



Clase 0 - Introducción a la programación

IIC1103 Sección 9 – 2019-2

Profesor: Felipe López

Fecha: 6 de agosto del 2019

Contenidos

1. Presentación
2. Programa
3. ¿Qué es un algoritmo?
4. Instalación Python

¿Quién soy?

- Ingeniero Civil Industrial mención Tecnologías de la Información
- Magíster en Ciencias de la Ingeniería mención Ciencias de la Computación
- Estudiante de Doctorado en Ciencias de la Ingeniería mención Ciencias de la Computación



¿Quién soy?

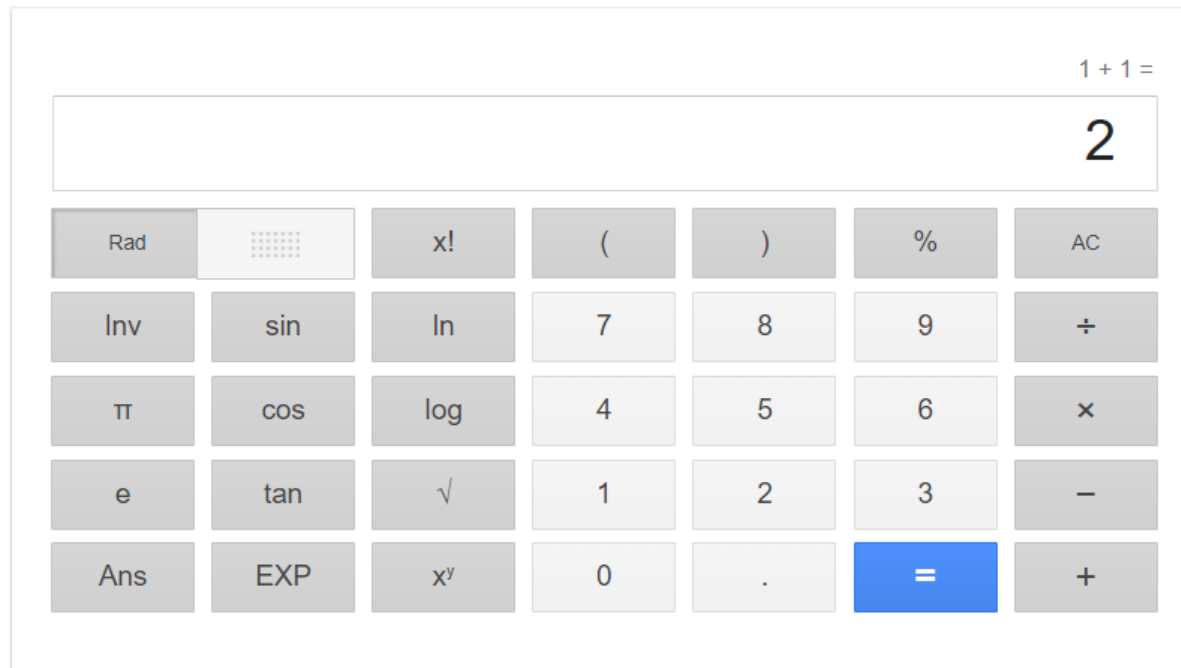
- Mi investigación: Medición del pensamiento crítico



¿De qué se trata el
curso?

¿Qué es la programación?

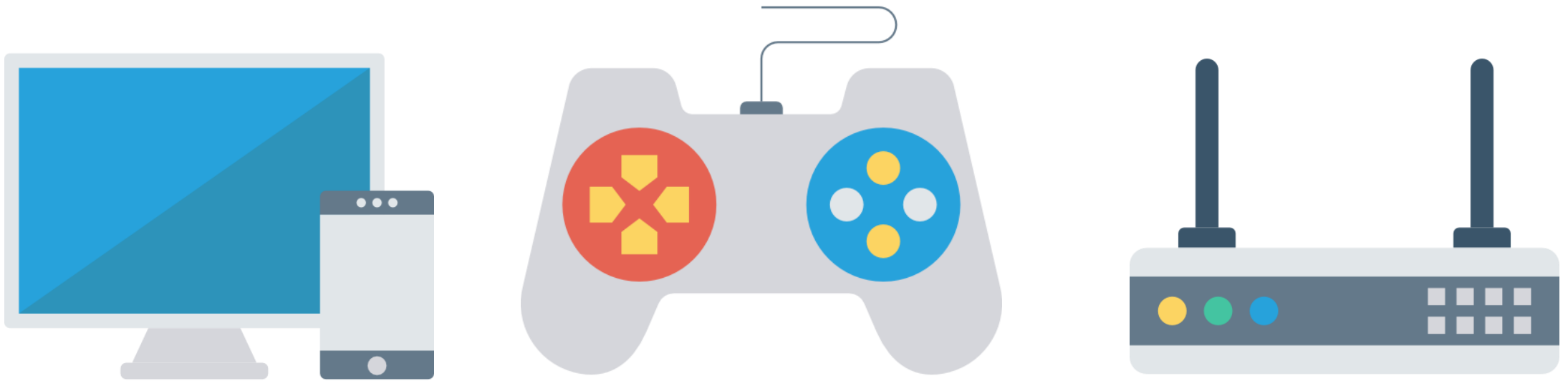
- Es darle instrucciones a un computador para poder ejecutar una tarea.
- Ejemplo:



Lenguajes de programación

- Estas instrucciones se dan a través una secuencia de comandos. El computador es capaz de interpretar esta secuencia de comando por medio de distintos **lenguajes de programación**.
- Hay **muchos** lenguajes de programación. Todos tienen distintos orígenes y propósitos.
- Algunos se ocupan en los más diversos ámbitos.
- Otros son sumamente específicos.

Programación en el mundo de hoy



Programación en el mundo de hoy



Programa del Curso

Programa

<https://tinyurl.com/programaIIIC110320192S9>



Instancias formales

- Cátedras: martes y jueves a las 10:00
- Laboratorios: lunes a jueves 15:30 – 18:30
- Ayudantías: Cuando sea necesario

Evaluaciones

- 2 Interrogaciones:
 - I1: Martes 24 de septiembre 18:30
 - I2: Viernes 30 de octubre 18:30
- Examen (E): Miércoles 27 de noviembre 15:30
- Tres tareas.

Notas

La nota final (NF) del curso se calcula en base a la siguiente fórmula:

$$NF = 15\% \cdot I1 + 15\% \cdot I2 + 30\% \cdot E + 10\% \cdot T1 + 10\% \cdot T2 + 10\% \cdot T3 + 10\% \cdot NP$$

Nota de participación

- Durante el semestre habrá un total de X desafíos.
- Se borrará la peor nota de esos desafíos (incluso si no hicieron uno)
- La nota de participación se calculará como el promedio simple de X-1 desafíos (después de borrar el peor).
- No obstante, existirán muchos bonus:
 - Cada clase de repaso antes de una interrogación o examen significará una bonificación de 2 décimas a la nota de participación.
 - Si van a las 3 clases de repaso antes de una interrogación significará un bonus total de 8 décimas en la nota de participación (por sobre el promedio calculado que se explicó anteriormente).
 - **La nota de participación no puede ser mayor que 7.0**
 - Es posible que hayan otras instancias “bonus”...

¿Cómo paso el ramo?

