

UNIVERSIDAD DE LIMA

Facultad de Ingeniería y Arquitectura

Carrera de Ingeniería de Sistemas

Sistemas Operativos

Laboratorio 03A

Introducción al Lenguaje C (I)



Agosto 2023



Objetivos

El desarrollo de la presente guía de laboratorio persigue los siguientes objetivos:

1. Familiarizarse con la programación en lenguaje C sobre Linux.
2. Programar estructuras de control en C y operadores lógicos.
3. Analizar la programación, desarrollo, funcionamiento y resultados de los algoritmos planteados.

01

Introducción

El lenguaje de programación “C” fue desarrollado por Dennis Ritchie entre 1969 y 1972 en los Laboratorios Bell para ser usado en UNIX. Es un lenguaje de programación de tipo estructurado, similar a Pascal, Fortran, Basic. “C” es bastante eficiente en los resultados y es el lenguaje de programación más utilizado en la creación de software de sistemas, siendo también utilizado para crear aplicaciones.

C utiliza instrucciones y sentencias como if, else, for, do y while. A pesar de ser un lenguaje de alto nivel (puesto que es estructurado y posee sentencias y funciones que simplifican su funcionamiento), se puede programar a bajo nivel (similar a lenguaje ensamblador).

02

Desarrollo de un programa en C desde línea de comandos en Linux

Para este laboratorio se utilizará un computador virtualizado con sistema operativo GNU/Linux Ubuntu 18.04 LTS. La programación se hará con lenguaje C usando línea de comandos (terminal).

INSTALACIÓN DEL COMPILADOR GCC

Para poder compilar y ejecutar un programa en C, es necesario tener los paquetes esenciales en el sistema operativo.

Paso N°01: Iniciar sesión como usuario **usistemas**

Paso N°02: Cargar una ventana de terminal (CTRL + ALT + T) y a continuación digitar los siguientes comandos:

```
sudo apt-get install build-essential
```

En caso tenga problemas con la instalación, seguir el siguiente procedimiento:

```
sudo apt install -y software-properties-common
sudo add-apt-repository -y ppa:ubuntu-toolchain-r/test
sudo apt install -y gcc-9 g++-9
sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-9
90 --slave /usr/bin/g++ g++ /usr/bin/g++-9 --slave /usr/bin/gcov
gcov /usr/bin/gcov-9
```

Esperar la descarga e instalación de paquetes

CODIFICAR PROGRAMA

Paso N°01: Generar el archivo *ProgramaBasico.c* en el editor de texto **gedit**

```
Sintaxis: gedit [nombre_archivo].c
Comando:  gedit ProgramaBasico.c
```

Paso N°02: Escribir el programa ejemplo que sigue a continuación y luego guarde los cambios:

```
#include<stdio.h>
int main()
{
    printf("Programación en C\n");
    return 0;
}
```

COMPILAR PROGRAMA

Paso N°01: Compile el programa en C con GCC (GNU Compiler Collection)

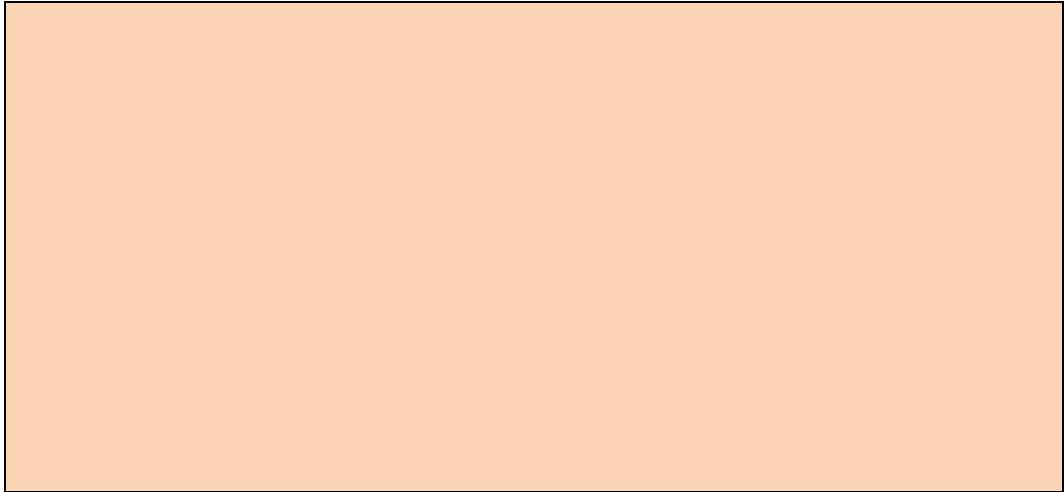
```
Sintaxis: gcc [nombre_programa].c -o nombre_programa
Comando:  gcc ProgramaBasico.c -o ProgramaBasicoCompilado
```

