protección solar con estructura de aluminio.",
"Estera de plástico tejido para proteger el suelo.",
"Farol de metal para iluminar el patio por la noche.",
"Gazebo plegable con cortinas para crear sombra."};

void strprnt(char n[]);

void separador();

void SALmenu(int c);

void BANmenu(int c);

void CUAMenu(int c);

void COCmenu(int c);

void PATmenu(int c);

int modprod(int *p, int *q);

void ticketprnt();

void checkticket(int n);

int ticket[6][8];

int main() {

char nombre[150], direc[150];

long numtar;

```
int flag, depto, prod = 0, cant = 0;
```

```
int vigmes, vigano, ccv, count;
```

```
char opc, mas, num;
```

```
char Abc[][6] = {"A", "B", "C", "D", "E", "F", "G"};
```

```
for (int i = 0; i < 5; i++) {
```

```
    for (int j = 0; j < 7; j++) {
```

```
        ticket[i][j] = 0;
```

```
    }
```

```
}
```

```
printf("\nHola! Bienvenido a la tienda en línea de IKEA\n\n");
```

```
opc = 'n';
```

```
do {
```

```
    fflush(stdin);
```

```
    printf("\nCual es tu nombre?\t");
```

```
    gets(nombre); // PROBLEMA CON LOOP
```

```
    printf("\nExcelente ");
```

```
    strprnt(nombre);
```

```
    printf(", Podrías compartirmos tu dirección?\t");
```

```
    gets(direc);
```

```
    printf("\n\nDe acuerdo, entonces tu domicilio es ");
```

```
    strprnt(direc);
```

```
    printf(",\nahí se hará la entrega de tus productos.\n\n");
```

```
    separador();
```



```
printf("\n\nAntes de continuar revisemos tus datos :)\n\n");
```

```
strprnt(nombre);
```

```
printf("\n");
```

```
strprnt(direc);
```

```
printf("\n\nEstás de acuerdo? [s/n]\t");
```

```
scanf(" %c", &opc);
```

```
printf("\n\n");
```

```
} while (opc == 'n'); // No pide el nombre cuando da loop
```

```
do {
```

```
    separador();
```

```
    separador();
```

```
printf("\n\nA continuación te presentamos los diferentes departamentos que "  
    "tenemos en línea.");
```

```
printf("\n\n1) SALA\n2) BAÑO\n3) CUARTO\n4) COCINA\n5) PATIO\n6) SALIR");
```

```
printf("\n\nQué departamento deseas visitar?\t");
```

```
scanf("%i", &depto);
```

```
printf("\n\n");
```

```
opc = 'n';
```

```
switch (depto) {
```

```
case 1:
```

```
    separador();
```

```
    printf("\n\n\t\t\t*** SALA ***\n\n");
```

```

for (int c = 0; c < 7; c++) {
    printf("%s\t", Abc[c]);
    SALmenu(c);
    printf("\t\t$ %.2f\n\t", precios[depto - 1][c]);
    SALmenu(c + 7);
    printf("\n\n");
}

do {
    modprod(&prod, &cant);
    ticket[depto][prod] = ticket[depto][prod] + cant;
    num = ticket[depto][prod];

    if (num < 0) {
        ticket[depto][prod] = 0;
    }

    ticket[depto][0] = ticket[depto][0] + cant;
    ticket[0][5] =
        ticket[0][5] + ticket[depto][prod] * precios[depto - 1][prod - 1];

    separador();
    printf("\n");
    checkticket(depto);
    separador();

    printf(
        "\n\nDeseas continuar (c) editando tu carrito o salir(s)? [c/s]\t");
    scanf(" %c", &opc);
} while (opc != 's');