- Promedio de evaluaciones:

$$NE = \frac{(15\% \cdot I1 + 15\% \cdot I2 + 30\% \cdot E)}{60\%} \geq 4.0$$

- Promedio de tareas:

$$NT = \frac{(10\% \cdot T1 + 10\% \cdot T2 + 10\% \cdot T3)}{30\%} \geq 4.0$$

... y obviamente, que el promedio del ramo sea mayor que 4.0

Bibliografía

Referencias

- Python software foundation, Python v3 Documentation, <http://docs.python.org/3/>.
- V. Ceder. The quick python book. Manning Publications Co., 2010.
- A. B. Downey. Think Python: How to think like a computer scientist. Green Tea Press, 2013
 - Versiones web:
 - <http://www.greenteapress.com/thinkpython/>
 - <http://interactivepython.org/courselib/static/thinkcspy/index.html>.
- J. M. Zelle. Python programming: An introduction to computer science. Franklin, Beedle & Associates, Inc., 2nd edition, 2010.
- **Py-Libre, Apunte interactivo para el curso Introduccion a la Programacion** <http://runest.ing.puc.cl>

- ¿La prueba se puede hacer en parejas?

- ¿La prueba se puede hacer en parejas?
- Nuestras pruebas son iguales porque estudiamos juntos

- ¿La prueba se puede hacer en parejas?
- Nuestras pruebas son iguales porque estudiamos juntos
- Nuestras pruebas son iguales porque un ayudante nos explicó a los dos

- ¿La prueba se puede hacer en parejas?
- Nuestras pruebas son iguales porque estudiamos juntos
- Nuestras pruebas son iguales porque un ayudante nos explicó a los dos
- Le pagamos a alguien para que nos ayudara con nuestra prueba

- ¿La prueba se puede hacer en parejas?
- Nuestras pruebas son iguales porque estudiamos juntos
- Nuestras pruebas son iguales porque un ayudante nos explicó a los dos
- Le pagamos a alguien para que nos ayudara con nuestra prueba
- ¡Solo le mostré mi prueba! No dejé que me copiara

- ¿La prueba se puede hacer en parejas?
- Nuestras pruebas son iguales porque estudiamos juntos
- Nuestras pruebas son iguales porque un ayudante nos explicó a los dos
- Le pagamos a alguien para que nos ayudara con nuestra prueba
- ¡Solo le mostré mi prueba! No dejé que me copiara
- Solo se la mandé para que la corrigiera
- Encontré los ejercicios de la prueba por internet

- ¿La prueba se puede hacer en parejas?
- Nuestras pruebas son iguales porque estudiamos juntos
- Nuestras pruebas son iguales porque un ayudante nos explicó a los dos
- Le pagamos a alguien para que nos ayudara con nuestra prueba
- ¡Solo le mostré mi prueba! No dejé que me copiara
- Solo se la mandé para que la corrigiera
- Encontré los ejercicios de la prueba por internet

¡LO MISMO APLICA PARA LAS TAREAS!

Tareas

- Las tareas del curso son **individuales**
- **No** está permitido realizar la tarea en **grupos** (ni parejas)
- **No** está permitido el intercambio o envío de código (ni trozos de código)
- **No** está permitido recibir código (ni trozos de código) de otras personas
- **No** está permitido entregarle la tarea a otra persona, aún si es “solo para ver el código”.
- **No** está permitido compartir contraseñas
- **No** está permitido recibir ayuda en forma de código

Temas éticos

Código de honor

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad”.

<http://www.uc.cl/codigodehonor/el-codigo>

Política de Integridad Académica DCC

Todo trabajo presentado por un alumno para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno, sin apoyo en material de terceros. [...]

En particular, si un alumno copia un trabajo, o si a un alumno se le prueba que compró o intentó comprar un trabajo, obtendrá nota final 1.1 en el curso y se solicitará a la Dirección de Docencia de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral.

Por “copia” se entiende incluir en el trabajo presentado como propio partes hechas por otro alumno. En este caso, la sanción anterior se aplicará a todos los alumnos involucrados. Por “compra” se entiende presentar como propio un trabajo hecho por otra persona.

En resumen

- El no hacer la tarea de forma individual (en cualquiera de las formas antes expuestas), te expone a la **sanción de reprobar el curso con un 1.1 y que esto sea notificado a la Dirección de Pregrado.**
- Esta decisión es del professor y del grupo de profesores del curso (**y se toma caso a caso**; no influye si en otros semestres/secciones/casos se han tomado decisiones distintas).

En resumen

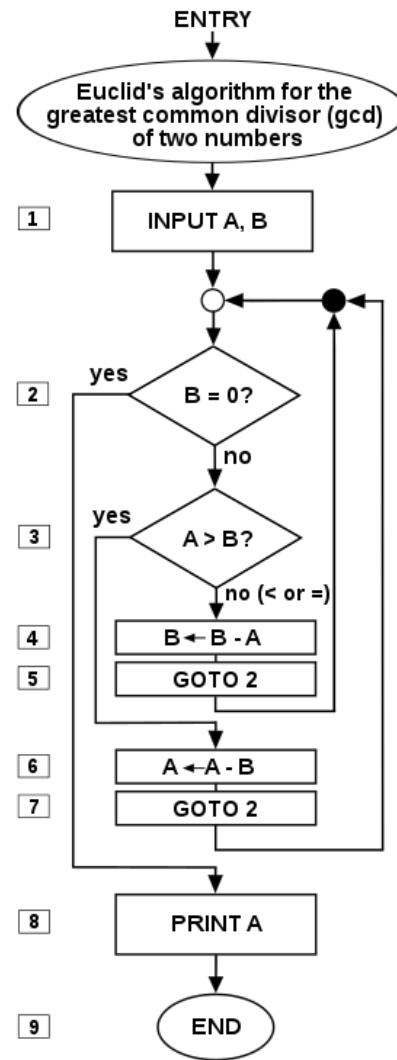
- ¡Aprovechen instancias formales!
 - Laboratorios
 - Ayudantes
 - Yo
 - Foros
 - Ayudantías extras*
- No dejen todo a última hora, vayan trabajando de a poco, y realicen entregas parciales en las tareas 😊

Algoritmos

¿Qué *#%”*! Es un algoritmo?

Definición

“Secuencia ordenada de pasos cuyo objetivo es resolver un problema (o generar un *output*) desde un estado inicial (o *input*)”



Receta

- 1 taza de zapallo cocido y molido
- 3 cucharadas de manteca derretida
- 1 cucharadita de sal
- 2 tazas de harina
- ½ taza de leche o agua caliente
- ½ litro o más de aceite para freír

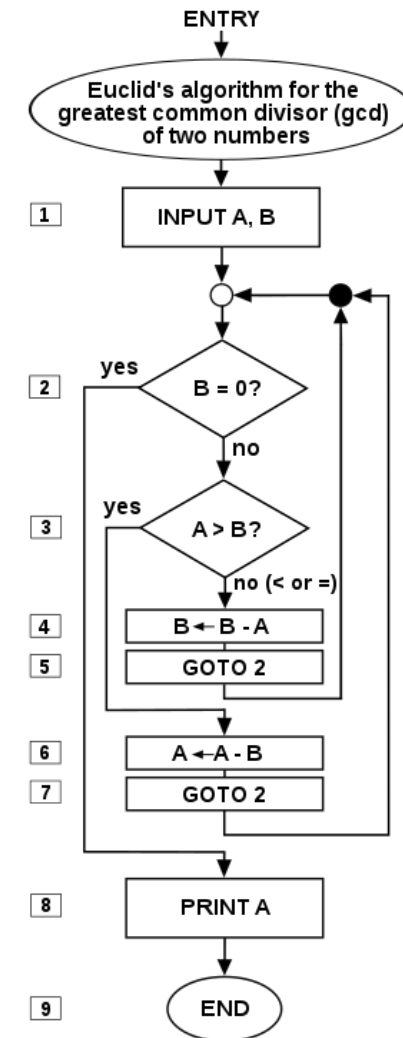
Lo primero que hay que tener en cuenta para saber cómo hacer sopaipillas, es hacer un “volcán” con la harina dejando un hueco al medio, vierta al centro la manteca derretida con la leche o el agua, la sal y el zapallo previamente molido hasta formar una pasta suave, mezcle todo hasta formar una masa suave y elástica, esta no debe tener grumos y no se tiene que pegar a la mesa.

