

Fundamentos de programación

Ing. Stalin Francis

December 6, 2021

Outline

Creación de grupos whatsapp

UNIDAD 0: Presentación.

UNIDAD 1: Introducción a las computadoras y los lenguajes de programación.

UNIDAD 2: Nociones de linux, vim, clang

UNIDAD 3: Metodología de la programación y Diagrama de flujo

Semana de Evaluación sumativa

UNIDAD 4: Programación en C++: Introducción.

UNIDAD 5: Flujo de control :Selección y Repetición.

UNIDAD 6: Flujo de control II: Estructura Repetitiva

UNIDAD 7: Funciones y librerías personales

UNIDAD 8: Programación Orientada a Objetos: clases



Cración de grupos watsapp



Grupos de Whatsapp

- ▶ Fundamentos-2020-2S-PB
 - ▶ Creador y administrador: Loor Perea Patrick
- ▶ Fundamentos-2020-2S-PA
 - ▶ Crador y administrador: Jorge Ortiz

Grupos de trabajos

- 5 personas





UNIDAD 0: Presentación.



<2021-09-13 Mon 09:00>--<2021-09-13 Mon 11:00> Session No 1

Presentación del Docente

1. Nombre: Stalin Francis Quinde.
2. Titulos:
 - ▶ **Pregrado: Ingeniero en computación (ESPOL).**
 - ▶ **Postgrado1: Magister en Currículo(UTLVTE)**
 - ▶ **Postgrado2: Magister en Ciencias de la Computación(ESPOL).**
3. Contacto:
 - ▶ **Correo: stalin.francis@utelvt.edu.ec**
 - ▶ **Telefono: 0997919650.**

Presentación de cada uno de los estudiantes.

- ▶ Apellidos y Nombres, ¿Cuántos años tienes?, Título de bachiller que obtuvo., ¿Qué le motivo a seguir la carrera de Tecnología en Tecnología de la Información?, ¿Qué aspira ser al terminar esta carrera?



<2021-09-14 Tue 11:00>--<2021-09-14 Tue 13:00> Session No 2

<2021-09-14 Tue 11:00>--<2021-09-14 Tue 10:00> Video motivacionales

- ▶ Se dedicará una clase a ver los videos y luego cada estudiante realizará un comentario del mensaje que dejo el video.
- ▶ <https://www.youtube.com/watch?v=xKka6kzTQgw&t=822s>

<2021-09-14 Tue 11:00>--<2021-09-14 Tue 13:00> Prueba de diagnostico:

- ▶ Para esta prueba se dedicará un sesión de clase para que los estuدياتes respondan en las dos horas de clases.



UNIDAD 1: Introducción a las computadoras y los lenguajes de programación.



<2021-09-15 Wed 09:00>--<2021-09-15 Wed 10:00> Introducción a la computadora

- ▶ ¿Que es el computador?
 - ▶ **¿De que esta compuesto el computador?**
 - ▶ **¿Para qué sirve el computador?**
- ▶ ¿Porqué se ha vuelto tan importante el computador?
- ▶ ¿Quien creo el computador?
- ▶ ¿Cómo sera el computador en el futuro?
- ▶ ¿Quién es el responsable de al arquitectura del computador?
- ▶ ¿Cuál es la arquitectura del computador?
- ▶ ¿Qué hace cada uno de las partes del computador?



<2021-09-15 Wed 09:00>--<2021-09-15 Wed 10:00> Historia del computador

Año	CIENTÍFICO	CONTRIBUCIÓN
1833	Charles Babbage	Diseña e intento construir la primera computadora (Máquina analítica)
1890	Herman Hollerith	Inventa la máquina tabuladora utilizada para el censo de Estados Unidos. Funda la IBM.
1936	Alan Turing	Formaliza los conceptos de algoritmo y de máquina de Turing, la clave de la computación moderna.
1944	En Inglaterra	Se construye la computadora Colossus (Colossus Mark I y Colossus Mark 2), para descifrar comunicaciones de los alemanes en la Segunda Guerra Mundial.
1947	En la Universidad de Pensilvania	Se construye ENIAC (Electronic Numerical Integrator And Calculator), funciona con válvulas y fue la primera computadora electrónica de propósito general.
1951	Comienza a operar EDVAC, concebida por John Von Neumann	ENIAC no era decimal, sino binaria, y tuvo el primer programa diseñado para ser almacenado.
1951	J. Presper Eckert y John William Mauchly	Crearon UNIVAC I que fue la primera computadora electrónica comercial en los Estados Unidos.