```

```
break;
```

case 2:

```
separador();
```

```
printf("\n\n\t\t\t*** BAÑO ***\n\n");
```

```
for (int c = 0; c < 7; c++) {
```

```
    printf("%s\t", Abc[c]);
```

```
    BANmenu(c);
```

```
    printf("\t\t$ %.2f\n\t", precios[depto - 1][c]);
```

```
    BANmenu(c + 7);
```

```
    printf("\n\n");
```

```
}
```

```
do {
```

```
    modprod(&prod, &cant);
```

```
    ticket[depto][prod] = ticket[depto][prod] + cant;
```

```
    num = ticket[depto][prod];
```

```
    if (num < 0) {
```

```
        ticket[depto][prod] = 0;
```

```
    }
```

```
    ticket[depto][0] = ticket[depto][0] + cant;
```

```
    ticket[0][5] =
```

```
        ticket[0][5] + ticket[depto][prod] * precios[depto - 1][prod - 1];
```

```
    separador();
```

```
printf("\n");  
checkticket(depto);  
separador();
```

```
printf(  
    "\n\nDeseas continuar (c) editando tu carrito o salir(s)? [c/s]\t");  
scanf(" %c", &opc);  
} while (opc != 's');
```

```
break;
```

case 3:

```
separador();  
printf("\n\n\t\t\t*** CUARTO ***\n\n");
```

```
for (int c = 0; c < 7; c++) {  
    printf("%s\t", Abc[c]);  
    CUAMenu(c);  
    printf("\t\t$ %.2f\n\t", precios[depto - 1][c]);  
    CUAMenu(c + 7);  
    printf("\n\n");  
}
```

```
do {  
    modprod(&prod, &cant);  
    ticket[depto][prod] = ticket[depto][prod] + cant;  
    num = ticket[depto][prod];
```

```
if (num < 0) {
```

```
    ticket[depto][prod] = 0;  
}
```

```
ticket[depto][0] = ticket[depto][0] + cant;  
ticket[0][5] =  
    ticket[0][5] + ticket[depto][prod] * precios[depto - 1][prod - 1];
```

```
separador();  
printf("\n");  
checkticket(depto);  
separador();
```

```
printf(  
    "\n\nDeseas continuar (c) editando tu carrito o salir(s)? [c/s]\t");  
scanf(" %c", &opc);  
} while (opc != 's');
```

```
break;
```

case 4:

```
separador();  
printf("\n\n\t\t\t\t\t*** COCINA ***\n\n");
```

```
for (int c = 0; c < 7; c++) {  
    printf("%s\t", Abc[c]);  
    COCmenu(c);  
    printf("\t\t$ %.2f\n\t", precios[depto - 1][c]);  
    COCmenu(c + 7);  
    printf("\n\n");
```

```
}
```

```
do {
```

```
    modprod(&prod, &cant);
```

```
    ticket[depto][prod] = ticket[depto][prod] + cant;
```

```
    num = ticket[depto][prod];
```

```
    if (num < 0) {
```

```
        ticket[depto][prod] = 0;
```

```
    }
```

```
    ticket[depto][0] = ticket[depto][0] + cant;
```

```
    ticket[0][5] =
```

```
        ticket[0][5] + ticket[depto][prod] * precios[depto - 1][prod - 1];
```

```
    separador();
```

```
    printf("\n");
```

```
    checkticket(depto);
```

```
    separador();
```

```
    printf(
```

```
        "\n\nDeseas continuar (c) editando tu carrito o salir(s)? [c/s]\t");
```

```
    scanf(" %c", &opc);
```

```
    } while (opc != 's');
```

```
    break;
```

```
case 5:
```

```
    separador();
```

```
printf("\n\n\t\t\t\t\t*** SALA ***\n\n");
```

```
for (int c = 0; c < 7; c++) {  
    printf("%s\t", Abc[c]);  
    PATmenu(c);  
    printf("\t\t$ %.2f\n\t", precios[depto - 1][c]);  
    PATmenu(c + 7);  
    printf("\n\n");  
}
```

```
do {  
    modprod(&prod, &cant);  
    ticket[depto][prod] = ticket[depto][prod] + cant;  
    num = ticket[depto][prod];
```

```
    if (num < 0) {  
        ticket[depto][prod] = 0;  
    }
```

```
    ticket[depto][0] = ticket[depto][0] + cant;  
    ticket[0][5] =  
        ticket[0][5] + ticket[depto][prod] * precios[depto - 1][prod - 1];
```

```
    separador();  
    printf("\n");  
    checkticket(depto);  
    separador();
```

```
    printf(  
        "\n\nDeseas continuar (c) editando tu carrito o salir(s)? [c/s]\t");
```

```

        scanf(" %c", &opc);
    } while (opc != 's');

    break;
}
} while (depto != 6);

ticket[0][4] =
    ticket[1][0] + ticket[2][0] + ticket[3][0] + ticket[4][0] + ticket[5][0];

float total_cost = 0.0;
for (int i = 1; i <= 5; i++) {
    for (int j = 1; j <= 7; j++) {
        total_cost += ticket[i][j] * precios[i - 1][j - 1];
    }
}

separador();

printf("\n\nBienvenido al área de checkout!\n");
printf("Revisemos tu ticket :)\n\n");

for (int c = 1; c <= 5; c++) {
    checkticket(c);
}

printf("\n\n\t\t");
printf("TOTAL de productos en tu ticket = %i", ticket[0][4]);
printf("\n\t\t");
printf("TOTAL en tu ticket = $%.2f", total_cost);

