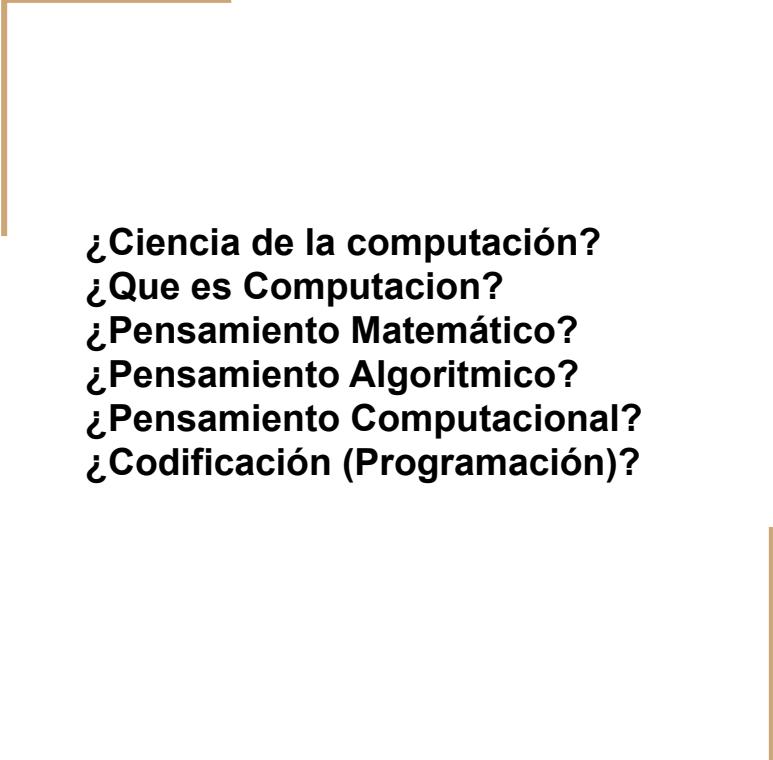


INTRODUCCIÓN



- ¿Ciencia de la computación?
- ¿Que es Computacion?
- ¿Pensamiento Matemático?
- ¿Pensamiento Algoritmico?
- ¿Pensamiento Computacional?
- ¿Codificación (Programación)?