El siguiente paso para saber cómo hacer sopaipillas es amasar la masa con un uslero o a mano dejándola de aproximadamente 5mm de grosor, luego cortarla en círculos de 10cm (esto es a gusto, pueden ser más gruesas y más pequeñas), se recomienda pincharlas con un tenedor para que al momento de freírlas la masa no se arruine.

El último paso es freír la masa en un sartén profundo y caliente el aceite a fuego alto, para probar el aceite lance un pequeño trozo de masa, debe burbujear y flotar en la superficie, coloque de 2 a 3 sopaipillas y fríalas 1 minuto por lado, no se deben dorar demasiado. Una vez fritas se sacan de la freidora y dejan en papel absorbente.



¿Podríamos hacer algo parecido a esto con la receta anterior?

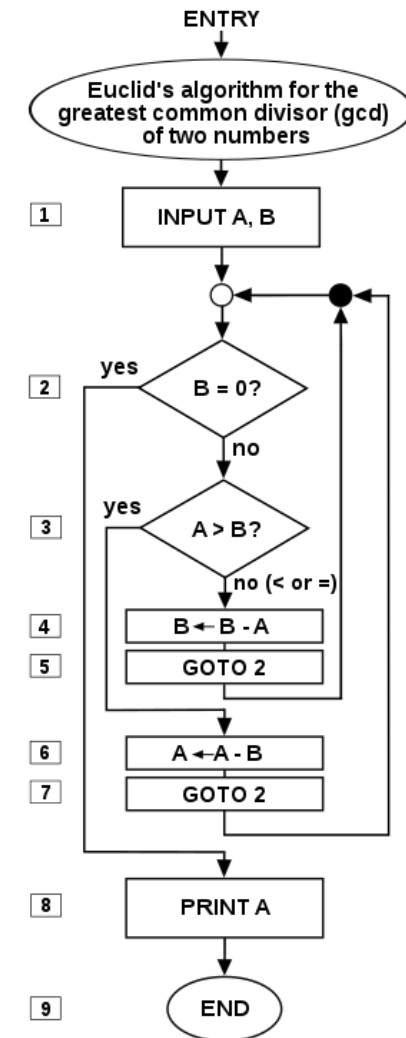


¿Podríamos hacer algo parecido a esto con la receta anterior?

¡Sí! Al menos intenten responder:

¿Cuál es mi estado inicial o *input*?

¿Cuál es mi estado final o *output*?



¿Cómo llego desde Plaza de Armas a la Moneda?

Plaza de Armas

Santiago, Región Metropolitana

↑ Head south on Plaza de Armas toward Compañía de Jesús

68 m

➔ Turn right onto Compañía de Jesús

250 m

↶ Turn left onto Morandé

[i Destination will be on the right](#)

450 m

La Moneda Palace

Moneda S/N, Santiago, Región Metropolitana

Plaza de Armas

Santiago, Región Metropolitana

↑ Head south on Plaza de Armas toward Compañía de Jesús

68 m

↑ Continue onto Paseo Ahumada

260 m

➔ Turn right onto Agustinas

250 m

↶ Turn left onto Morandé

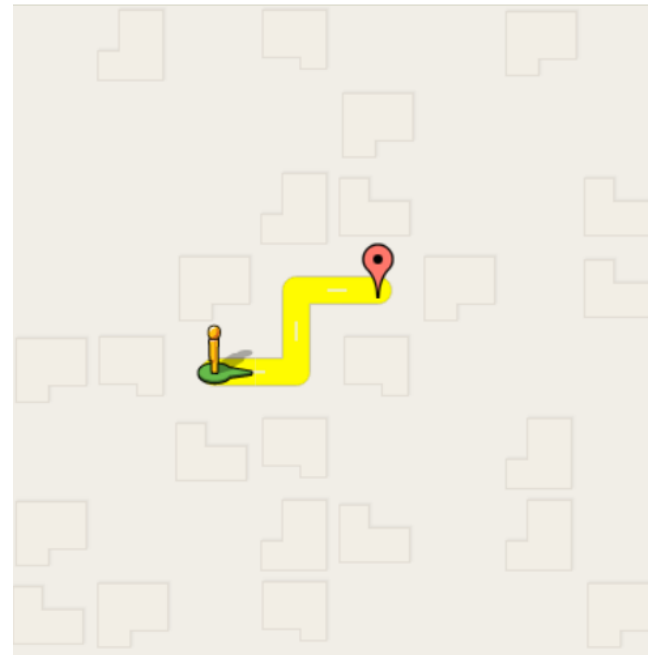
[i Destination will be on the right](#)

