|  |  |  |
| --- | --- | --- |
| Imagen que contiene dibujo, taza  Descripción generada automáticamente | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Ingeniería Karina García Morales |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 1120 |
| *No de Práctica:* | *6º* |
| *Integrante:* | Yushua Haza Jibaja |
| *No. de Equipo de cómputo empleado:* |  |
| *No. de Lista o Brigada:* | 22 |
| *Semestre:* | 2022 - 1 |
| *Fecha de entrega:* | 28 de septiembre de 2021 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**Objetivo:**

El alumno elaborará programas en lenguaje C utilizando las instrucciones de control de

tipo secuencia, para realizar la declaración de variables de diferentes tipos de datos, así

como efectuar llamadas a funciones externas de entrada y salida para asignar y mostrar

valores de variables y expresiones.

**Actividades:**

● Crear un archivo de texto (utilizando algún editor) y escribir un programa en

lenguaje C que contenga variables de diferentes tipos, asignación de valores

(por lectura desde la entrada estándar o asignación directa) y escritura del

valor de las variables en la salida estándar

● Compilar un código fuente y ejecutarlo.

● Modificar y actualizar un programa usando un editor.

● Elaborar expresiones relacionales/lógicas en un programa en C y mostrar el

resultado de su evaluación.

Introducción

Cuando un problema ya fue analizado, se diseñó un algoritmo para su resolución y que se haya representado dicho algoritmo de manera gráfica y escrita (pseudocódigo) se puede proceder a la codificación en C++

La codificación se lleva a cabo en la mayoría de los lenguajes de programación estructurada, en este caso se está usando C++.

Etapas fundamentales:

-Edición del código fuente

-Compilación

-Ejecución

Entorno de C++

Un lenguaje de programación permite traducir instrucciones que puede pensar un ser humano y las traduce al lenguaje de las computadoras en alguna de sus variantes, como es este caso el lenguaje C++

C++ es muy útil porque combina las características de un lenguaje de alto nivel con las de uno de bajo nivel, por lo que es bastante efectivo para programar instrucciones para computadoras y aparatos electrónicos a nivel mundial ya que es un lenguaje más fácil de aprender, interpretar y comprender.

Un algoritmo o programa en C++ se realiza desarrollando cada bloque de instrucciones de acuerdo con las reglas establecidas como palabras reservadas o uso de signos para poder llevar a cabo una compilación, que es el proceso de comprobación de errores de interpretación del código hacia la máquina. Un compilador toma una entrada de texto (el propio código) y produce un programa ejecutable, haciendo posible que tanto hardware como software lo puedan ejecutar. El lenguaje C++ o C simple, tiene la característica de ser portable, es decir, cualquier máquina con un compilador o interpretador de C o C++, puede ejecutar los programas/algoritmos desarrollados en dichos lenguajes, a esta característica se le denomina también multiplataforma.

Editores

Un editor es un programa de edición de texto simple que puede desarrollar el código necesario para un compilador de lenguaje C++, estos van de la mano porque son software indispensable para llevar a cabo el desarrollo de dicho programa, además de que deben poseer una interfaz gráfica que se pueda observar para el entendimiento del usuario, una interfaz gráfica es la visualización del texto, la colocación de los íconos, los colores y la manera en que todos estos elementos interactúan unos con otros.

Es importante mencionar que no es lo mismo un editor de texto que un procesador de texto, un editor edita un texto básico que puede tener muchas utilidades como guardar una configuración o tener escrito un programa y será interpretado hasta que se haga una lectura de éste, un procesador de texto permite desarrollar texto e imágenes de una manera distinta, pues puede ingresar diferentes tipos de letra, tamaños, visuales, márgenes, agregados como formas geométricas o gráficas, etc.