Computer Science

An Overview

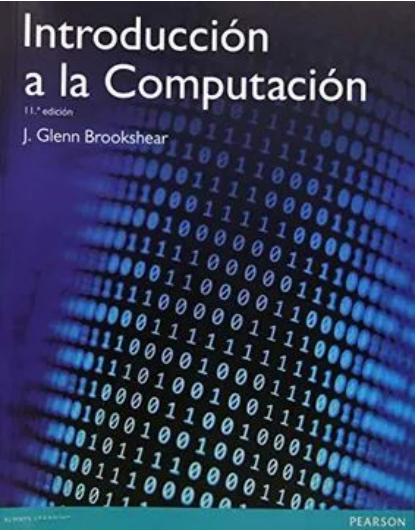
TWELFTH EDITION



J. Glenn Brookshear • Dennis Brylow

ALWAYS LEARNING

PEARSON

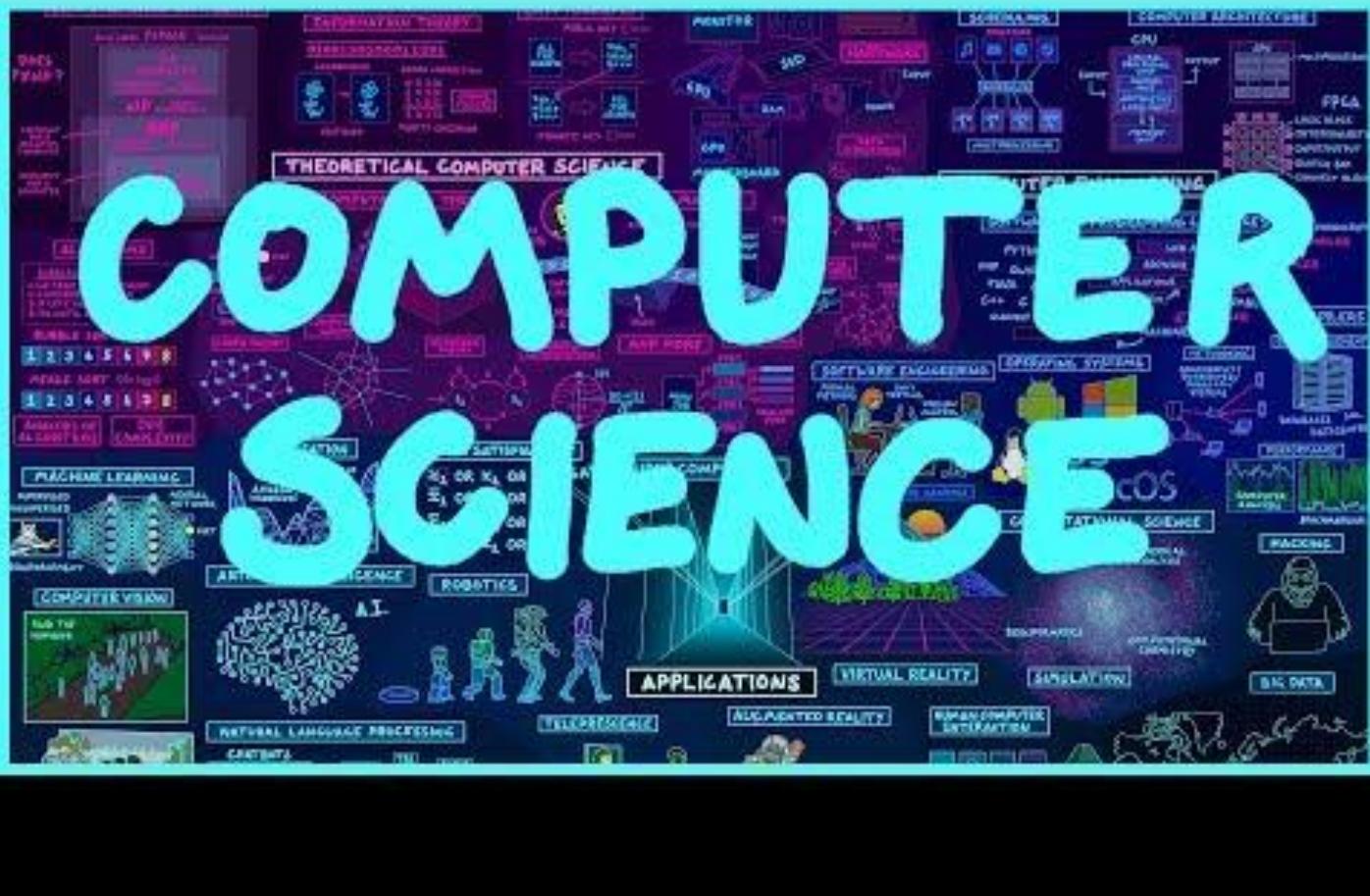


Introducción a la Computación

11.ª edición

J. Glenn Brookshear

PEARSON



Pensamiento Matemático

- Conteо
- Aritmética
- Álgebra
- Geometría
- Cálculo
- Teoría de Conjuntos
- Topología

Pensamiento Computacional

- Solución de problemas
- Modelamiento
- Análisis e interpretación de datos
- Estadísticas y probabilidad
- Simulación
- Minería de datos
- Redes
- Recolección de datos automática
- Juegos
- Razonamiento Algorítmico
- Robótica
- Programación

Pensamiento Algorítmico

Ver Fischbeck (2006), González y Trezise (2015), Lockwood et al. (2016), Sadykova y Llaníez (2018) y Sadykova e Ilbáñez (2019).

Pensamiento Estadístico

Ver Batanero Bernabeu (2013).

Pensamiento Variacional

Ver Martínez-López y Guadarrama-Pinto (2018) y Ondóñez-Ortega et al. (2019).

Pensamiento Matemático Avanzado

Ver Tall (2002), Dubinsky y Tall (2002) y Rasmussen et al. (2005).

Pensamiento Combinatorio

Ver Fernández Millán (2013).

Pensamiento Geométrico

Ver Thom y McGarvey (2015).

Pensamiento Algebráico

Ver Kilham y Bokking (2019).

Pensamiento Cuantitativo

Ver Ruiz y Valderrama (2004).

Pensamiento Lógico

Ver Coscañelli et al. (1976); y Voskoglou y Buckley (2012).

Pensamiento Crítico

Ver Voskoglou y Buckley (2012); Kaloglu y Gülbahar (2014) y Doleck et al. (2017).

Pensamiento Creativo

Ver Doleck et al. (2017) y Hidayat et al. (2018).

Pensamiento Numérico y Aritmético

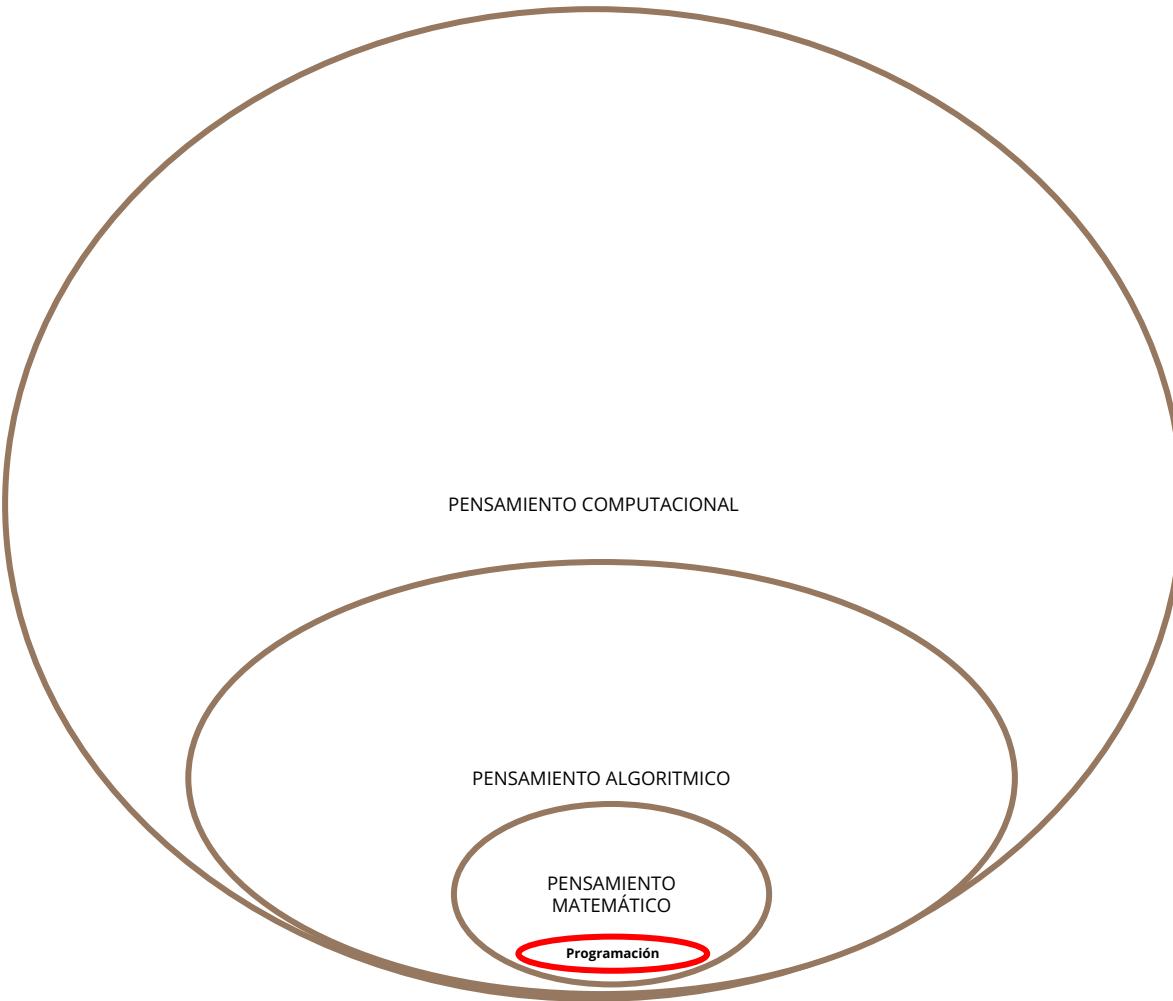
Ver Gallardo Romero (2004) y Congacha et al. (2018).