```



```

separador();
printf("\n\n\t\t\t*** PAGO ***\n\n");
separador();

do {
    printf("\nPor favor introduce tu número de tarjeta a 16 dígitos\n\t");

    scanf("%ld", &numtar);

    int count = 0;
    long temp = numtar;

    while (numtar != 0) {
        numtar = numtar / 10;
        count++;
    }
    if (count == 16)
        break;
    else {
        printf("\t\t\tEl número de tarjeta debe tener 16 dígitos. Por favor, "
            "inténtalo de nuevo.\n");
        fflush(stdin);
    }
} while (1);

printf("\nPor favor introduce la vigencia de tu tarjeta [mm/aa] separados "
    "por comas\n\t");
scanf("%i,%i", &vigmes, &vigano);

printf("\nPor favor introduce el Código de seguridad de tu tarjeta [CCV] a "

```

```
        "tres dígitos\n\t");
scanf("%i", &ccv);

printf("\n\n\n\n");
separador();
printf("\n\n\t\t\t*** TICKET DE COMPRA ***\n\n");
separador();

printf("\nTicket No.");
printf(" %d%d", rand(), rand());

time_t rawtime;
struct tm *timeinfo;

time(&rawtime);
timeinfo = localtime(&rawtime);
printf("\nGenerado el ");
printf("%s", asctime(timeinfo));

printf("\nCliente\t");
strprnt(nombre);

printf("\nDirección\t");
strprnt(direc);

printf("\n\n");

for (int c = 1; c <= 5; c++) {
    checkticket(c);
}
```

```
printf("\n\n\t\t");
printf("TOTAL de productos en tu ticket = %i", ticket[0][4]);
printf("\n\t\t");
printf("TOTAL en tu ticket = $%.2f", total_cost);
}
// FIN CODIGO
```

```
// FUNCIONES
```

```
void strprnt(char n[]) {
    for (int c = 0; c < strlen(n); c++) {
        printf("%c", n[c]);
    }
}
```

```
void separador() {
    printf("\n* - * - * - * - * - * - * - * - *\n");
}
```

```
void SALmenu(int c) { printf("%s\t", SALT[c]); }
```

```
void BANmenu(int c) { printf("%s\t", BANT[c]); }
```

```
void CUAmenu(int c) { printf("%s\t", CUAT[c]); }
```

```
void COCmenu(int c) { printf("%s\t", COCT[c]); }
```

```
void PATmenu(int c) { printf("%s\t", PATt[c]); }
```

```
int modprod(int *p, int *q) {
```

```
char c, o;

printf("\nDeseas agregar(a) o eliminar (e) un producto?\t");
scanf(" %c", &o);

while (o != 'a' && o != 'e') {
    printf("\n[a] Agregar / [e] Eliminar\t");
    scanf(" %c", &o);
}

if (o == 'a')
    printf("\nQué producto quisieras agregar a tu carrito?\t");
else
    printf("\nQué producto quisieras eliminar de tu carrito?\t");

scanf(" %c", &c);

c = toupper(c);

if (c >= 'A' && c <= 'G') {
    *p = c - 'A' + 1;

    if (o == 'a')
        printf("\nDe acuerdo, y cuántos productos %c quieres agregar?\t", c);
    else
        printf("\nDe acuerdo, y cuántos productos %c quieres eliminar?\t", c);

    scanf(" %i", &q);

    if (o == 'e')
```

```
*q = -(*q);
```

```
if (o == 'a')
```

```
    printf("\n\tAgregaste %i productos %c a tu carrito\n\n", *q, c);
```

```
else
```

```
    printf("\n\tEliminaste %i productos %c a tu carrito\n\n", abs(*q), c);
```

```
}
```

```
else
```

```
    printf("\nCaracter no válido. Debe ser un caracter de la A a la G.\n");
```

```
}
```

```
void ticketprnt() {
```

```
    for (int i = 0; i < 6; i++) {
```

```
        for (int j = 0; j < 8; j++) {
```

```
            printf("%i ", ticket[i][j]);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
}
```

```
void checkticket(int n) {
```

```
    int costo;
```

```
    char noms[][10] = {"*SALA*", "*BAÑO*", "*CUARTO*", "*COCINA*", "*PATIO*"};
```

```
    printf("\t%s", noms[n - 1]);
```

```
    printf("\n%i productos en el depto:\n", ticket[n][0]);
```

```
    printf("\t");
```

```

for (int i = 1; i < 7; i++) {
    printf("%ipz ", ticket[n][i]);
    switch (n) {
    case 1:
        printf("%s ", SALt[i - 1]);
        break;
    case 2:
        printf("%s ", BANT[i - 1]);
        break;
    case 3:
        printf("%s ", CUAt[i - 1]);
        break;
    case 4:
        printf("%s ", COCt[i - 1]);
        break;
    case 5:
        printf("%s ", PATt[i - 1]);
        break;
    }

    printf("- $%.2f  ", (precios[n - 1][i - 1] * ticket[n][i]));
    if (i == 2 || i == 4 || i == 6)
        printf("\n\t");
    }
    printf("\n\n");
}