200 m

La Moneda Palace

Moneda S/N, Santiago, Región Metropolitana

Link



▶ Ejecutar el programa

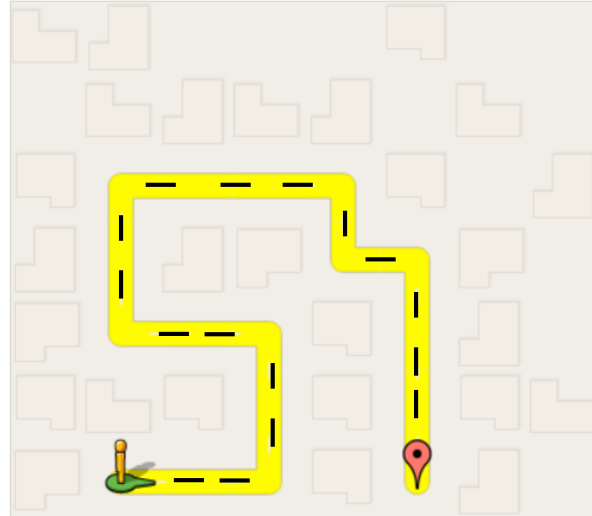
avanzar



girar a la izquierda ↶

girar a la derecha ↷

Blockly

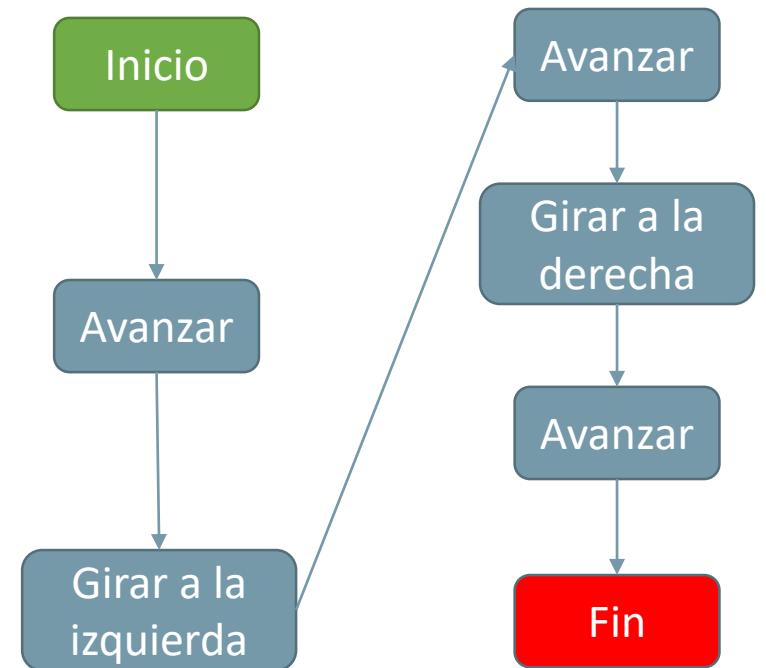
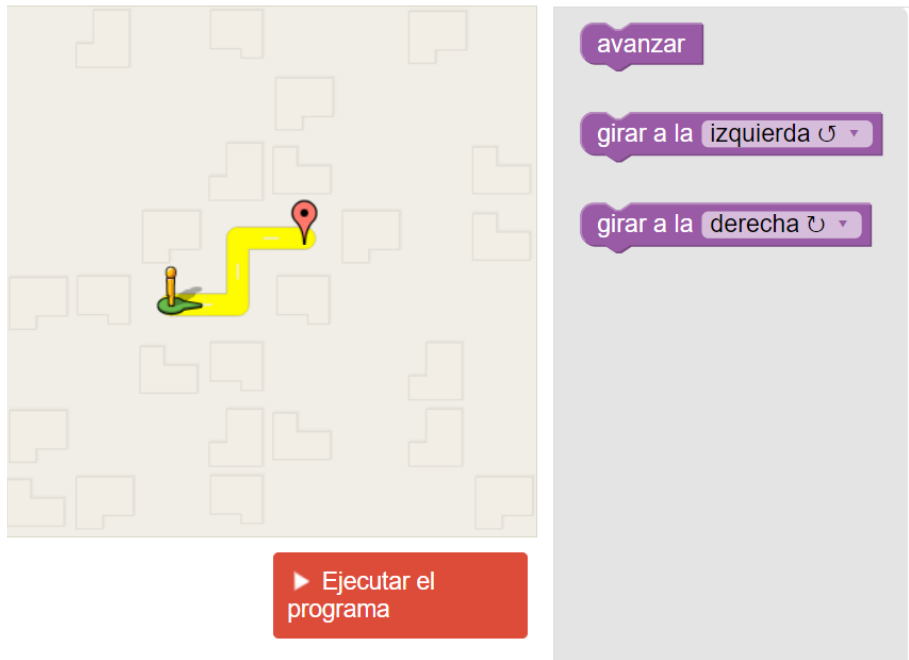
Link



ACTIVIDAD: ¿Cómo puede llegar  hasta  ?
Indica la secuencia de bloques necesaria usando los números 1, 2 y 3.