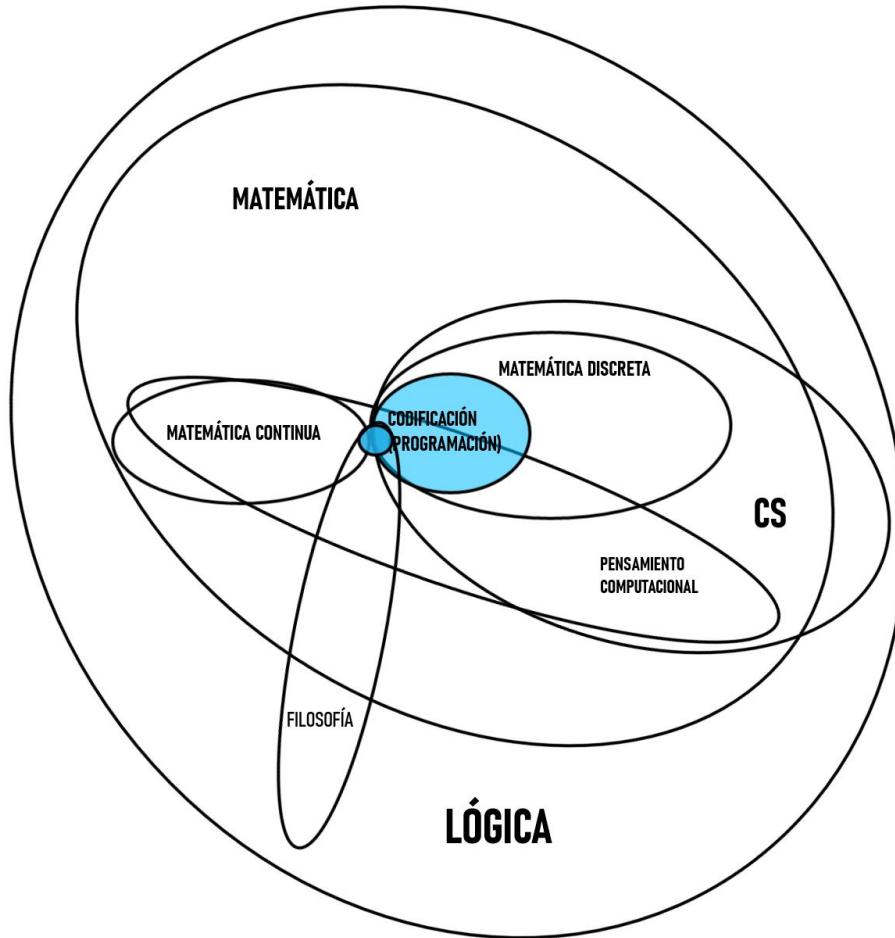


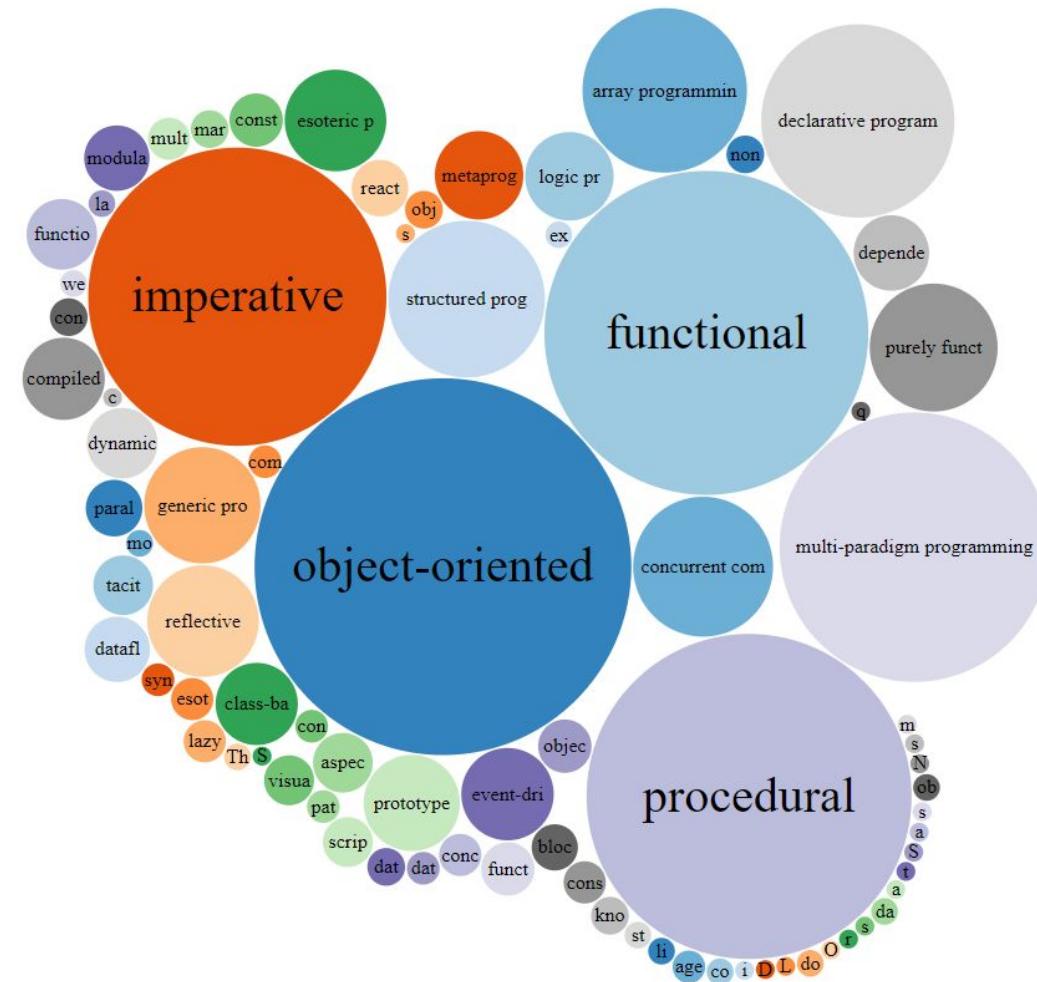
Pensamiento Computacional

- Simulación
- Minería de datos
- Redes
- Recolección de datos automática
- Juegos
- Razonamiento Algorítmico
- Robótica
- Programación
- Solución de problemas
- Modelamiento
- Análisis e interpretación de datos
- Estadísticas y probabilidad