**Editor Visual Interface de GNU/Linux (vi)**

“El editor vi (visual interface) es el editor más común en cualquier distribución de sistemas operativos con núcleo basado en UNIX. Está disponible en línea de comandos y si el sistema operativo tiene entorno gráfico se puede acceder a él desde la terminal. vi es un editor que puede resultar difícil de usar en un inicio. Aunque existen editores más intuitivos en su uso; en muchas ocasiones vi es el único disponible. Para iniciar vi, debe teclearse desde la línea de comandos:

vi nombre\_archivo[.ext] Donde nombre\_archivo es el nombre del archivo a editar o el nombre de un archivo nuevo que se creará con vi, y [.ext] se refiere a la extensión que indica que el texto es una programa escrito en algún lenguaje o es texto plano, por ejemplo. Es válido incluir la ruta donde se localiza o localizará el archivo. Existen más métodos de apertura para usuarios más avanzados.” (FACULTAD DE INGENIERÍA,2021)

Modo comando (Figura 2)

Es el modo por defecto de vi cuando se abre. Las teclas presionadas ejecutan diversas

acciones predeterminadas y no se puede editar el texto libremente. Los comandos son

sensitivos a las mayúsculas y a las minúsculas. Algunos ejemplos son:

● ↑ o k mueve el cursor hacia arriba.

● ↓ o j mueve el cursor hacia abajo.

● ← o h mueve el cursor hacia la izquierda.

● → o l mueve el cursor hacia la derecha.

● 1G lleva el cursor al comienzo de la primera línea.

● G lleva el cursor al comienzo de la última línea.

● x borra el carácter marcado por el cursor.

● dd borra o corta la línea donde está el cursor.

● ndd donde n es la cantidad de líneas que se borrarán o cortarán después del

cursor.

● D borra o corta desde la posición de cursor hasta el final de la línea.

● dw borra o corta desde la posición del cursor hasta el final de una palabra.

● yy copia la línea donde está el cursor.

● p pega un contenido copiado o borrado.

● u deshace el último cambio.

**Modo de última línea**

Se puede acceder a él utilizando el modo comando, pero los comandos no tendrán efecto hasta que se presiona la tecla Enter además de que se visualizará el comando en la última línea del editor. Es posible cancelar el comando con la tecla Esc. Los comandos de última línea se caracterizan porque inician con /, ? o :. Algunos ejemplos son:

● /texto donde la cadena texto será buscada hacia delante de donde se

encuentra el cursor.

● ?texto donde la cadena texto será buscada hacia atrás de donde se encuentra

el cursor.

● :q para salir de vi sin haber editado el texto desde la última vez que se guardó.

● :q! para salir de vi sin guardar los cambios.

● :w para guardar los cambios sin salir de vi.

● :w archivo para realizar la orden “guardar como”, siendo archivo el nombre

donde se guardará el documento.

● :wq guarda los cambios y sale de vi.

Es un editor de texto que funciona para computadoras con sistema operativo base UNIX en línea de comandos. Se puede acceder a la interfaz gráfica con una aplicación de terminal. Este editor es más intuitivo que “vi” pero menos potente, lo único que se requiere de este es la interfaz que describe los comandos básicos

Para poder iniciar nano, se debe usar el comando “nano nombre\_archivo[.ext]” donde nombre del archivo es el nombre del propio archivo y si este esta separado por carácter de espacio, se sustituye por un guion bajo.

Una vez en el editor, en la parte inferior se pueden observar los comandos básicos. Si

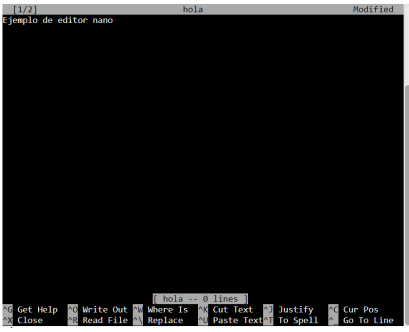
se presiona la tecla F1 es posible visualizar la ayuda con la lista de todos comandos que

existen.

Los atajos de teclado pueden corresponder a:

● ^ que es la tecla Ctrl.

● M- que es la tecla Esc o bien Alt.



**Otros**

Actualmente existen más editores de texto como lo son:

-Visual Studio Code

-Atom, Sublime Text

-Notepad

-entre otros.

**Compiladores**

Son softwares que traducen un programa escrito en un lenguaje de programación de alto nivel (C / C ++, COBOL, etc.) en lenguaje de máquina. Un compilador generalmente genera lenguaje ensamblador primero y luego traduce el lenguaje ensamblador al lenguaje máquina. Una utilidad conocida como «enlazador» combina todos los módulos de lenguaje de máquina necesarios en un programa ejecutable que se puede ejecutar en la computadora.