Diagramas de Flujo

- Representación gráfica de un algoritmo

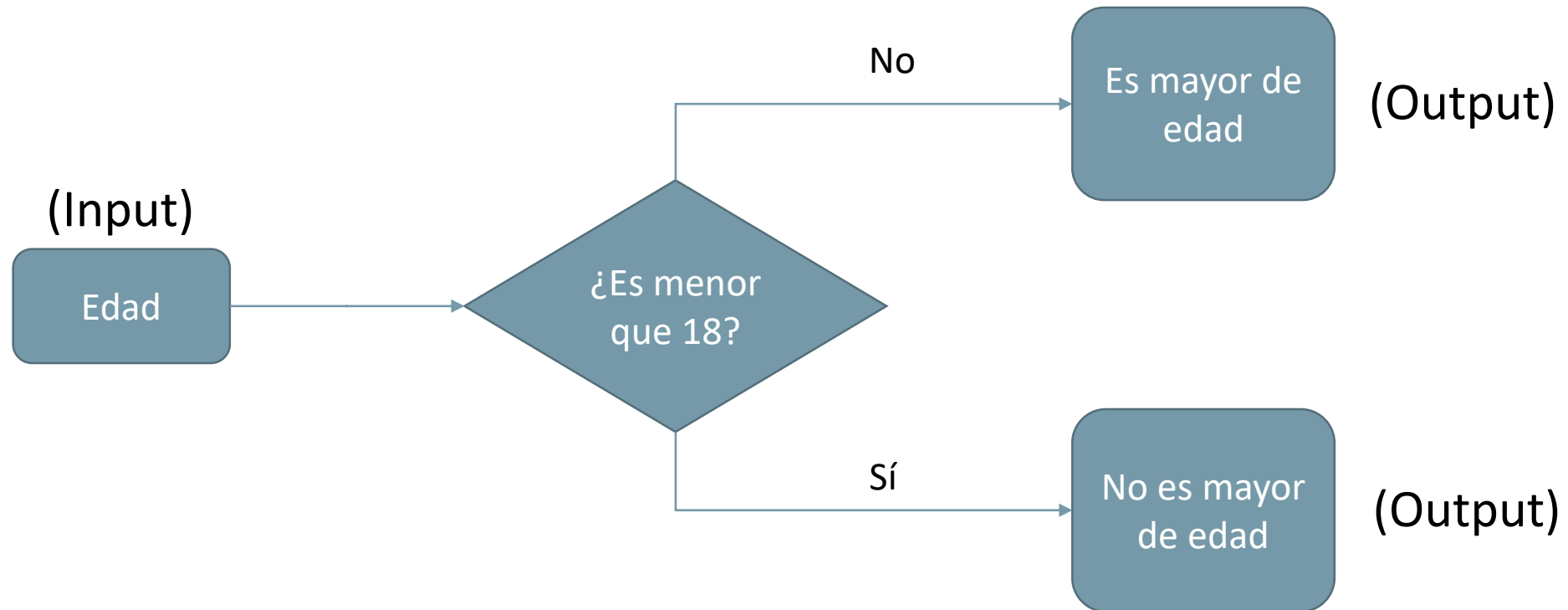


Diagramas de Flujo

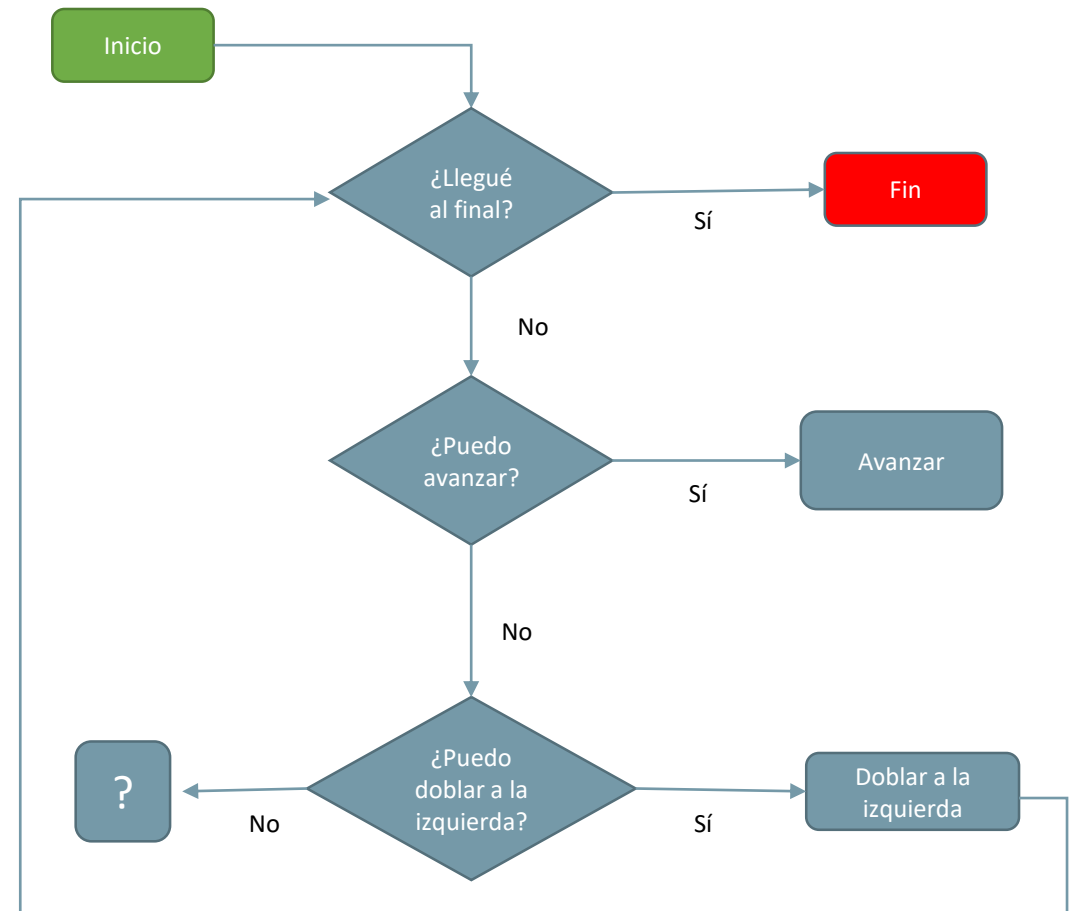
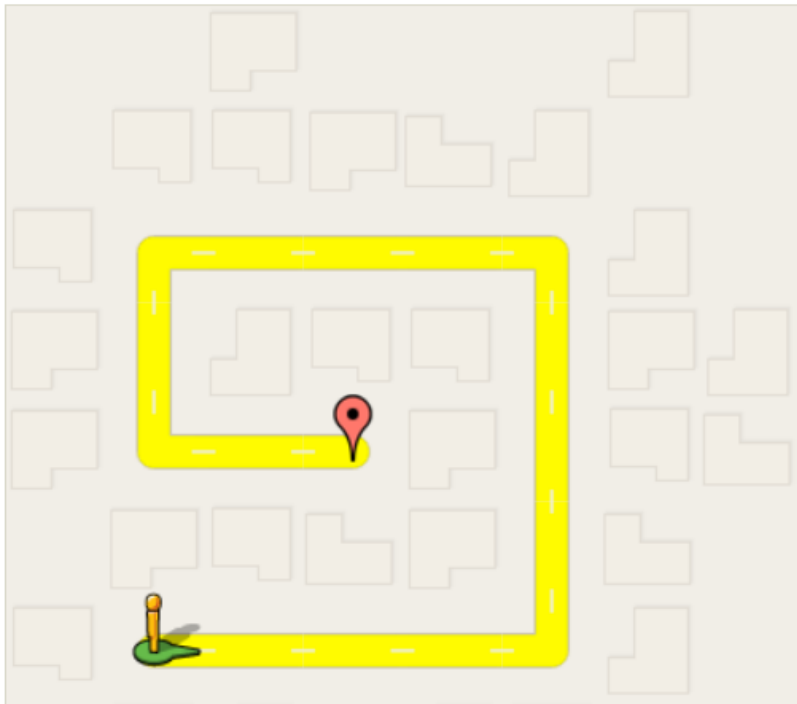
¿Qué pasa si tenemos un condicional?

Por ejemplo, queremos escribir un algoritmo que pueda determinar si una persona es o no mayor de edad.

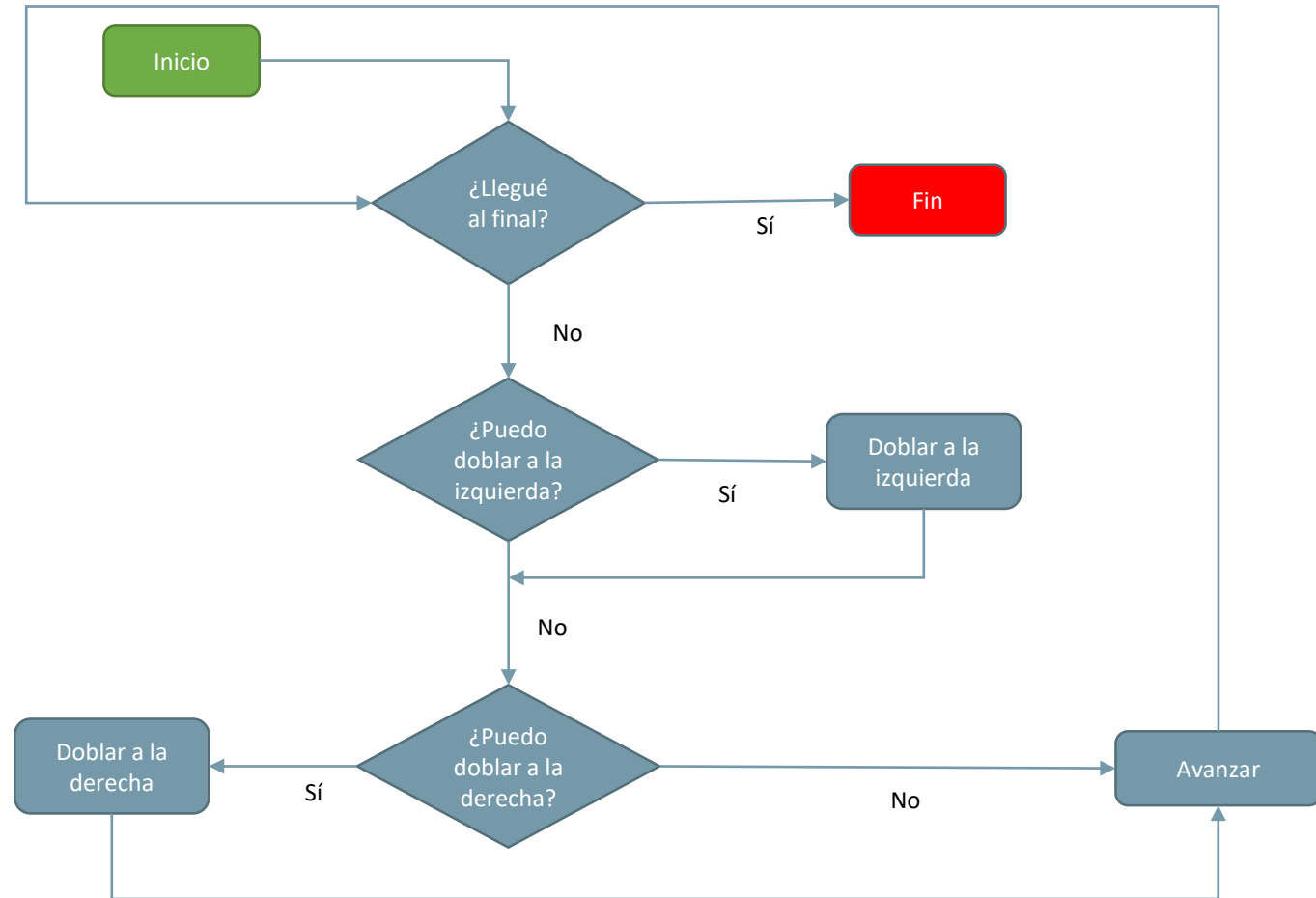
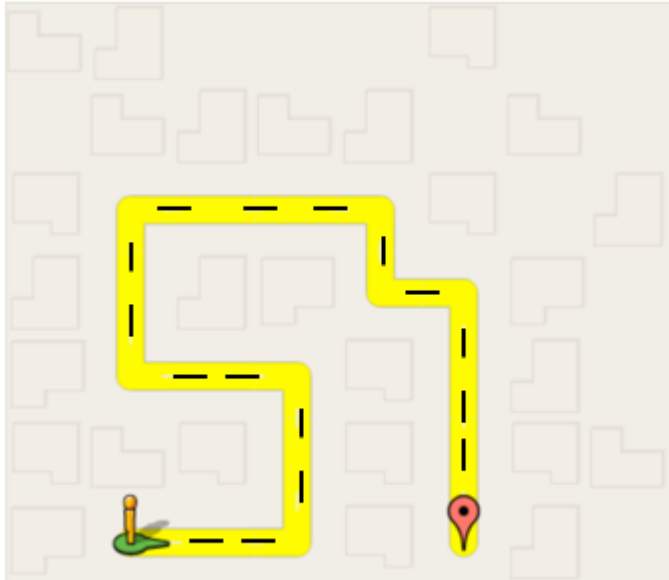
Diagramas de Flujo



