

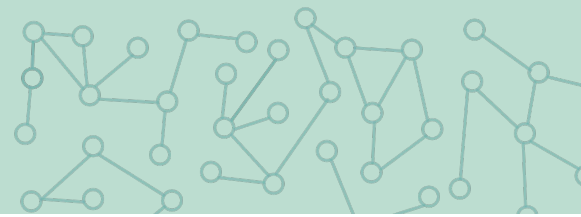


Clase N°3

Control de flujo

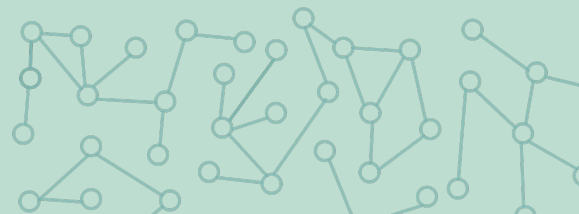
Taller de programación
Computines





- errores
- debug
- resolver en conjunto un problema
- materia control de flujo
- ejercicio corto control de flujo
- ejercicios hackerrank

¿Qué vimos la clase pasada?



Tipos de variable:

- `int`
- `double`
- `char`
- `String`

```
int edad = 24;  
edad = edad + 1;
```

```
double mitad = (double)(edad) / 2;
```

```
char inicial = 'z';
```

```
String nombre = "Fernando";
```

// ¿Qué es una variable?

¿Qué vimos la clase pasada?

Operaciones y métodos:

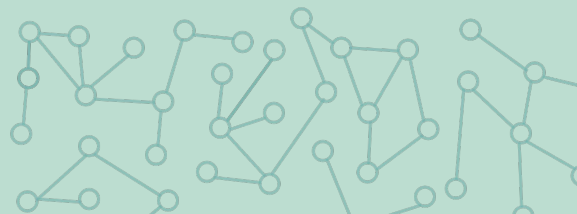
- ¿Qué operaciones entre variables podemos realizar?
- Métodos básicos

+ suma
- resta
* multiplicación
/ división
% módulo

```
System.out.println("El resultado es: " + result);
```

```
Scanner scanner = new Scanner(System.in);  
String texto = scanner.nextLine();  
int numero = scanner.nextInt();
```

Errores típicos



Traducción error: tipos incompatibles: posible pérdida al convertir desde double a int

! error: incompatible types: possible lossy conversion from double to int

```
int r = ((double) a) / ((double) b);
```

Traducción error: se esperaba un ';'

! error: ';' expected

```
result = ((double) a) / ((double) b)
```

92

0800 Antenn started
1000 " stopped - antenn ✓

13⁰⁰ MC (032) MP-MC { 1.2700 9.037 847 025
~~2.130476415~~ 9.037 846 995 correct
(033) PRO-2 2.130476415
correct 2.130676415

Relays 6-2 in 033 failed special speed test
in Relay "11,000 test."

1100 Started Cosine Tap (Sine check)
1525 Started Multy Adder Test.

1545

Relay #70 Panel F
(moth) in relay.

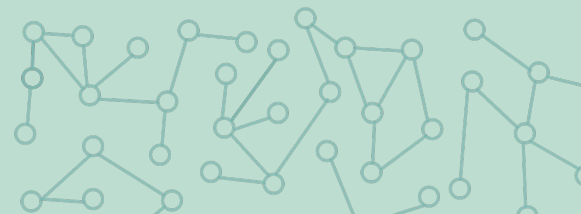
First actual case of bug being found.
1700 antenae started.
1700 closed down.

Relay 2145
Relay 3371

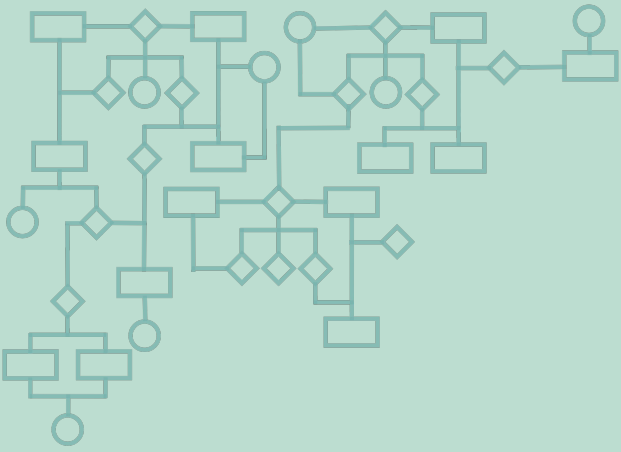
El primer “*bug*”



¿Debugging?