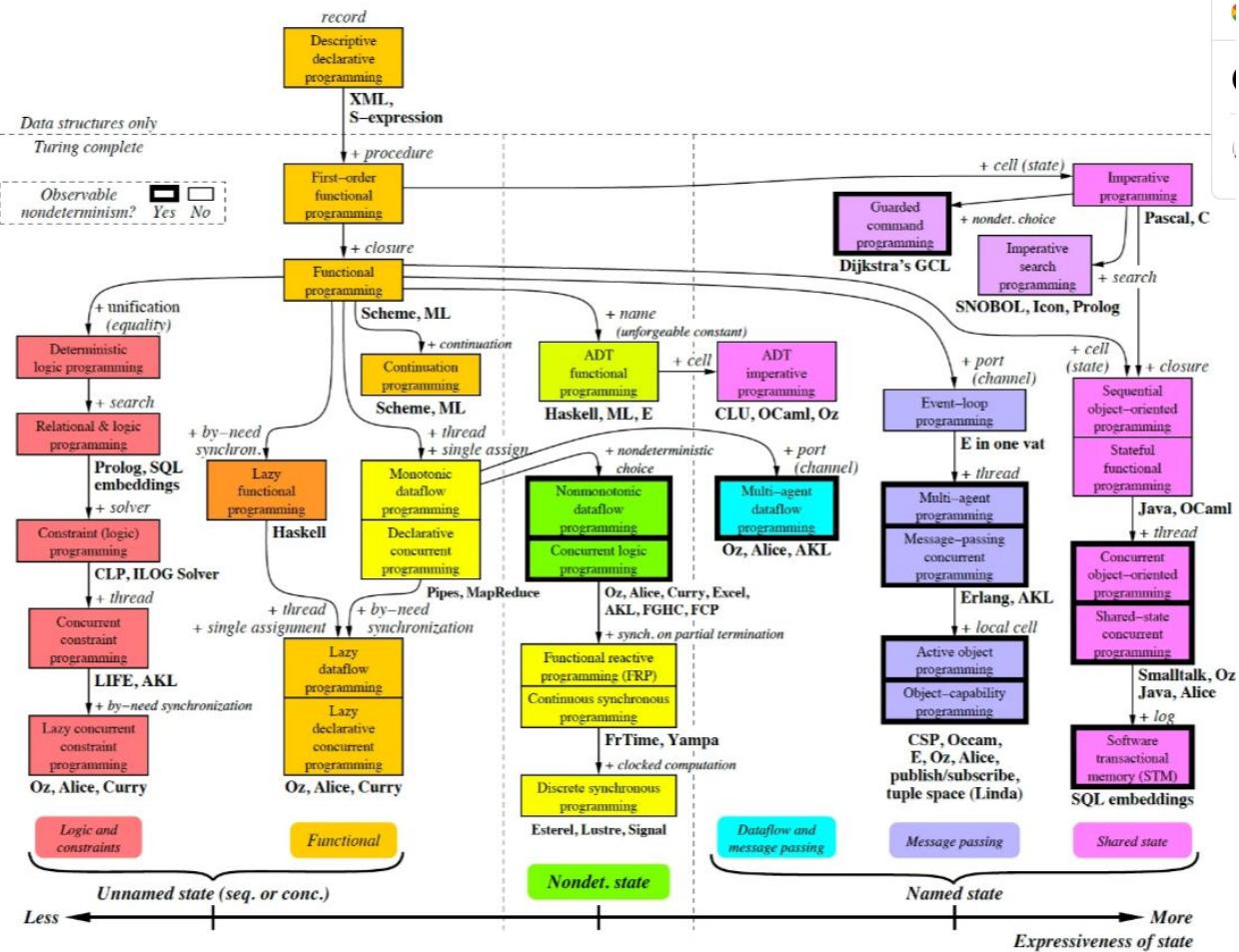
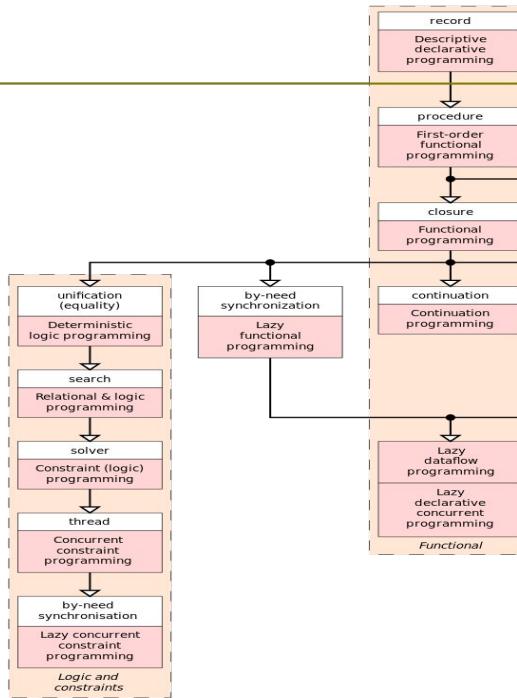


Figure 2. Taxonomy of programming paradigms

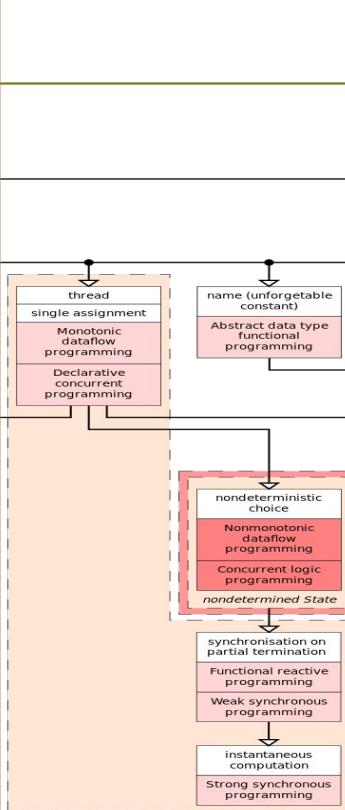
Markup languages (only Datastructures)