Diagramas de Flujo

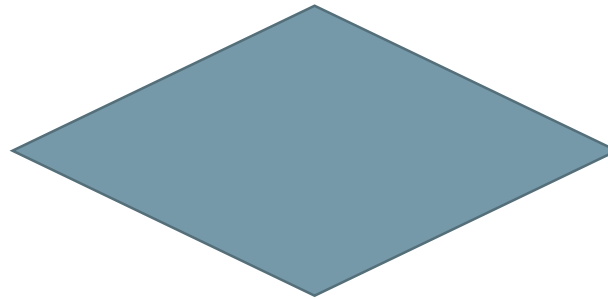


Diagramas de Flujo



Diagramas de Flujo

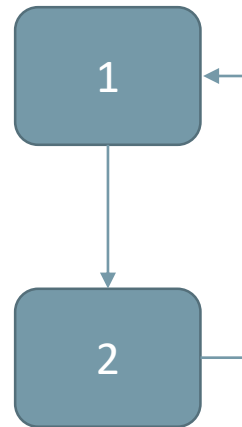
Para incluir un condicional en un diagrama de flujo, se debe ocupar la figura:



Esta figura debe contener alguna pregunta que derive en más de una opción. La formulación de la pregunta es vital para poder determinar qué camino sigue.

Diagramas de Flujo

¿Qué pasa si queremos volver a un paso intermedio del algoritmo?



Diagramas de Flujo

¿Qué pasa si queremos volver a un paso intermedio del algoritmo?

Por ejemplo, escribamos un algoritmo que saque la potencia de un número n .

Diagramas de Flujo

Escribamos un algoritmo que saque la potencia de un número n .

Estado inicial:

Secuencia de pasos:

Estado final:

Diagramas de Flujo

Por ejemplo, escribamos un algoritmo que saque el valor de 2^5 .

Estado inicial: 2

Secuencia de pasos:

Estado final:

Diagramas de Flujo

Por ejemplo, escribamos un algoritmo que saque el valor de 2^5 .

Estado inicial: 2

Secuencia de pasos: Multiplicar 2 por 2 sucesivamente 5 veces.

Estado final:

Diagramas de Flujo

Por ejemplo, escribamos un algoritmo que saque el valor de 2^5 .

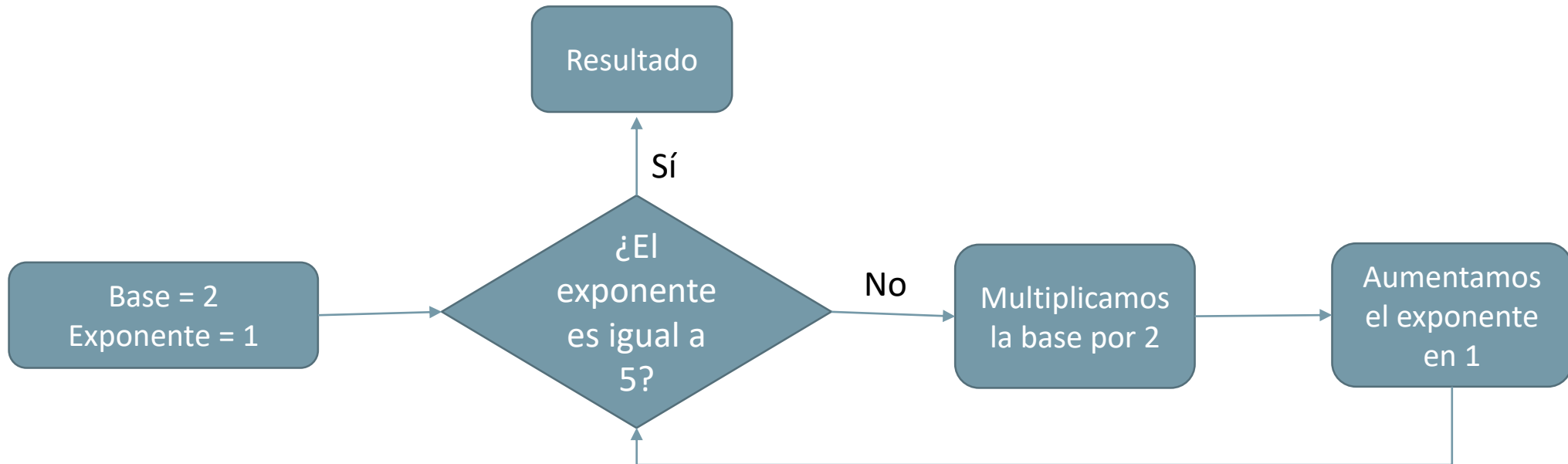
Estado inicial: 2

Secuencia de pasos: Multiplicar 2 por 2 sucesivamente 5 veces.

Estado final: 2^5

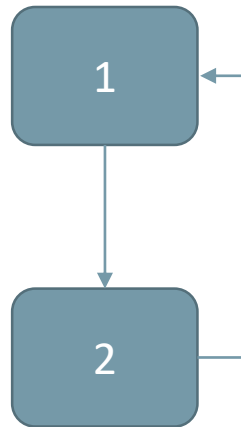
Diagramas de Flujo

Por ejemplo, escribamos un algoritmo que saque el valor de 2^5 .