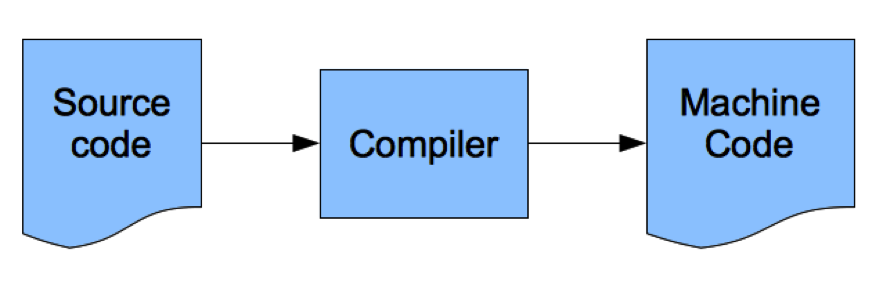


Figura TAL “Diagrama de flujo simple del compilador”

<https://www.europeanvalley.es/noticias/que-es-un-compilador-en-programacion/>

**gcc (GNU Compiler Collection)**

“Es un conjunto de compiladores de uso libre para sistemas operativos basados en UNIX. Entre sus compiladores existe el que sirve para programas escritos en C. Se encuentra por defecto en diversas distribuciones de GNU/Linux. El compilador trabaja en línea de comandos.

Existe también una versión modificada que puede ejecutar y crear programas para plataformas Windows en un paquete llamado MinGW (Minimalist GNU for Windows). Al compilar un programa en C el compilador genera diversos archivos intermedios que corresponden a las distintas fases que realiza. Éstas no son de interés por el momento y son eliminadas una vez obtenido el archivo ejecutable. gcc tiene diferentes opciones de ejecución para usuarios más avanzados.” (FACULTAD DE INGENIERÍA,2021)

**Entorno de Desarrollo Integrado**

Los IDE están diseñados para maximizar la productividad del programador

proporcionando componentes muy unidos con interfaces de usuario similares. Los IDEpresentan un único programa en el que se llevan a cabo las diversas etapas de desarrollo de software. Generalmente, este programa suele ofrecer muchas

características para la creación, modificación, compilación, implementación y

depuración de software. Esto contrasta con el desarrollo de software utilizando

herramientas no relacionadas, como vi, GNU Compiler Collection (gcc) o make.

-Dev-C++: Este emplea el compilador MinGW. Se trata de un software libre, sencillo,

ligero y eficiente, para la plataforma Windows.

-Code Blocks: Este es un software libre, multiplataforma. Code Blocks es una alternativa a Dev-C++ y desarrollada mediante el propio lenguaje C++. Sus capacidades son bastante buenas y es muy popular entre los nuevos programadores. Se puede encontrar separado del compilado o la versión “mingw” que incluye g++ (gcc para C++).

-Xcode: Este es uno de los mejores IDE para programar en Mac con el compilador gcc e Interface Builder.

Ejecución

Es el paso del uso del programa donde la propia maquina está llevando a cabo el programa ya en un lenguaje que solo la máquina maneja en tiempo real, de este modo, si hemos desarrollado por ejemplo un programa que calcula el cuadrado de cualquier número que se ingrese, el usuario tiene la capacidad de ingresar un número cualquiera y el programa le retornará como resultado el cuadrado de dicho número pero llevando a cabo comandos que la computadora comprende pero un ser humano interpreta de manera más abstracta y menos precisa.

Sin embargo, es en este paso del ciclo de vida del software donde se puede comprobar por interpretación humana externa si el programa se está ejecutando y arrojando los resultados que hipotéticamente debe retornar independientemente de que el programa se haya podido compilar.

**Lenguaje de programación C**

“El lenguaje de programación C o lenguaje de programación de sistemas ofrece una economía

sintáctica, control de flujo y estructuras sencillas y un buen conjunto de operadores. Este fue creado

por Brian Kernaghan y Dennis Ritchie a mediados de los años 70.

