

Instituto Tecnológico de Costa Rica
Sede Regional San Carlos

Escuela de Computación

Lenguajes de programación

Proyecto 1

Responsables:
Crisly González Sánchez

19 de abril del 2016

Santa Clara, San Carlos

Introducción

El manejo de memoria es un recurso que se debe distribuir de manera eficiente e inteligente, una mala asignación de memoria en algunos lenguajes de desarrollo que lo requieren, puede causar que el programa en proceso de desarrollo no funcione. Cabe destacar que algunos lenguajes de programación no es necesario la asignación de recursos de memoria, es por esto que se debe investigar la sintaxis y modo de uso de los recursos para asignar correctamente lo que corresponda a utilizar.

El proyecto desarrollado se realizó en el lenguaje de programación C, este lenguaje es caracterizado porque se requiere una minuciosa asignación de los recursos para que sus programas funcionen además no es un lenguaje de alto nivel y propicia el uso de estructuras.

El paradigma utilizado es imperativo, el cual se basa en describir la programación de acuerdo al estado de programa, los elementos más importantes de este paradigma son: variables, tipos de datos y expresiones.

Análisis del Problema

Se requiere la utilización de archivos de texto para el almacenamiento de los productos que posee la tienda para que sean cotizados, esos archivos contarán con información previamente insertada por el programador para agilizar procesos. Es importante conocer cómo se almacenará la información en el archivo, para asimismo hacer la lectura en él.

El programa requiere que el cliente vea los productos que tiene la tienda mediante sistema de cotización, para darle solución es requerido que contemos con estructuras de productos, que tenga reservado en memoria los atributos más relevantes que se deseen conocer por parte del cliente, en los cuales podemos enumerar descripción, cantidad de productos, precio de producto en dólar e identificador.

Teniendo una estructura de productos, podemos almacenar en una lista general de stock todos los productos con los que cuenta la tienda, así será más sencillo mostrarlos al usuario y tener un control de los mismos.

Además el usuario requiere solicitar una cotización, ingresar el nombre del negocio, ubicación y nombre del usuario y seleccionar al menos 9 piezas básicas de las cuales son las siguientes Memoria, Tarjeta Madre, HDD, Video, Teclado, Mouse, Monitor y Torre., es importante mencionar que la cotización no debe llevar piezas repetidas sean básicas o no.

Se requiere una estructura para la generación de facturas y asimismo una lista donde sean almacenadas, los atributos a considerar para formar estructura de factura es nombre de negocio, ubicación, cliente, precio dolar, una lista de productos. Como requisito no funcional se incluye que dicha factura debe ser impresa en consola y en archivo html.

Para cumplir con los requisitos no funcionales se requiere la implementación de archivos con estructura html, que permitan el ingreso de la información generada en la consola. Además el programa requiere que se controle el tipo de cambio del dólar y este pueda ser modificable, es por esto que debe ser lo primero a elegir para poder darle valor a una variable global que debe ser creada con el fin de que los precios en dólares puedan ser mostrados en colores en la factura.

Y finalmente para que la factura se genere correctamente se requiere un método que compruebe que los productos que están siendo cotizados no están repetidos, porque de ser así debe pedirlo uno nuevamente.

En la imagen podemos ver la información y la estructura de la factura que el negocio solicita que sea programada en C y se muestre en html.

Mercado PC, S.A.
San Carlos, Costa Rica
Cliente: Oscar Víquez A

* * * C O T I Z A C I O N * * *
Fecha: 25 de Agosto del 2004

Item	Descripción	Valor US\$	Colones
1	Tarjeta Madre ASUS	97	41710
2	Pentium IV 2.4 GHz	120	51600
3	Memoria 256 DIMM	50	21500
4	Disco Duro 80 Gb, Maxtor	65	27950
5	Tarjeta Video GFORCE4	100	43000
6	Tarjeta Sonido SB Live	52	22360
7	Fax MODEM	15	6450
8	Torre ATX 250W	40	17200
9	Parlantes Altec Lansing	70	30100
10	CD-ROM Creative 52X	29	12470
11	Monitor AOC 19'	113	48590
12	Mouse Optico Logitec	14	6020
13	Teclado Microsoft 101 teclas	21	9030

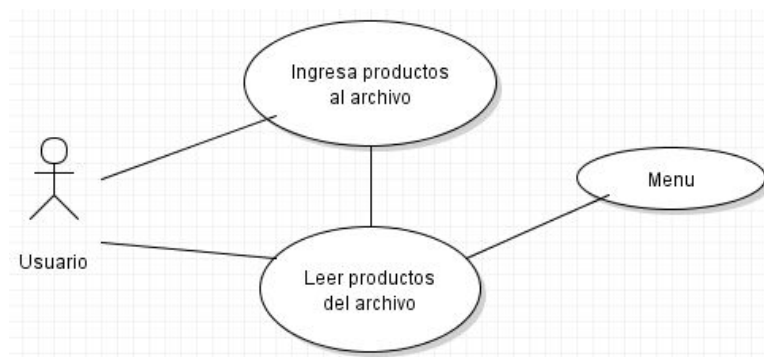
Total US\$ 786
Total Colones 337980
IVI (13%) 43937.4
Valor Total 381917.4

Solución del problema

Se implementaron archivos que guardan estructuras de productos, debido a eso se requiere que para ingresar un producto adicional de los que ya están almacenados se debe ingresar el producto desde consola, los archivos de estructuras almacenan la informacion de numeros en binario por esto no se pueden insertar desde el archivo y leerlo luego.