More declarative Paradigms

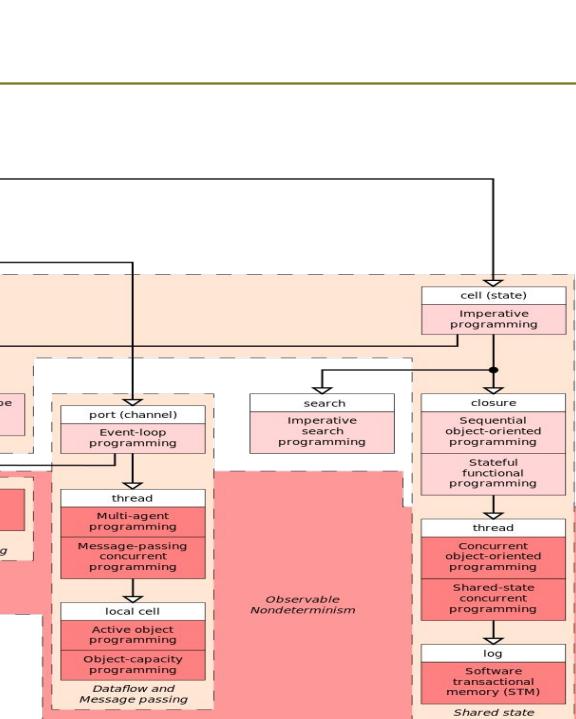
Unnamed state (sequential or concurrent)



Undeterministic state



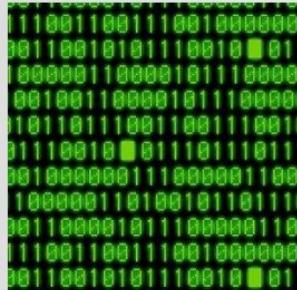
Named state



Turing complete Languages

Linea de tiempo generaciones de lenguajes de programacion

RAMIREZ VEGA EDWIN CARIM



Segunda generación

La segunda generación de lenguajes de programación consistía se identificaron como lenguajes ensambladores. Un lenguaje ensamblador convierte las secuencias de 0s y 1s a un lenguaje entendido por los seres humanos, como "añadir", "sumar", etc.



1940



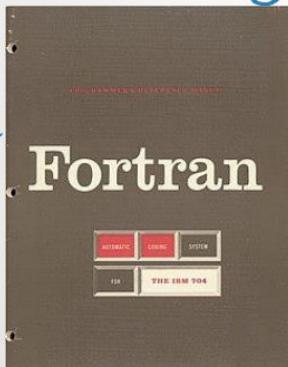
1950

Tercera generación

La tercera generación de lenguajes de programación se conoce como lenguajes de alto nivel. Un lenguaje de alto nivel tiene una gramática y sintaxis similar a las palabras en una oración. Un compilador se encarga de traducir el lenguaje de alto nivel a lenguaje ensamblador o código máquina. Todos los lenguajes de programación de software necesitan ser traducidos a código de máquina para utilizarlos.

Primera generación 1GL

La primera generación de lenguajes de programación consistía enteramente de una secuencia de 0s y 1s que los controles de la computadora interpretan como instrucciones, eléctricamente. Representan los primeros lenguajes de programación que la computadora podía entender, son conocidos como lenguaje máquina.



Cuarta generación

La tercera generación de lenguajes de programación se conoce como lenguajes de alto nivel. Un lenguaje de alto nivel tiene una gramática y sintaxis similar a las palabras en una oración. Un compilador se encarga de traducir el lenguaje de alto nivel a lenguaje ensamblador o código máquina. Todos los lenguajes de programación de software necesitan ser traducidos a código de máquina para utilizarlos.



1982



1995

Quinta generación

Imitación de la mente humana
La quinta generación de lenguajes de programación es utilizada para redes neuronales. Una red neuronal es una forma de inteligencia artificial que trata de imitar la mente humana.

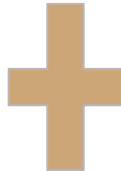


Conclusiones: Los lenguajes de programación han cambiado mucho a lo largo de los años, el juntarlos en generaciones nos ayuda a poder identificar y comparar dichos cambios

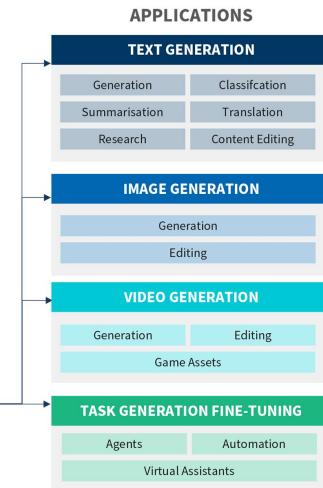
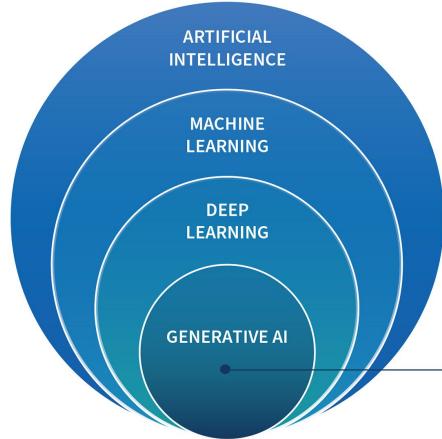
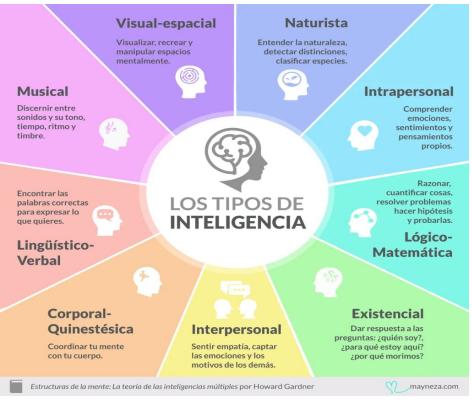
Fuentes:

- <https://www.timetoast.com/timelines/generaciones-de-los-lenguajes-de-programacion-78279214-40c7-4337-be29-0bcd0e376bd2>