EJECUTAR PROGRAMA

Paso N°01: Compile el programa en C con GCC (GNU Compiler Collection)

```
Sintaxis: ./nombre_programa
Comando:  ./ProgramaBasicoCompilado
```

Paso N°02: Explique la funcionalidad del programa desarrollado y cada parte de la estructura



03

Manipulación de bits: Operadores lógicos, bitwise, bit masking

Ejercicio N°01:

Para las variables $a = 23$ y $b = 90$, haga su representación en binario y luego realice la operación AND.

Escriba el programa en C, compile, ejecute y compare los resultados.

```
#include <stdio.h>
int main ( )
{
    char a = 23;
    char b = 90;
    char resultado;

    printf ("a=%d b=%d",a,b);
    resultado = a & b;
    printf ( "Resultado = %d",resultado) ;
    return 0 ;
}
```

Ejercicio N°02:

Se tiene el número $a = 75$ y se quiere averiguar si su cuarto bit es 1 o 0. Haga su representación en binario y luego realice la operación AND con 8 en binario.

Escriba el programa en C, compile, ejecute y compare los resultados.

```
#include <stdio.h>

int main ( )
{
    char a = 75;
    char b = 8;
    char resultado;

    printf ("a=%d b=%d",a,b);

    if (a & b)
        printf ("El cuarto bit de la variable a es 1\n") ;
    else
        printf ("El cuarto bit de la variable a es 0\n") ;
    return 0 ;
}
```

Ejercicio N°03:

Para las variables $a = 23$ y $b = 90$, haga su representación en binario y luego realice la operación OR.

Escriba el programa en C, compile, ejecute y compare los resultados.

```
#include <stdio.h>
int main()
{
    char a = 23;
    char b = 90;
    char resultado;

    printf ("a=%d b=%d", a, b);
    resultado = a | b;
    printf ( "Resultado = %d", resultado) ;
    return 0 ;
}
```

EJERCICIOS PROPUESTOS

Ejercicio N°01:

Para las variables a = 23 y b = 90, haga su representación en binario y realice la operación XOR, luego escriba el programa en C, compile, ejecute y compare los resultados

Ejercicio N°02:

Para las variables a = 23 y b = 90, haga su representación en binario y realice la operación complemento a 1 a la Base, luego escriba el programa en C, compile, ejecute y compare los resultados.

PROBLEMAS PROPUESTOS

Problema N°01:

Desarrollar el algoritmo y su posterior código en C que permita calcular el volumen de un cubo

Problema N°02:

Supongamos que la bolsa de un kilo de azúcar y un paquete de 500g de café molido están a 4.30 y 26.90 soles respectivamente, desarrollar en primer término un algoritmo y luego un programa en lenguaje C que permita efectuar lo siguiente:

1. Solicitar por teclado una cantidad soles, pudiendo expresarlo en céntimos inclusive (por ejemplo, 100.80)
2. Si la tercera parte de la cantidad introducida se destina a comprar azúcar, la mitad a comprar café, y el resto no se gasta. Calcule cuántos kilos de azúcar y café (datos reales) se pueden comprar con dicha cantidad de dinero, así como la cantidad (dato real) de dinero sobrante
3. Muestre por pantalla los resultados, expresado en datos reales

Problema N°03:

Escribir en lenguaje C un programa que cumpla con las siguientes indicaciones:

1. Solicitar por teclado dos números (datos enteros) y almacenarlos en dos variables, llamadas **v1** y **v2**
2. Intercambie los valores de las variables
3. Mostrar en pantalla los valores de las variables antes y después del intercambio

Problema N°04:

Desarrollar en primer término un algoritmo y luego un programa en lenguaje C un programa que cumpla con las siguientes indicaciones:

1. Solicitar por teclado tres números (datos enteros) y almacenarlos en tres variables, llamadas **v1**, **v2** y **v3**
2. Intercambie los valores de las variables del siguiente modo:
 - a. El contenido de **v1** pasa a **v2**
 - b. El contenido de **v3** pasa a **v1**
 - c. El contenido de **v2** pasa a **v3**
3. Mostrar en pantalla los valores de las variables antes y después del intercambio