El lenguaje C es un lenguaje estructurado, en el mismo sentido que lo son otros lenguajes de

programación tales como el Pascal, el Ada o el Modula-2, pero no es estructurado por bloques, o sea,

no es posible declarar subrutinas (pequeños trozos de programa) dentro de otras subrutinas, a

diferencia de como sucede con otros lenguajes de estructurados como Pascal.

Uno de los objetivos de diseño del lenguaje C es que sólo sean necesarias unas pocas instrucciones en

lenguaje de máquina para traducir cada elemento del lenguaje, sin que haga falta un soporte intenso

en tiempo de ejecución.

Con la popularidad de las microcomputadoras muchas compañías comenzaron a implementar su

propio C por lo cual surgieron discrepancias entre sí.

Por esta razón ANSI (American National Standars Institute, por sus siglas en inglés), estableció un

comité en 1983 para crear una definición no ambigua del lenguaje C e independiente de la máquina

que pudiera utilizarse en todos los tipos de C.

Algunos de las C existentes son:

-Quick C

-C++

-Turbo C

-Turbo C ++

-Borland C

-Borland C++

-Microsoft C

-etc.

Este manual se basa en el C estándar el cual

puede utilizarse en todos los tipos de C.

C es un lenguaje de programación de nivel medio

ya que combina los elementos del lenguaje de alto nivel con la funcionalidad del ensamblador.

Su característica principal es ser portable, es decir, es posible adaptar los programas escritos para un

tipo de computadora en otra.

Otra de sus características principales es el ser estructurado, es decir, el programa se divide en

módulos (funciones) independientes entre sí.

El lenguaje C inicialmente fue creado para la programación de

-Sistemas operativos

1

-Intérpretes

-Editores

-Ensambladores

-Compiladores

-Administradores de bases de datos.

Actualmente, debido a sus características, puede ser utilizado para todo tipo de programas

**Licencia de GPL de GNU**

El software presente en esta guía práctica es libre bajo la licencia GPL de GNU, es decir,

se puede modificar y distribuir mientras se mantenga la licencia GPL.

/\*

\* This program is free software: you can redistribute it and/or modify

\* it under the terms of the GNU General Public License as published by

\* the Free Software Foundation, either version 3 of the License, or

\* (at your option) any later version.

\* This program is distributed in the hope that it will be useful,

\* but WITHOUT ANY WARRANTY; without even the implied warranty of

\* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

\* GNU General Public License for more details.

\* You should have received a copy of the GNU General Public License

\* along with this program. If not, see <http://www.gnu.org/licenses/>.

\* Authors: Julio A. de León, Jorge A. Solano and Hugo Zuñiga

\*/

Comentarios

Los comentarios son texto que el usuario ingresa en determinadas secciones del código para hacer señalizaciones o avisos para que se entienda que hace un bloque o una línea de código determinada, así facilita la compresión del código a otros usuarios o así mismo para recordar o entender que se hizo o que hace cada parte del código

En C existen 2 tipos de comentarios, el comentario de línea y el comentario de bloque

El comentario de línea se inicia con una doble diagonal (//texto), y termina con el salto de línea, es decir hasta donde termina el renglón

El comentario de bloque se inicia cuando se inserta diagonal asterisco (/\* texto) y termina cuando se encuentran los símbolos asterisco diagonal (\*/) y por ello este tipo de comentario puede abarcar varios renglones seguidos

**El conjunto de caracteres en C**

“Del mismo modo que en nuestro lenguaje habitual utilizamos un conjunto de caracteres

para construir instrucciones que tengan significado, los programas que se realicen en C

se escriben utilizando un conjunto de caracteres formado por lo siguiente:

• Las 26 letras minúsculas del alfabeto inglés (a b c d e f g h i j k l m n o p q r s t u v

w x y z).

• Las 26 letras mayúsculas (A B C D E F G H I J K L M N O P Q R S T U V W X Y Z).

• Los 10 dígitos (1 2 3 4 5 6 7 8 9 0).

• Los símbolos especiales (@ ^ { } [ ] ( ) & $ % # ~ ‘ “ / ? ; : \_ . , = / \* - +).

• El espacio en blanco o barra espaciadora.