GAMA ALTA	GAMA MEDIA	GAMA BAJA
CORE i7 9700	CORE i5 9600	CORE i3 9100
CORE i7 9700K	CORE i5 9600K	CORE i5 9300
RYZEN 7 3700X	RYZEN 5 3600	RYZEN 3 3200G
RYZEN 7 3800X	RYZEN 5 3600X	RYZEN 3 2300X



GPT4 VISION

ANÁLISIS COMPLETO



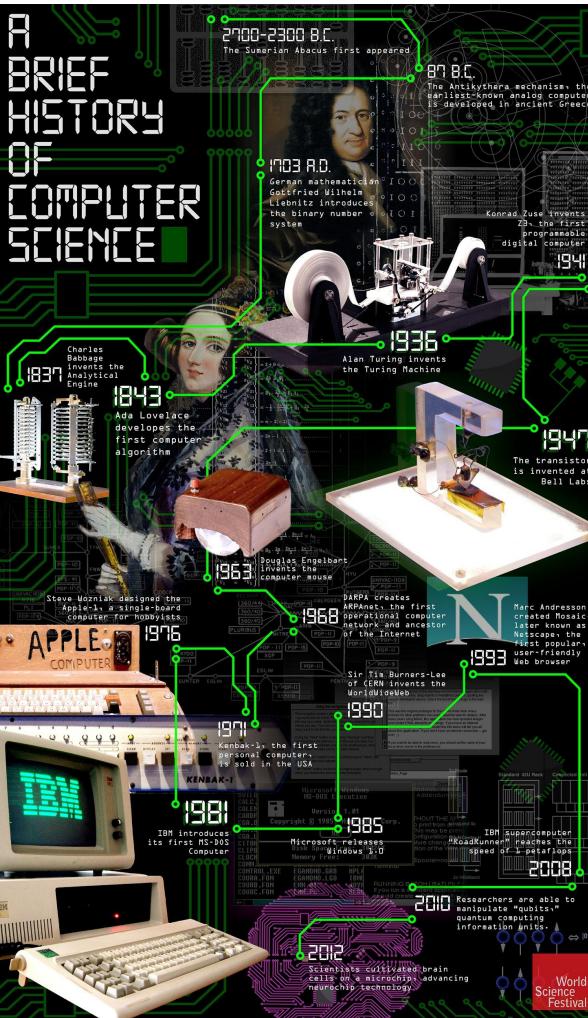


create an image that identifies generative artificial intelligences



❖ Sure, here is an image that identifies generative artificial intelligences:







Descargar ebook



En el transcurso de solo unas pocas décadas, las computadoras han revolucionado la investigación científica y cada vez más científicos están escribiendo programas de computadora para hacer su trabajo. A pesar del uso omnipresente de las computadoras en la ciencia, pocos investigadores de las ciencias naturales tienen estudios en ciencias de la computación, ingeniería de software o análisis numérico. Por lo general, adquieren sus conocimientos informáticos "en el trabajo" y, a menudo, se sienten abrumados por la cantidad de conocimientos informáticos que deben absorber.

La computación en la ciencia proporciona una base en computación para los científicos que utilizan métodos computacionales. El libro explica cómo se utiliza la informática en las ciencias naturales y proporciona una descripción general de alto nivel de los aspectos relevantes de la informática y la ingeniería de software con un enfoque en conceptos, resultados y aplicaciones. El objetivo de este libro es explicar estos principios básicos y mostrar cómo se relacionan con las tareas del trabajo diario de un científico en un lenguaje familiar para ellos. Su característica única es conectar los puntos entre la ciencia computacional, la teoría de la computación y la información y la ingeniería de software. Compensará la falta general de formación formal en informática y teoría de la información, lo que permitirá a los lectores comprender mejor cómo utilizan las computadoras en su trabajo y cómo funcionan las computadoras.

Esta nueva edición se ha actualizado y ampliado significativamente para reflejar los avances en la informática científica, incluidos nuevos ejemplos y referencias. También incluye un nuevo capítulo sobre reproducibilidad que refleja la importancia que juega la reproducibilidad computacional. Acompañado por un sitio web mantenido por el autor que alberga código complementario y material complementario, está destinado tanto a estudiantes graduados como a científicos experimentados. Es muy deseable cierta experiencia práctica con la informática, pero no se espera ninguna competencia en ninguna tecnología informática específica.

[Ocultar resumen completo](#)



Exportar cita y resumen

BibTeX

RIS

Comparte este libro



Para aprovechar este servicio, su institución debe tener acceso al contenido de este libro electrónico de IOP.

[Lee mas](#)

[Recomendar a su bibliotecario](#)

Contenido relacionado

ARTÍCULOS

PERIODÍSTICOS

[La plataforma de Internet VISPA para la divulgación, la educación y la investigación científica en varios experimentos](#)

[Investigación científica en las escuelas](#)

[China aumenta el gasto al llegar al cuarto lugar en el ranking mundial](#)

CAPÍTULOS DE LIBROS

[Computación en la ciencia: ¿Qué es la computación?](#)

[Computación en ciencia: formalización de la computación](#)

[Computación en la ciencia: Computación en la ciencia](#)



1



2



3



4

ANDEAN CALCULATORS

Despite the general conviction that the data recorded on *quipus* were calculated by means of *yupanas* (Mackey et al., 1990; Urton, 2002: 119; Sempat, 2002:124; Quilter, 2002: 215) (while doing reckoning, making and untangling knots would be really too difficult....), not only have these *yupanas* had hard being recognized as the main instruments of calculations in the Andean world, mostly outside Peru, but, actually they have often been presented as nothing less than fortress models even toys, too.

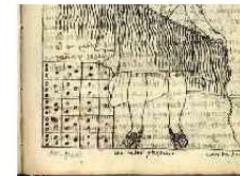
The acknowledgement of their cultural function is not yet clear at all because, typical though they are of the Late Horizon, they might also be considered previous or pan-Andean as some typologies of theirs have been found in different areas and contexts far from one another and without any particular gradients from Cuzco towards the outskirts.