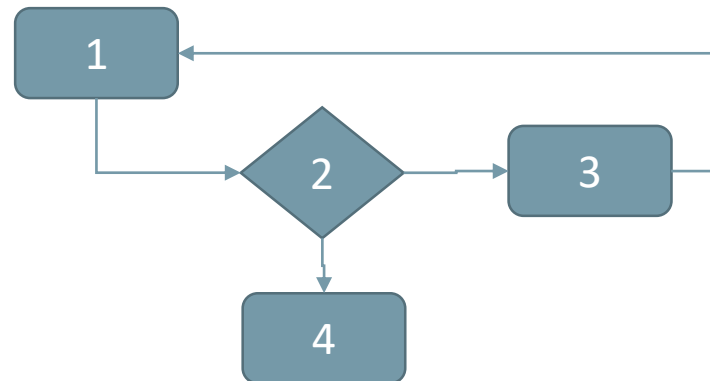
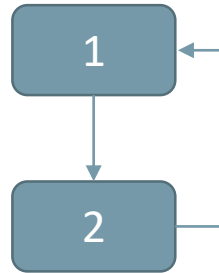


Diagramas de Flujo

En cualquier momento uno puede hacer un ciclo en un diagrama de flujo. Simplemente hay que hacer el camino correspondiente.



Diagramas de Flujo



Es importante cuidar que al crear un ciclo el diagrama de flujo permita alguna “salida”. De no ser así,
¡El algoritmo nunca terminará!

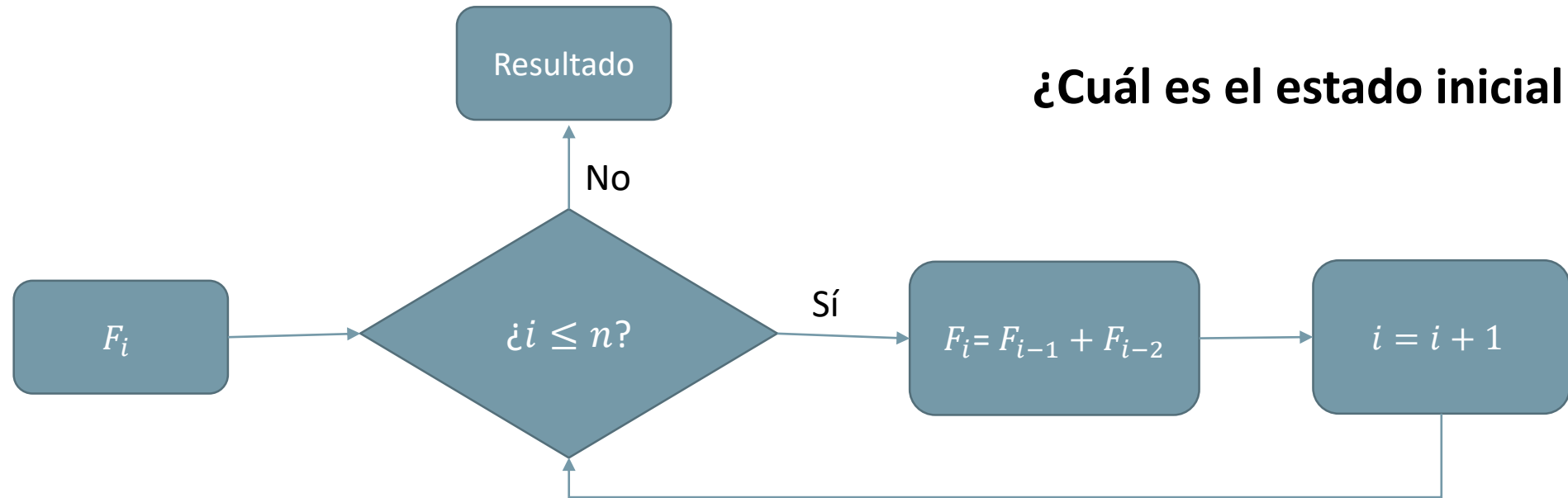
Ejemplos

Escribir un algoritmo que nos devuelva el n-ésimo término de una secuencia de Fibonacci.

$$F_n = F_{n-1} + F_{n-2}$$

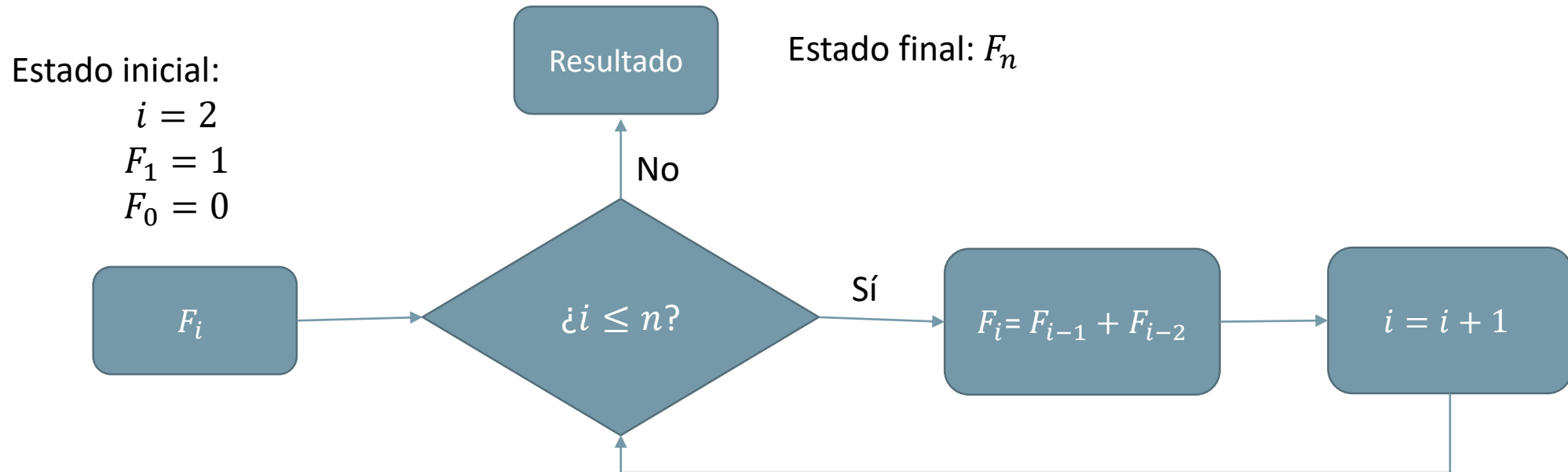
Ejemplos

Escribir un algoritmo que nos devuelva el n-ésimo término de una secuencia de Fibonacci.



Ejemplos

Escribir un algoritmo que nos devuelva el n -ésimo término de una secuencia de Fibonacci.



Introducción a la Programación

(ahora sí)

¿Qué es programar?

- Es poder entregarle un algoritmo a un computador para que sea capaz de resolver un problema.
- Le entregamos al computador cada uno de los pasos del algoritmo como un comando.
- El computador debe ser capaz de interpretar esta secuencia de comandos mediante algún tipo de lenguaje.