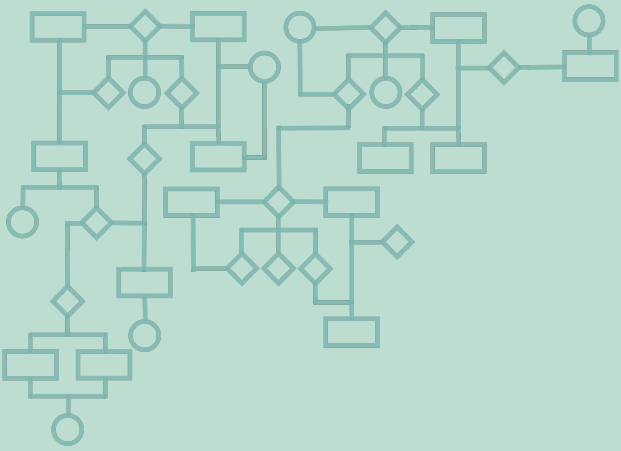


- El proceso de identificar y corregir errores de programación.
- La técnica (o arte) de encontrar dónde está el error en mi programa.



Mentimeter

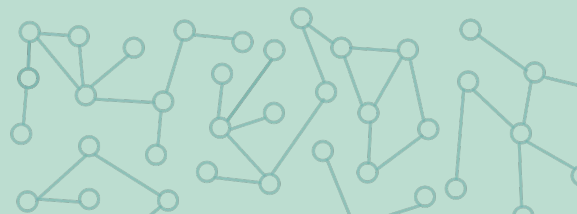




Control de flujo



Booleans



```
boolean miBoolean;  
miBoolean = true;  
miBoolean = false;
```

Comparaciones entre números

Comparaciones (==, !=, <, >, <=, >=)

```
int numero = 5;  
boolean menorQueSiete = numero < 7;  
boolean igualQueNueve = numero == 9;  
boolean distintoDeDos = numero != 2;  
boolean mayorOigualQueCinco = numero >= 5;
```

```
System.out.println(menorQueSiete);  
System.out.println(igualQueNueve);  
System.out.println(distintoDeDos);  
System.out.println(mayorOigualQueCinco);
```

Comparaciones entre números

Comparaciones (==, !=, <, >, <=, >=)

```
int numero = 5;
boolean menorQueSiete = numero < 7;
boolean igualQueNueve = numero == 9;
boolean distintoDeDos = numero != 2;
boolean mayorOigualQueCinco = numero >= 5;

System.out.println(menorQueSiete); // true
System.out.println(igualQueNueve);
System.out.println(distintoDeDos);
System.out.println(mayorOigualQueCinco);
```

Comparaciones entre números

Comparaciones (==, !=, <, >, <=, >=)

```
int numero = 5;
boolean menorQueSiete = numero < 7;
boolean igualQueNueve = numero == 9;
boolean distintoDeDos = numero != 2;
boolean mayorOigualQueCinco = numero >= 5;

System.out.println(menorQueSiete); // true
System.out.println(igualQueNueve); // false
System.out.println(distintoDeDos);
System.out.println(mayorOigualQueCinco);
```

Comparaciones entre números

Comparaciones (==, !=, <, >, <=, >=)

```
int numero = 5;
boolean menorQueSiete = numero < 7;
boolean igualQueNueve = numero == 9;
boolean distintoDeDos = numero != 2;
boolean mayorOigualQueCinco = numero >= 5;

System.out.println(menorQueSiete); // true
System.out.println(igualQueNueve); // false
System.out.println(distintoDeDos); // true
System.out.println(mayorOigualQueCinco);
```

Comparaciones entre números

Comparaciones (==, !=, <, >, <=, >=)

```
int numero = 5;
boolean menorQueSiete = numero < 7;
boolean igualQueNueve = numero == 9;
boolean distintoDeDos = numero != 2;
boolean mayorOigualQueCinco = numero >= 5;

System.out.println(menorQueSiete); // true
System.out.println(igualQueNueve); // false
System.out.println(distintoDeDos); // true
System.out.println(mayorOigualQueCinco); // true
```

Comparación entre palabras

```
String nombre1 = "Belen";  
String nombre2 = "Nebil";  
boolean comparacion1 = nombre1.equals(nombre2);  
System.out.println(comparacion1);
```