Figure: Historia del computador



<2021-09-15 Wed 09:00>--<2021-09-15 Wed 10:00> Historia del computador

1947	John Bardeen, Walter Brattain, William Shockley de los laboratorio Bell	Inventan el transistor que reemplaza al tubo del vacío
1957	Robert Noyce y Jack Kilby	Inventaron el circuito integrado o microchip, motor de la revolución de los ordenadores personales.
1971	Intel	Presenta el primer procesador comercial, el primer chip: el microprocesador intel 4004 .
1973	Xerox	Desarrollo el Xerox PARC que fue uno de los primeros ordenadores personales de la historia.
1975	Bill Gates y Paul Allen	Fundadores de la empresa Microsoft
1976	Steve Jobs, Steve Wozniak, Mike Markkula	Fundadores de la empresa Apple
1977	Apple	Presenta el primer computador personal que se vende a gran escala Apple II .
1981	IBM	Lanza al mercado la IBM PC que se convierte en un éxito comercial.
1983	Microsoft	presenta el sistema operativo MS-DOS, por encargo de IBM.

Figure: Historia del computador

<2021-09-15 Wed 09:00>--<2021-09-15 Wed 10:00> Arquitectura del computador

Diagrama de la arquitectura Von Neuman

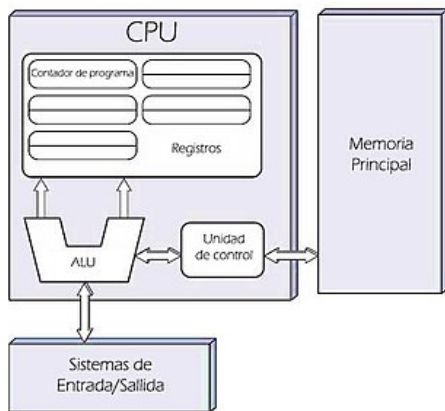


Figure: Arquitectura Von Neumann



El software.

- ▶ ¿Que es el software?
- ▶ ¿Qué es numeración binaria?
- ▶ ¿Que es sistemas de numeración?
- ▶ ¿Como transformas de un sistemas de numeración a otro?
- ▶ Ejercicio para el fin de semana.

Sistema de Numeración Binaria

Address	Value
0x00	01001010
0x01	10111010
0x02	01011111
0x03	00100100
0x04	01000100
0x05	10100000
0x06	01110100
0x07	01101111
0x08	10111011
...	...
0xFE	11011110
0xFF	10111011

Figure: La memoria del computador



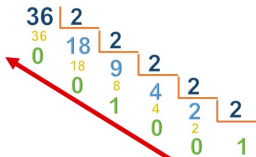
Sistemas de Numeración

- ▶ Sistema de numeración Binaria.
- ▶ Sistema de numeración Octal.
- ▶ Sistema de numeración Decimal.
- ▶ Sistema de numeración Exadecimal.

Conversión Sistemas de Numeración.

DECIMAL → BINARIO
(Parte fraccionaria)

$$(36.75)_{10} = (100100.11)_2$$



36		2	
36		2	
18		2	
18		2	
9		2	
9		2	
4		2	
4		2	
2		2	
2		2	
0		2	
0		2	
0		2	
1		2	

$$0.75 \times 2 = 1.5$$

$$0.5 \times 2 = 1.0$$

,11

Figure: Conversión Sistemas de Numeración.

Lenguaje de programación



Figure: Todos los lenguajes de programación

Lenguaje a utilizar para esta curso.



The image shows the C++ logo in large blue letters. Below it is a snippet of C++ code in a monospaced font, with some words highlighted in pink, purple, and blue. The code is as follows:

```
#include <iostream>
using namespace std;

int main(){
    cout<<"Hola Facebook\n";
    return 0;
}
```