Nota: Lenguaje C es sensible a minúsculas y mayúsculas.”CITA PRAAACTICA

**Palabras reservadas**

Las palabras reservadas (Tabla 1) son palabras que tienen un significado predefinido

estándar y sólo se pueden utilizar para su propósito ya establecido; no se pueden utilizar

como identificadores definidos por el programador. En lenguaje C son las siguientes:

auto, double, int, struct, break, else, long, switch,case enum register typedef, char, extern, return, union, const, float, short, unsigned, continue, for, signed, void, default, goto, sizeof, volatile, do, if, static, while.

**Tipos de datos**

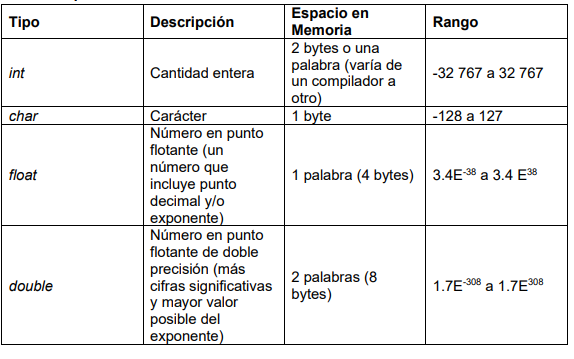
En lenguaje, los datos pueden corresponder a diferentes categorías de organización, y cada tipo de dato se encuentra representado de distinta manera en la memoria

Tabla 2 “Tipos de datos”

**Código declaración de variables (se declaran variables y se asignan datos)**

“El siguiente programa muestra la manera en la que se declaran y asignan variables de

diferentes tipos: numéricas (enteros y reales) y tipo carácter.

No compila por la línea unsigned double puntoFlotanteNumero2 = 238.2236,

lo cual es correcto y es lo que se espera. La razón de que no compile es porque los

valores reales siempre llevan signo y no es correcto usar algún calificador de signo, en

este caso el unsigned.”

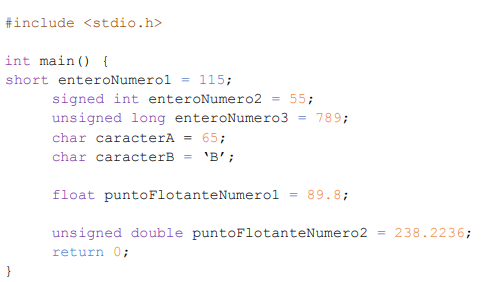
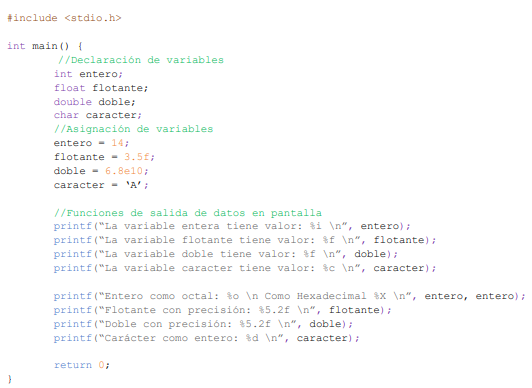


Imagen TAL “Ejemplo de programa asignando variables”

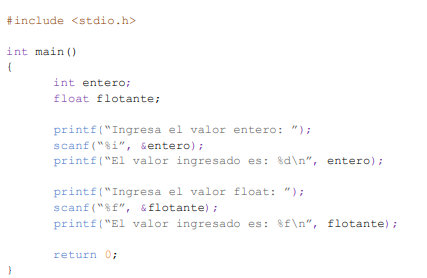
**Código que ejemplifica el uso de la función printf.**

El siguiente programa imprime en pantalla diversos tipos de datos utilizando la función printf.



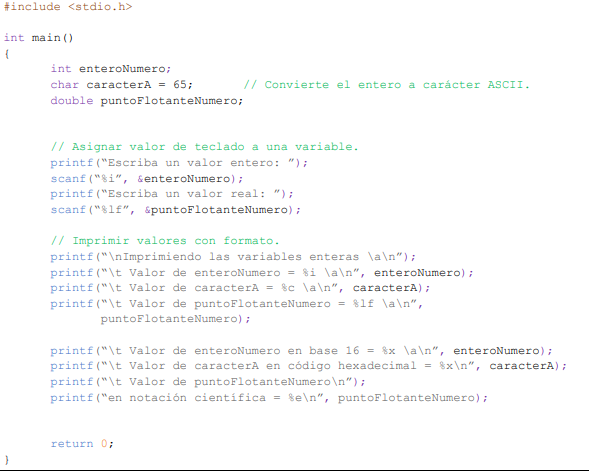
**Código que ejemplifica el uso de la función scanf.**

El siguiente programa muestra como se almacenan los datos de las variables.