So far *yupanas* haven't been rewarded the attribution they actually deserve because:

- firstly, unlike *quipus*, their different shapes evoke stylised and idealized models of architectural buildings instead of mathematical instruments;
- secondly, they have never been quoted in any chronicles apart from a dictionary by Gonzales Holguin where some sentences associate only the word *yupana*, not the object itself, to calculations: "Yupani. Reckoning / Yupan. Calculations / Yupani trispa. Summing up or making brief summaries / Yupana. Letter, the figures of numbers" (Gonzales Holguin, 1607:371);
- finally the passages where the Incas' counting system is described are quite ambiguous.

With regard to this matter, as on other occasions, Acosta has proved himself to be one of the most accurate writers of the seventeenth century. Here is what he wrote about *yupanas*: "...watching them in another type of *quipus*, where corn grains are used to count, is a fascinating experience, because these Indians can do the same accounts that a very expert accountant is to do by means of a pen, some ink and a lot of complicated operations to know, for example, the true amount of a tribute he is due. With the help of their grains, they put one seed here, three seeds there and eight seeds I don't know where and, after moving one of them from here and changing down three of them, they succeed in doing their accounts without the smallest error and so they can decide the exact amount that each person is to pay much better than we can do by means of our pens and ink. Whoever can judge whether these people are men or beasts. What I affirm with dead certainty is that these people excel us by far in what they work hard" (Acosta 1954:190).

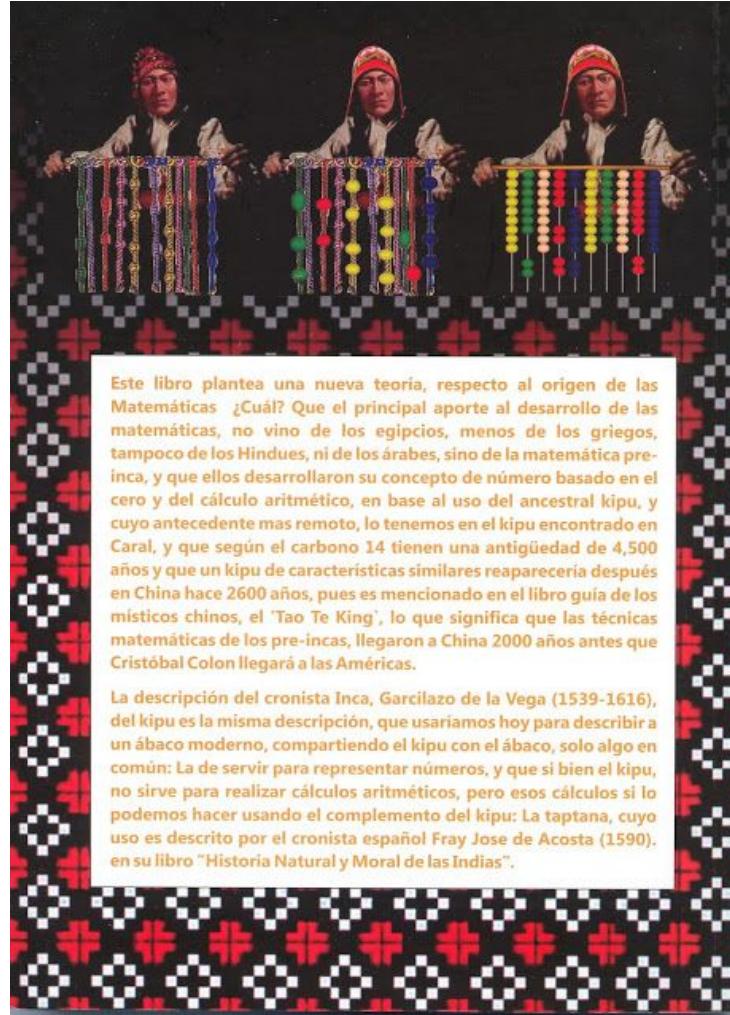
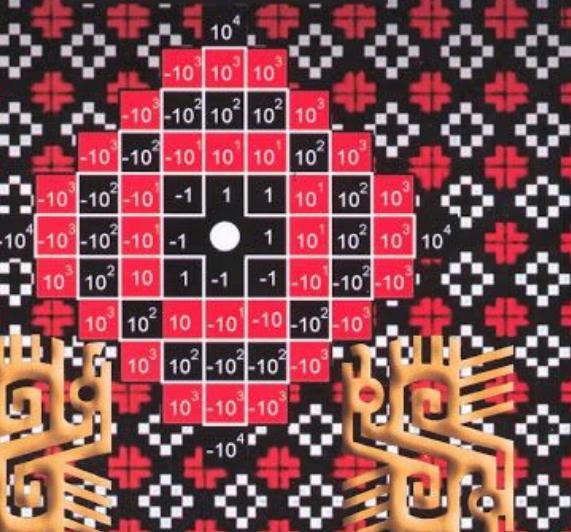
Since the thirties, after the public exhibition of a very famous drawing by Guaman Poma (1936:362 (360)) from the *Nueva Crónica y Buen Gobierno*, where a tablet with black and white small circles was drawn next to "a contador mayor y tesorero", numerous scientists have tried to explain how *yupana* worked. In fact, the tablet was soon considered to be the frame of a *yupana* and, moreover, the author himself wrote the words "cuentan en tablas" to comment his drawing and hinted at the use of the *yupana* as a calendar (Guaman Poma 1936:361-363 (359-363)).



SEMIOTIKA ARKETIPIKA DE LA MATEMATIKA INKA

¿Evolucionó el ábaco a partir del kipu?

Sergio Huanca Yruri



Este libro plantea una nueva teoría, respecto al origen de las Matemáticas ¿Cuál? Que el principal aporte al desarrollo de las matemáticas, no vino de los egipcios, menos de los griegos, tampoco de los Hindus, ni de los árabes, sino de la matemática pre-inka, y que ellos desarrollaron su concepto de número basado en el cero y del cálculo aritmético, en base al uso del ancestral kipu, y cuyo antecedente mas remoto, lo tenemos en el kipu encontrado en Caral, y que según el carbono 14 tienen una antigüedad de 4,500 años y que un kipu de características similares reaparecería después en China hace 2600 años, pues es mencionado en el libro guía de los místicos chinos, el 'Tao Te King', lo que significa que las técnicas matemáticas de los pre-incas, llegaron a China 2000 años antes que Cristóbal Colón llegaría a las Américas.

