

Laboratorio I

Toma de Decisiones



FECyT
Facultad de Ciencias
Exactas y Tecnológicas



UNSE
Universidad Nacional
de Santiago del Estero



Sentencia *if*

- La sintaxis de la sentencia **if** es la siguiente:

```
If (expresión)  
sentencia ;  
  
siguiente_sentencia;
```

- En donde la **expresión** entre paréntesis puede ser cualquier expresión que resulte en un valor **verdadero (true)** o **falso (false)**; y la **sentencia** una sentencia simple o bien un **bloque de sentencias**

Laboratorio I – Versión Preliminar - Aldo Roldán



Sentencia *if*

- **Ejemplo:**

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

int main(int argc, char *argv[])
{
    int edad;
    printf("Ingrese su edad:");
    scanf("%d",&edad);
    if(edad>=18)
        printf("Usted es mayor de edad.\n");
    return (EXIT_SUCCESS);
}
```

Laboratorio I – Versión Preliminar - Aldo Roldán



Expresiones Lógicas

- Las **expresiones lógicas o booleanas** son aquellas que pueden resultar en solo uno de dos valores: **verdadero** o **falso**.
- Resulta importante destacar que cualquier valor numérico distinto de cero resultara en **verdadero**, y el valor cero será considerado como **falso**.
- En las mismas pueden hacerse presente tanto operadores relacionales, como lógicos

Laboratorio I – Versión Preliminar - Aldo Roldán



Operadores Relacionales

- Los **operadores relacionales** permiten comparar valores para la toma de decisiones.

Operador	Significado	Ejemplo
<	Es menor que	$A < 35$
==	Es igual que	$A == B$
>	Es mayor que	$A > B$
<=	Es menor o igual que	$A <= 35$
!=	Es distinto que	$A != B$
>=	Es mayor o igual que	$A >= B$

Laboratorio I – Versión Preliminar - Aldo Roldán



Operadores Lógicos

- Los **operadores lógicos** permiten combinar dos o más expresiones para la toma de decisiones.

Operador	Significado	Ejemplo
&&	AND (y)	$(A < 35) \&\& (B != 1)$
	OR (o)	$(A == B) (A >= 15)$
!	NOT (no)	$!(A == B)$

Laboratorio I – Versión Preliminar - Aldo Roldán



Bloque de Sentencias

- Un **bloque de sentencias** es un conjunto de sentencias encerradas entre llaves (`{ }`), las cuales a los efectos de la sintaxis se consideran como una única sentencia.
- Esto significa que cuando en el formato de cualquier sentencia se observe **sentencia**, esta podrá ser una sentencia simple o bien un **bloque de sentencias**.
- En otros lenguajes se las denominan **sentencias** compuestas.

Laboratorio I – Versión Preliminar - Aldo Roldán



Bloque de Sentencias

- La sintaxis de la sentencia **if** cuando se usa un **bloque de sentencias** es la siguiente:

```
If (expresión)
{
    sentencia 1;
    sentencia 2;
    .....
    sentencia n;
}

Siguiente Sentencia;
```

Laboratorio I – Versión Preliminar - Aldo Roldán



Bloque de Sentencias

- **Ejemplo:**

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

int main(int argc, char *argv[])
{
    int edad;
    printf("Ingrese su edad:");
    scanf("%d",&edad);
    if(edad>=18)
    {
        printf("Usted es mayor de edad.\n");
        printf("Ya que tiene %d años\n", edad);
    }
    return (EXIT_SUCCESS);
}
```



Sentencia *if.. else*

- La sintaxis de la sentencia *if - else* es la siguiente:

```
If (expresión)
    sentencia1;
else
    sentencia2;

siguiente_sentencia;
```



Sentencia *if.. else*

- **Ejemplo:**

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{
    int edad;
    printf("Ingrese su edad:");
    scanf("%d", &edad);
    if(edad >= 18)
    {
        printf("Usted es mayor de edad.\n");
        printf("Ya que tiene %d años\n", edad);
    }
    else
        printf("Usted es menor de edad.\n");
    return (EXIT_SUCCESS);
}
```

Laboratorio I – Versión Preliminar - Aldo Roldán



Sentencia *if.. else if* para múltiples decisiones

- La sintaxis de la sentencia *if - else if* es la siguiente:

```
If (expresión1)
    sentencia1;
else if (expresión2)
    sentencia2;
else if (expresión3)
    sentencia3;
.....
else
    sentencia_n

siguiente_sentencia;
```

Laboratorio I – Versión Preliminar - Aldo Roldán



Sentencia *if.. Else* para multiples decisiones

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

int main(int argc, char *argv[])
{
    int edad;
    printf("Ingrese su edad:");
    scanf("%d",&edad);
    if(edad>=25)
        printf("Usted tiene mas de 25 años.\n");
    else if ((edad>=18)&&(edad<=21))
        printf("Usted tiene entre 18 y 21 años.\n");
    else if (edad==17)
        printf("Usted tiene 17 años.\n");
    else
        printf("Usted no esta en ningun rango.\n");
    }
    return (EXIT_SUCCESS);
}
```

Laboratorio I – Versión Preliminar - Aldo Roldán



Sentencia *switch*

- La sintaxis de la sentencia **switch** es la siguiente:

```
switch (expresión entera)
{
    case expresión constante 1:
        sentencia1;
        break;
    .....
    case expresión constante N:
        sentenciaN;
        break;
    default
        sentencia;

    siguiente_sentencia;
```

Laboratorio I – Versión Preliminar - Aldo Roldán



Sentencia *switch*

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
int main(int argc, char *argv[])
{
    char letra;
    printf("Ingrese una vocal:"); scanf("%c",&letra);
    letra = tolower(letra);
    switch (letra)
    {
        case 'a':
            printf("Ingreso una letra a\n");
            break;
        case 'e':
            printf("Ingreso una letra e\n");
            break;
        .....
        default:
            printf("Ingreso un caracter que no es una vocal\n");
    };
    return 0;
}
```



Operador condicional

- El **operador condicional** es un operador ternario el cual presenta el siguiente formato:
condición ? expresión 1: expresión2;
en donde si :
 - La **condición** es verdadera, el resultado de la operación será **expresión 1**.
 - La **condición** es falsa, el resultado de la operación será **expresión 2**.

Ejemplo:

incremento = antigüedad > 10 ? 300 : 150;

Laboratorio I – Versión Preliminar - Aldo Roldán