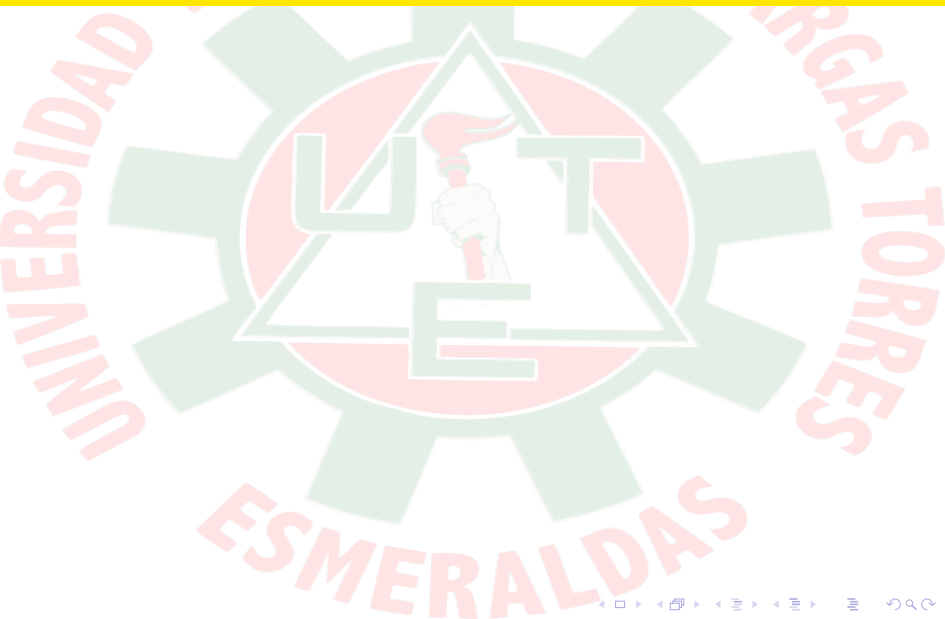
Figure: Lenguaje a utilizar en este



UNIDAD 2: Nociones de linux, vim, clang



Introducción a Linux y termux



Paquetes de Linux: ejercicios prácticos





Introducción a Vim y sus comandos.



Ejercicios prácticos con Vim.





UNIDAD 3: Metodología de la programación y Diagrama de flujo



Introducción a al programación



Ciclo de Vida del Software.



Diagrama de Flujo: Hola Mundo.



Semana de Evaluación sumativa



UNIDAD 4: Programación en C++: Introducción.

Elementos básicos en un programa en c++.

Básicos.

- ▶ Palabras reservadas (main, return, if while, do ,.. etc.).
- ▶ Identificadores (nombre de variables, nombre de funciones, nombres de programas, etc.).
- ▶ Caracteres especiales (coma,punto,punto y coma, etc.).
- ▶ Constantes.
- ▶ Variables.
- ▶ Expresiones.
- ▶ Instrucciones.

Derivados.

- ▶ Bucles.
- ▶ Contadores.
- ▶ Acumuladores.

Un programa básico en C++.





UNIDAD 5: Flujo de control :Selección y Repetición.

Estructuras de control son:

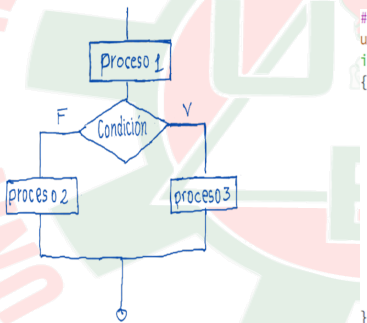
Estructura de selección

- ▶ Estructura if-else
- ▶ Estructura if
- ▶ Estructura switch

Estructura de repetición

- ▶ Estructura do-while
- ▶ Estructura while
- ▶ Estructura for

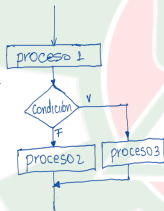
Estructura de selección (if-else).



```
#include<iostream>
using namespace std;
int main()
{
    int a;
    cout<<"Ingrese un número : "; cin>>a;

    if(a%2==0){
        cout<<"En número ingresado es PAR";
    }else{
        cout<<"En número ingresado es IMPAR";
    }
    return 0;
}
```

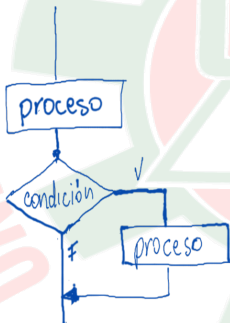
Estructura de control (if-else)



```
#include<iostream>
using namespace std;
int main()
{
    int a;
    cout<<"Ingrese un número : "; cin>>a;

    if(a%2==0){
        cout<<"En número ingresado es PAR";
    }else{
        cout<<"En número ingresado es IMPAR";
    }
    return 0;
}
```

Estructura de control (if)



```
#include<iostream>
using namespace std;
int main()
{
    int a;
    cout<<"Ingresa un número : "; cin>>a;
    if(a%2==0){
        cout<<"Ingreso un número PAR"<<endl;
    }
    return 0;
}
```



Estructura de control (if excluyentes)



```
#include<iostream>
using namespace std;
int main()
{
    int a;
    cout<<"Ingrese un número : "; cin>>a;

    if(a==1)
    {
        cout<<"One"<<endl;
    }

    if(a==2)
    {
        cout<<"Two"<<endl;
    }

    if(a==3)
    {
        cout<<"Thre"<<endl;
    }

    if(a==4)
    {
        cout<<"For"<<endl;
    }

    return 0;
}
```



Estructura de control (switch)



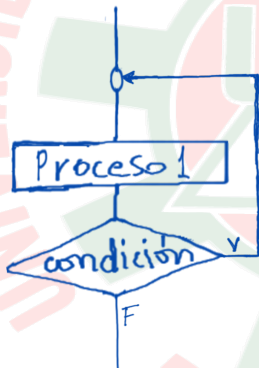
```
#include<iostream>
using namespace std;
int main()
{
    int a;
    cout<<"Ingrese un número : ";
    cin>>a;
    switch(a)
    {
        case 1: cout<<"one"<<endl;
                break;

        case 2: cout<<"one"<<endl;
                break;

        case 3: cout<<"one"<<endl;
                break;
        default: cout<<"Esta número no esta contemplado"<<endl;
    }
    return 0;
}
```