**Código que utiliza las funciones printf y scanf.**

Este programa muestra cómo almacenar e imprimir variables.



**Operadores**

Los operadores aritméticos que maneja el lenguaje C se describen en Tabla:

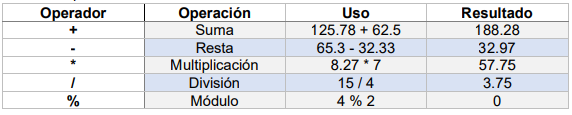
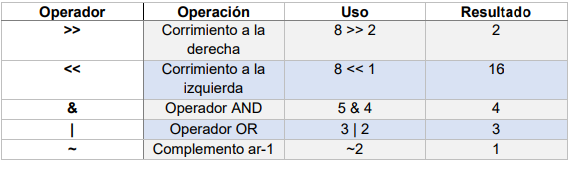


Tabla de operadores con bits



**Expresiones lógicas**

“Las expresiones lógicas están constituidas por números, caracteres, constantes o

variables que están relacionados entre sí por operadores lógicos. Una expresión lógica puede tomar únicamente los valores verdadero o falso.

Los operadores de relación permiten comparar elementos numéricos, alfanuméricos,

constantes o variables (Tabla 8).” (FACULTAD DE INGENIERÍA,2021)

Tabla 8 para operadores relacionales

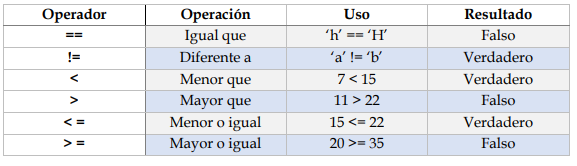
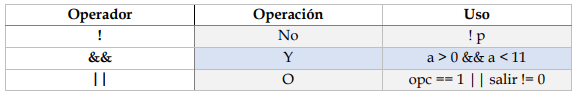


Tabla 9 para operadores lógicos



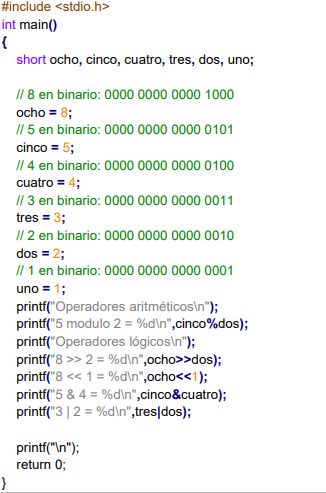
**Código que ejemplifica el uso de operadores a nivel de bits.**

El siguiente programa muestra cómo manipular números a nivel de bits:

- Corrimiento de bits a la izquierda y a la derecha

- Operador AND a nivel de bits

- Operador OR a nivel de bits



Lenguaje C posee operadores para realizar incrementos y decrementos de un número. El operador ++ agrega una unidad (1) a su operando. Es posible manejar preincrementos (++n) o pos-incrementos (n++). El operador -- resta una unidad (1) a su operando. Se pueden manejar pre-decrementos (--n) o pos-decrementos (n-).

**Actividad en clase**

Para llevar a cabo el aprendizaje de los objetivos en clase se trabajó en una terminal tipo Linux



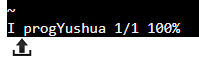
Como primeros pasos se enseño el nuevo comando de creación y edición de programas (vi)

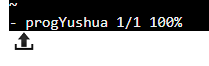


En este formato de la consola se permite la creación de código interpretable por la consola, usando las reglas habituales de la programación como en C++

Para poder salir y editar el programa se hace uso de la letra i y tecla esc (escape)

En modalidad “i” se permite la edición del programa, con esc se deshabilita



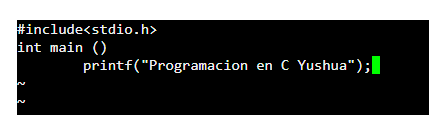


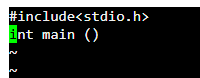
Posteriormente se probó el comando “:wq” que sirve para guardar y salir el programa en cuestión, esto solo se puede hacer con la edición deshabilitada



El comando “vi “nombreprograma” también sirve para reacceder al programa y a la edición de este

El comando “dd” sirve para borrar una línea de código entera, esto se hace fuera del modo de edición.