`.equals()`

Comparación entre palabras

```
String nombre1 = "Belen";  
String nombre2 = "Nebil";  
boolean comparacion1 = nombre1.equals(nombre2);  
System.out.println(comparacion1); // false
```



`.equals()`

Comparación entre palabras

```
String nombre1 = "Belen";  
String nombre2 = "Nebil";  
boolean comparacion1 = nombre1.equals(nombre2);  
System.out.println(comparacion1); // false
```

`.equals()`

```
int largo1 = nombre1.length();  
int largo2 = nombre2.length();  
boolean comparacion2 = largo1 == largo2;  
System.out.println(comparacion2);
```

`.length()`

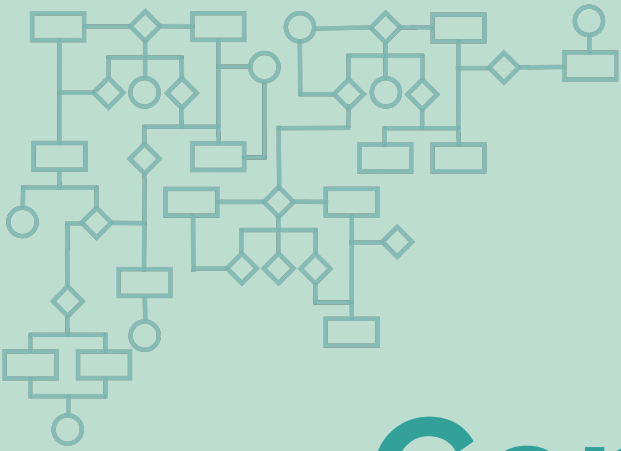
Comparación entre palabras

```
String nombre1 = "Belen";  
String nombre2 = "Nebil";  
boolean comparacion1 = nombre1.equals(nombre2);  
System.out.println(comparacion1); // false
```

`.equals()`

```
int largo1 = nombre1.length();  
int largo2 = nombre2.length();  
boolean comparacion2 = largo1 == largo2;  
System.out.println(comparacion2); // true
```

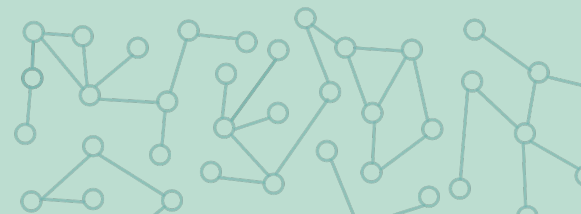
`.length()`



Control de flujo



if



```
if (condicion) {  
    // bloque de código que sólo se ejecuta  
    // en caso de que sí se cumpla la condición  
}
```

else



```
if (condicion) {  
    // bloque de código que sólo se ejecuta  
    // en caso de que sí se cumpla la condición  
}  
else {  
    // bloque de código que se sólo ejecuta  
    // en caso de que no se cumpla la condición  
}
```

else if



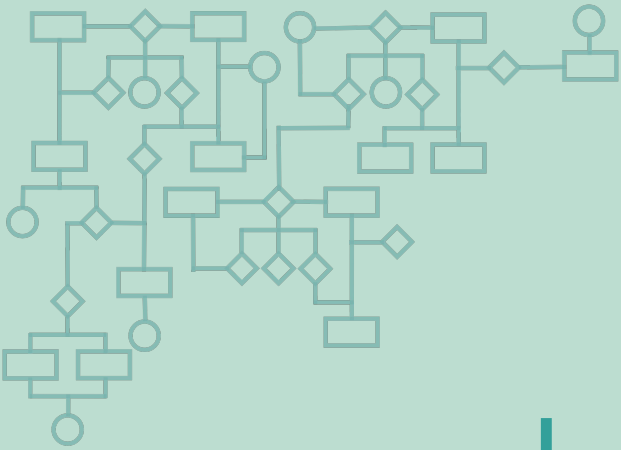
```
if (condicion) {  
    // bloque de código que sólo se ejecuta  
    // en caso de que sí se cumpla la condición  
}  
else if (condicion2) {  
    // bloque de código que sólo se ejecuta  
    // en caso de que sí se cumpla la condición 2  
}  
else {  
    // bloque de código que sólo se ejecuta  
    // en caso de que no se cumpla ninguna condición  
}
```

Clase N°3

Control de flujo

Taller de programación
Computines





[www.hackerrank.com/
c03-control-de-flujo](http://www.hackerrank.com/c03-control-de-flujo)