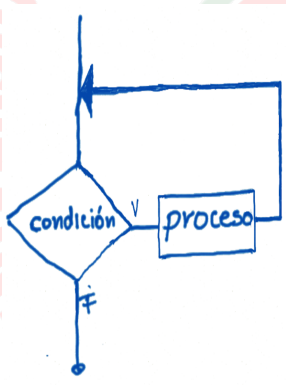


Estructura de control (do-while)



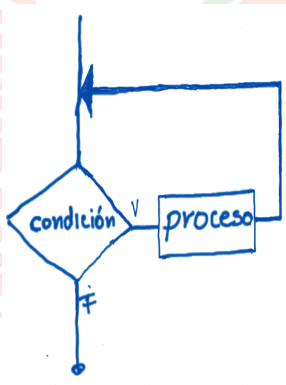
```
#include<iostream>
using namespace std;
int main()
{
    int i=0,a[10]={3,4,5,6,3,2,3,4,5,5};
    do{
        cout<<a[i]<<"-";
        i=i+1;
    }while(i<10);
    cout<<endl;
    return(0);
}
```


Estructura de control (while)



```
#include<iostream>
using namespace std;
int main()
{
    int i=0, a[10]={2,3,4,5,5,6,4,3,2,9};
    while(i<10){
        cout<<a[i]<<"-";
        i++;
    }
    cout<<endl;
    return 0;
}
```

Estructura de control (for)



```
#include<iostream>
using namespace std;
int main()
{
    int a[10]={2,3,5,6,6,7,4,2,3,8};
    for(int i=0;i<10;i++)
    {
        cout<<a[i]<<"-";
    }
    cout<<endl;
    return 0;
}
```



UNIDAD 6: Flujo de control II: Estructura Repetitiva



org-mode + beamer = love

<example block>

Simple block
it's that easy!



UNIDAD 7: Funciones y librerías personales



monolítico

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5
6
7
8
9
10
11
12
13     return 0;
14
15 }
```

monolítico-1.cpp



monolítico2

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5     float s1,s2,r;
6     cout<<"Ingrese el primer sumando s1: "; cin>>s1;
7     cout<<"Ingrese el segundo sumando s2: "; cin>>s2;
8
9     r=s1+s2;
10
11     cout<<"El resultado de la suma fue : "<<s1<<"+"<<s2<<"="<<r<<endl;
12
13     return 0;
14
15 }
```

monolítico-2.cpp



Programación monolítica con funciones

```
1 #include<iostream>
2 using namespace std;
3
4 float suma(float S1,float S2)
5 {
6     float res;
7     res=S1+S2;
8     return(res);
9 }
10
11
12 int main()
13 {
14     float s1,s2,r;
15     cout<<"Ingrese el primer sumando s1: "; cin>>s1;
16     cout<<"Ingrese el primer sumando s2: "; cin>>s2;
17     r=suma(s1,s2);
18
19     cout<<"El resultado de la suma es : "<<s1<<"+"<<s2<<"="<<r<<endl;
20
21     return 0;
22
23
24
25 }
```




Programación monolítica con prototipo de funciones

```
1 #include<iostream>
2 using namespace std;
3
4 //Prototipo de la función
5 float suma(float,float);
6
7 //Función principal
8 int main()
9 {
10     float s1,s2,r;
11     cout<<"Ingrese el primer sumando s1: "; cin>>s1;
12     cout<<"Ingrese el primer sumando s2: "; cin>>s2;
13     r=suma(s1,s2);
14
15     cout<<"El resultado de la suma es : "<<s1<<"+"<<s2<<"="<<r<<endl;
16
17     return 0;
18 }
19
20
21
22 //Función suma
23 float suma(float S1,float S2)
24 {
25     float res;
26     res=S1+S2;
27     return(res);
28 }
```

Programa monolítico.

```
1 #include<iostream>
2 using namespace std;
3 #include "milibreria.h"
4
5
6 int main()
7 {
8     float s1,s2,r;
9     cout<<"Ingrese el primer sumando s1 : "; cin>>s1;
10    cout<<"Ingrese el segundo sumando s2 : "; cin>>s2;
11
12    r=suma(s1,s2);
13
14    cout<<"El resultado de la suma es : "<<s1<<"+"<<s2<<"="<<r<<endl;
15
16    return 0;
17
18 }
```

principal.cpp

```
1 float suma(float S1,float S2)
2 {
3     float res;
4     res=S1+S2;
5     return(res);
6
7 }
```

milibreria.h



UNIDAD 8: Programación Orientada a Objetos: clases