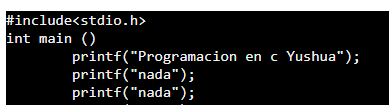


También se puede utilizar el comando “dd” con un numero inicial que será la cantidad de líneas de código a borrar

El comando U sirve como el control z de Windows, funciona para restaurar la última modificación



El comando “gg” reemplaza el control c de Windows, realiza un copiado de una línea de código, para pegar la copia se usa la tecla “p”



Comandos “W”

1-Con el comando **w+nombre** se puede guardar el programa en el momento de la edición sin necesidad de deshabilitar esta última

2-Con el comando **w** se sale del programa sin guardado del mismo

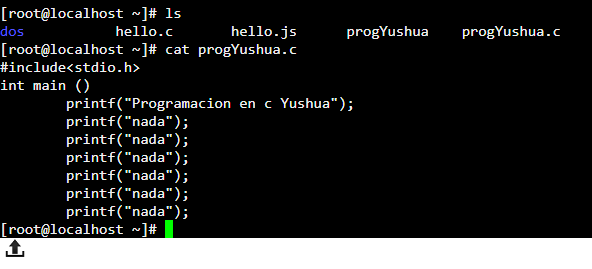


3-Con el comando **wq** se guarda y sale en automático del programa y su edición



Comando “cat”

El comando cat permite la visualización del programa, no se permite editar

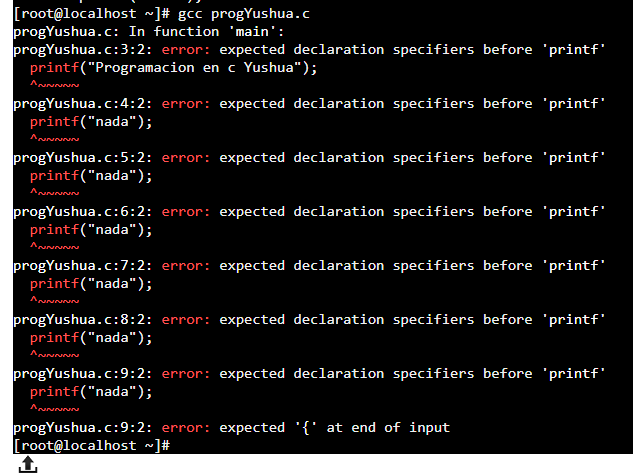


**Comando nano**

Nano es un comando para habilitar la edición del código

**Comando para compilar**

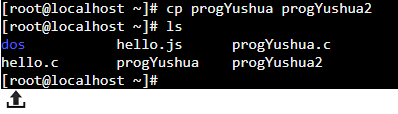
El comando de compilación es el “ggc nombre del programa”



Sin embargo, este programa funciona como prueba de lo que arroja la consola cuando el programa no está bien consolidado

Comando de copiado

El comando de copiado es el “cp nombre del programa”



Tarea:

1.- Investigar cual es el dato que se encuentra por default en en lenguaje C( signed o unsigned)

Signed: variable que llevará signo, es un dato que viene por default

2.- Indicar que sucede cuando en una variable tipo carácter se emplea el formato %d, %i, %o, %x

R: Se va a interpretar-leer como una variable entero, octal, hexadecimal o como dato de doble precisión

3.- Mencionar las características con las que debe crearse una variable

R: 1-No debe interferir con palabras reservadas

2-Debe ser fácil de ubicar e interpretar, nombrarla con un nombre largo solo entorpece la elaboración del código

3-Debe elaborarse con letras del alfabeto romano

4.- ¿Cuál es la diferencia entre variable estática y constante?