El programa de cotización posee un método que ingresa los productos que se quieren registrar en el stock y posteriormente uno de lectura que retorna la información y la almacena en una lista de productos en memoria una vez que se inicia el programa.

El procedimiento una vez que se inicia el programa es así:



Para almacenar productos en un archivo o mostrarlos en la lectura en memoria se requirió asignar memoria para cada atributo mínimo que íbamos a conocer de los productos: id, descripción, cantidad de productos, precio dolar, el precio en colones no se almacena en cada producto porque a partir de precio en dolar podemos conocerlo.

Una vez que se ingresaron y leyeron los productos con sus estructuras ya definidas se almacenan los productos previos del archivo en una lista en memoria. El procedimiento fue leerlos y construir el nodo de producto e insertarlo a la lista productos a través del método insertar productos, este método se encarga de ingresarlos a la lista recibiendo la información de los atributos.

El tipo de cambio de dólar fue resuelto mediante una variable global que cambia su valor una vez que se ingrese el programa y el usuario le de valor, es importante que sea cambiando antes de realizar la cotización de los productos para dar un resultado en colones más exacto. El impuesto fue regido por el 13% ya que no se indico en la documentación previa proporcionada por el cliente si debía ser modificable.

El procedimiento de realización de la cotización es fijar previamente el valor del tipo de dólar, y posteriormente ingresar el nombre del negocio, ubicación y nombre del cliente. La fecha se obtiene de la PC a través de la librería `time.h` y `conio.h`.

Antes de ingresar los productos el usuario escoge la cantidad que desea ingresar, si la cantidad es menor al mínimo de productos que debe ingresar como pieza básica es decir 9 debe volver a ingresar la cantidad nuevamente. Luego se imprimen los productos almacenados y se podrá escoger por ID los que desee cotizar, en caso de que sea repetido lo rechaza y habilita para ingresar un ID diferente. Este procedimiento de comprobación es posible porque cada vez que se ingresa un id se envía a una lista de identificadores, donde comprueba que no esté ingresado anteriormente.

Una vez que los productos ya se reservaron se construye la factura con una lista de productos de detalle de factura, estos son impresos tal como se ingresaron y se calcula el precio en colones, el impuesto del 13%, fecha y el Total. Luego se envían los datos para que el archivo tipo `html` construya su factura.

En el archivo `html` se deben aplicar las función `itoa` para convertir `int` a `string` ya que los archivos `html` solo muestran información de tipo `string`.

Finalmente una vez impresa la lista de cotización en la misma carpeta automáticamente se construye un archivo `html` que tendrá la misma información y estará disponible para ejecutarse en el browser.

Análisis de resultados

Tarea	Estado	Observaciones
Creación / Lectura de archivos	100%	archivos implementados son de estructuras
Ingreso minimo de 9 productos básicos	100%	
Lista de productos en memoria	100%	
Implementación de tipo de cambio de dólar	100%	
Estructura de Cotización/ Productos	100%	
Generación de una cotización	100%	
Implementación de fecha en la cotización	100%	
Generación de archivo HTML	100%	
Datos previamente insertados	100%	
Cálculo del impuesto, total de compra, precio dólar y precio colones.	100%	

Conclusión

El trabajo de estructuras con punteros requiere que el programador sea cuidadoso de la asignación de memoria, para evitar pérdida de información y segmentation fault. A su vez es importante saber que por la falta de un garbage collector es necesario liberar memoria a cualquier estructura que se le haya asignado antes.

Bootstrap es una herramienta que permite crear mejores interfaces gráficas para el usuario, esta herramienta fue utilizada para la creación de la cotización que el sistema debe generar de manera automática.

Recomendaciones

- ❖ Especificación detallada de los requisitos funcionales y no funcionales de la aplicación para cotizaciones
- ❖ Interfaz atractiva, diferente de la consola de IDE utilizado
- ❖ Mayor utilización del archivo HTML, implementación de diseño personalizado y atractivo.
- ❖ Valorar si es mejor cambiar los archivos de estructuras por los de texto en lenguaje C.
- ❖ Especificar cómo debe ser el proceso de cotización, para que sea adaptado a las necesidades del cliente.
- ❖ Especificar cómo se agruparán los productos de que sean de igual función pero diferente marca.

Bibliografía

dccia. (2010/2011). Lenguajes y Paradigmas de Programación. 14/03/2016, de SF Sitio web: <http://www.dccia.ua.es/dccia/inf/asignaturas/LPP/2010-2011/clases-cristina/s2.pdf>

carlosmoralestic. (28/11/2004). Paradigma Imperativo y características. 24/03/2016, de SR Sitio web: <https://carlostics.wordpress.com/2014/11/28/paradigma-imperativo-y-caracteristicas/>

SR. (SR). C itoa() function. 24/03/2016, de javakode Sitio web: <http://www.javakode.com/c-programming/c-type-casting/c-itoa-function/>