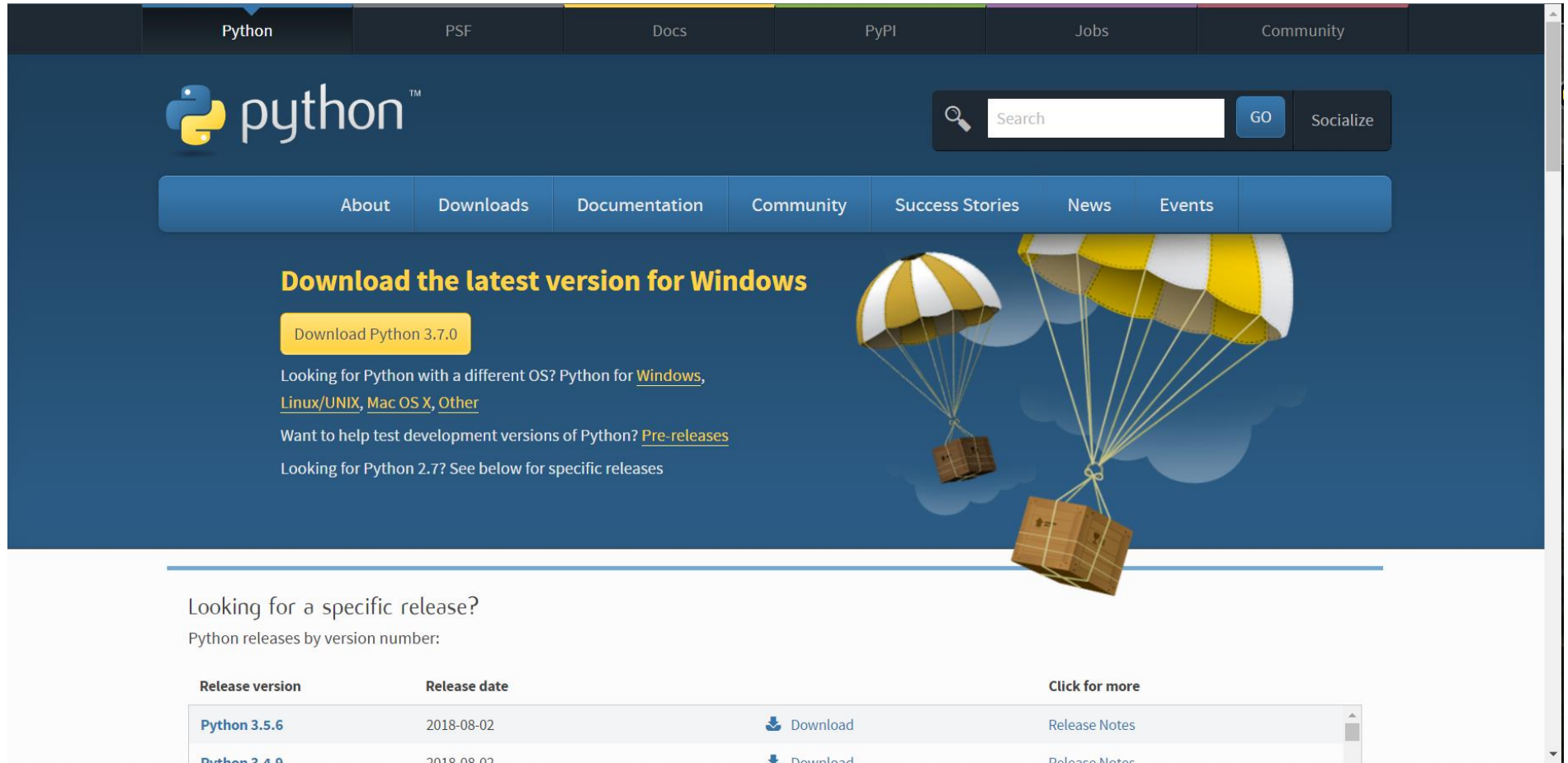
Lenguaje de Programación



- El lenguaje de programación que ocuparemos es Python v3.x
- Es un lenguaje simple y efectivo para los objetivos del curso

Instalación

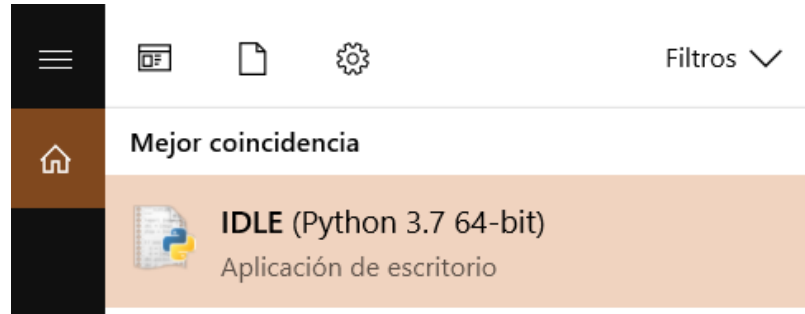
- [Link](#)



The screenshot shows the Python.org website. The top navigation bar includes links for Python, PSF, Docs, PyPI, Jobs, and Community. Below this is a search bar and a 'Socialize' button. A secondary navigation bar contains links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a large banner with the text 'Download the latest version for Windows' and a yellow button labeled 'Download Python 3.7.0'. Below the button, there are links for 'Python for Windows, Linux/UNIX, Mac OS X, Other', 'Pre-releases', and 'Looking for Python 2.7? See below for specific releases'. To the right of the text is an illustration of two parachutes carrying boxes. Below the banner, there is a section titled 'Looking for a specific release?' with the text 'Python releases by version number:'. This section contains a table with two columns: 'Release version' and 'Release date'. The table lists 'Python 3.5.6' (released 2018-08-02) and 'Python 2.7.15' (released 2018-08-02). Each row has a 'Download' link and a 'Release Notes' link.

Release version	Release date	Click for more
Python 3.5.6	2018-08-02	Download Release Notes
Python 2.7.15	2018-08-02	Download Release Notes

Instalación



```
Python 3.7.0a4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0a4 (v3.7.0a4:07c9d85, Jan  9 2018, 07:07:02) [MSC v.1900 64 bit (AMD
64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Python

- PROBLEMA: ¿Cómo podemos decirle al programa que imprima “Hola mundo”?

```
Python 3.7.0a4 (v3.7.0a4:07c9d85, Jan  9 2018, 07:07:02) [MSC v.1900 64 bit (AMD  
64)] on win32  
Type "copyright", "credits" or "license()" for more information.  
>>> print("Hola mundo")  
Hola mundo
```




Así se va a ver en IDLE

Python

- PROBLEMA: ¿Cómo podemos decirle al programa que imprima “Hola mundo”?

```
print("Hola mundo")
```



Este es el código que pueden
copiar y pegar en su IDLE para
porbarlo

Tarea

- Bajar e instalar Python
- Abrir IDLE
- Escribir la siguiente línea de código

```
print("Hola mundo")
```

- Ejecutarla

Tarea

- Contestar la siguiente encuesta (si 90% del curso la contesta todos reciben una décima extra en la nota de participación):

<https://forms.gle/Q7Uq1DkE79uyBdf89>



Clase 0 - Introducción a la programación

IIC1103 Sección 9 – 2019-2

Profesor: Felipe López

Fecha: 6 de agosto del 2019