R: Las variables constantes son incambiables, las variables estáticas pueden cambiar sus valores como vaya requiriendo el programa y el usuario

5.- Menciona en cual momento empleas los dos tipos de diferentes ( < >   != )   Referente a pseudocódigo y codificación

<> ¡= son referentes a pseudocódigo y codificación

<> se usan para declarar una librería como la de acentos o la stdio,h

I=o se usa para declarar el valor de una variable

Los <,> también sirven como declaradores aritméticos que simbolizan que algo es mayor o menor que otra cosa

6.- Crea un programa en el que declares 4 variables haciendo uso de las reglas signed/unsigned, las cuatro variables deben ser solicitadas al usuario (se emplea scanf) y deben mostrarse en pantalla (emplear printf)

**Problema:** creación de un programa de 4 variables

**Datos de entrada:** 4 variables cuales sean

**Restricción:** Dadas las características solicitadas, serán únicamente 4 variables dadas por el usuario

**Datos de salida:** aparentemente nada, solo la declaración de 4 variables

**Pseudocódigo**

#include<stdio.h>

int main ()

Escribir ("Bienvenido, favor de ingresar 4 variables");

Definir b,c,d,e como entero

Escribir ("Ingresa los valores de las 4 variables");

leer b

leer c

leer d

leer e

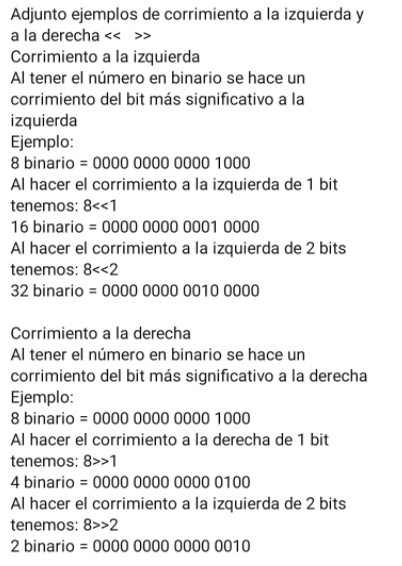
Escribir ("Sus variables son" b,c,d,e);

return 0;

7.- Crea un programa que le solicite su edad al usuario, leer el datos(emplear scanf) y mostrarlo en pantalla

8.- Revisar y colocar cuando se emplea MOD y cuando se emplea % (pseudocódigo y códificación) agrega un ejemplo de su uso.

9.- Da tu opinión sobre cada uno de los ejemplos realizados en la imagen.

  
Profesionalmente no comprendo la finalidad de este programa más que sea la codificación de cantidades a base binaria, a lo cual desconozco la finalidad del corrimiento izquierda o derecha, puesto que la máquina puede que sea más eficiente en un corrimiento de un sentido al otro, sin embargo, considero que ese texto corresponde a un intento de análisis y codificación de instrucciones legibles a humano.

10.- Comparación entre Editor de Texto y Procesador de Texto (Realizar una tabla comparativa)

|  |  |
| --- | --- |
| Procesador | Editor |
| Se pueden elaborar documentos de gran extensión | Esta pensado para la elaboración de programas |
| El contenido no sigue una regla de elaboración | Para elaborar código se deben acatar las reglas de la elaboración de código |
| Hay posibilidad de editar la propia forma del texto, tamaño y fuente | Solo puede manejar texto universal |
| La interfaz como de Word está pensada para ser intuitiva | La interfaz de los editores no es intuitiva ni comoda al ojo humano |
| Se manejan con idiomas humanos | Se pretende usar para que la computadora lo interprete |
| El más famoso procesador de texto es Word | Los editores más reconocidos son C++, Notepad++,brakets |
| Permite usar tablas, gráficas, imagenes | No permite utilizar más que el texto universal del alfabeto romano |

11.- Indica los comandos ejecutados para compilar y para ejecutar un programa en iOS o Linux

Compilar: gcc nombre del programa

Ejecutar: gcc nombre del programa -o nombre del programa.out (se debe tener la versión del programa en cuestión con extensión .out)

12.- Compilación y prueba del programa antes mostrado en DEV C++ u otro IDE en sistema operativo Windows.

13.- Genera un programa que solicite dos variables enteras al usuario y realice las 4 operaciones básicas, compila y ejecuta el programa utilizando terminal y los comandos indicados para cada instrucción.

Referencias

-