```

Repositorio Github

Todos los documentos relacionados a este código (inclusive este mismo documento) estarán en el siguiente repositorio: <https://github.com/CuellarJM/ProyectoFinalFP>

Referencias

Rancel, M. R. (s. f.). Funciones en C ¿qué significa void? ¿Qué es el tipo de retorno? ¿Para qué sirve return? Módulos (CU00547F). aprenderaprogramar.com.
https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=947:funciones-en-c-ique-significa-void-ique-es-el-tipo-de-retorno-ipara-que-sirve-return-modulos-cu00547f&catid=82&Itemid=210

Programación en C. (s. f.).
http://maxus.fis.usal.es/FICHAS_C.WEB/01xx_PAGS/0103.html

TylerMSFT. (2023b, octubre 12). gets, _getws. Microsoft Learn.
<https://learn.microsoft.com/es-es/cpp/c-runtime-library/gets-getws?view=msvc-170>

TylerMSFT. (2023a, abril 3). fflush. Microsoft Learn.
<https://learn.microsoft.com/es-es/cpp/c-runtime-library/reference/fflush?view=msvc-170>

¿Qué es fflush (stdin) en el lenguaje de programación C? ¿Cuándo usará este código? (s. f.). Quora.
<https://es.quora.com/Qu%C3%A9-es-fflush-stdin-en-el-lenguaje-de-programaci%C3%B3n-C-Cu%C3%A1ndo-usar%C3%A9-este-c%C3%B3digo>

Algoritmos y programación. (s. f.).
<https://lab.anahuac.mx/~hselley/ayp/arreglos.html#:~:text=Un%20arreglo%20unidimensional%20es%20un,datos%20solicitados%20por%20el%20programador.>

Rancel, M. R. (s. f.-a). Arrays o arreglos multidimensionales en C. Matrices. Ejemplos y ejercicios resueltos varios índices (CU00515F). aprenderaprogramar.com.
https://aprenderaprogramar.com/index.php?option=com_content&view=article&id=904:arrays-o-arreglos-multidimensionales-en-c-matrices-ejemplos-y-ejercicios-resueltos-s-varios-indices-cu00515f&catid=82&Itemid=210

Multidimensional Arrays - Learn C - Free Interactive C tutorial. (s. f.).
<https://www.learn-c.org/es/Multidimensional%20Arrays>

Adrián, D. R. H. (2022a, agosto 22). Ficheros - Abrir, cerrar, leer y agregar archivos de texto. Diseño Web akus.net. <https://disenowebakus.net/ficheros.php>

valor crítico Archivos - Viva el Software Libre. (s. f.). Viva el Software Libre. <https://vivaelssoftwarelibre.com/tag/valor-critico/>