La descripción del cronista Inca, Garcilazo de la Vega (1539-1616), del kipu es la misma descripción, que usariamos hoy para describir a un ábaco moderno, compartiendo el kipu con el ábaco, solo algo en común: La de servir para representar números, y que si bien el kipu, no sirve para realizar cálculos aritméticos, pero esos cálculos si lo podemos hacer usando el complemento del kipu: La taptana, cuyo uso es descrito por el cronista español Fray Jose de Acosta (1590). en su libro "Historia Natural y Moral de las Indias".

¿Cómo resolver problemas?

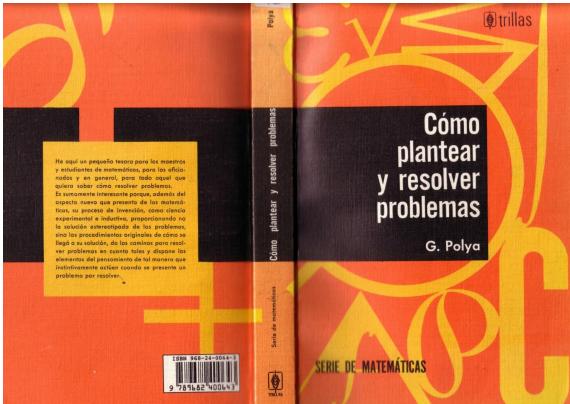
Sobre los problemas

Un problema es una situación en la cual se pretende llegar a una meta y, en función de lograrlo, se deben hallar y utilizar medios y estrategias. La mayoría de los problemas tienen una serie de elementos compartidos: un estado inicial, una meta a lograr, un conjunto de recursos, unas indicaciones en torno a lo que está permitido hacer y utilizar y lo que no, un dominio sobre el que se aplica y, por último experiencias de casos similares. En términos generales:

"un problema surge cuando existen obstáculos entre una situación dada y la situación a la que se quiere llegar, es querer encontrar un camino para poder llegar del estado actual al estado final, o al que se quiere obtener" (Torres, 2011:64).

Los problemas son situaciones que ubican a quien los resuelve ante la necesidad de desplegar su actividad cognitiva en un intento de búsqueda de estrategias, de elaboración de conjeturas y de toma de decisiones (Azcue, Diez, Lucaneray Scandrolí, 2006). Cuando se resuelven problemas desde una disciplina, se utilizan estrategias específicas de esta (por ejemplo, en problemas matemáticos se usan estrategias propias del área). Pero también se dispone de estrategias generales: una de ellas es la heurística, que se basa en el uso de reglas empíricas para llegar a una solución.

George Pólya fue un matemático destacado que en gran parte de su vida académica se dedicó a investigar en el área de resolución de problemas. Es considerado pionero en la temática por sus aportes, que se basan en una perspectiva global y para nada restringida a las matemáticas. Pólya enfocó la resolución de problemas desde un punto de vista que permitió plantear una serie de procedimientos que se aplican en la vida cotidiana:



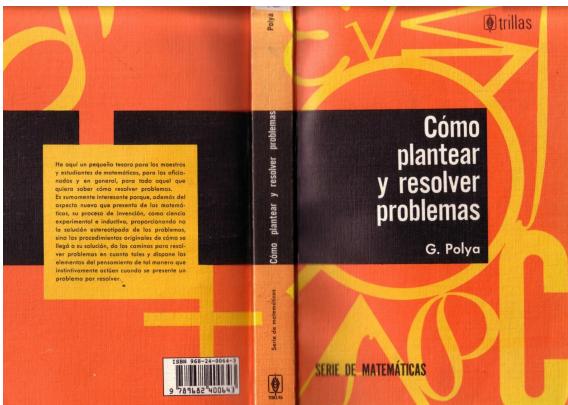
¿Cómo resolver problemas?

"Mi punto de vista es que la parte más importante de la forma de pensar que se desarrolla en matemática es la correcta actitud de la manera de acometer y tratar los problemas. Tenemos problemas en la vida diaria, en las ciencias, en la política, tenemos problemas por doquier. La actitud correcta en la forma de pensar puede ser ligeramente diferente de un dominio a otro, pero solo tenemos una cabeza y por lo tanto es natural que en definitiva haya solo un método de acometer toda clase de problemas. Mi opinión personal es que lo central en la enseñanza de la matemática es desarrollar tácticas en la resolución de problemas.". (Polya, 1969)

Para Polya (1945), la heurística es el área que trata de comprender el método que conduce a la solución de problemas y, en particular, se centra en las operaciones mentales útiles en este proceso. La heurística hace uso de la sistematización de la experiencia de resolver problemas a partir de cómo lo hacen los expertos.

Para ello se propone una serie de cuatro pasos:

- 1. Comprender el problema.** Reconocer qué se pregunta, identificar lo que hay que resolver y las condiciones asociadas
- 2. Elaborar un plan.** Se trata de establecer la vinculación entre los datos presentes y el problema a resolver, determinar los recursos que se utilizarán, verificar la similitud con otros problemas previamente resueltos y también la posibilidad de utilizar teorías o modelos útiles, todo esto en función de buscar una manera de resolver el problema.
- 3. Ejecutar el plan.** Desarrollar el resultado de la respuesta, a partir de ejecutar el plan, avanzando y verificando cada paso.
- 4. Revisar y verificar la solución.** Controlar qué hace y que dice el resultado, con vistas a considerar la posibilidad de transferir la solución a otros problemas.



¿Cómo resolver problemas?



> Comprender
el problema



> Concebir
un plan



> Ejecutar
el plan



> Examinar y
extender
la solución

¿Cómo resolver problemas?

Etapas del método Pólya de resolución de problemas

Cada una de las etapas anteriores se ejecuta en forma de secuencia. A cada etapa, para su mejor comprensión, se le asocian una serie de preguntas orientadoras:

1. Comprender el Problema.

Etapa en la que se identifica qué se pide de modo completamente independiente de las diversas condiciones impuestas y limitaciones asociadas al problema. En particular, se determina cuál es el objetivo de trabajo, los datos con que cuentan las condiciones, y la incógnita u objetivo de trabajo.

- ¿Es claro el enunciado?
- ¿Podés replantear el problema con tus palabras?
- ¿Cuál es la incógnita o el objetivo de trabajo?
- ¿Cuáles son los datos?
- ¿Cuál es la condición?
- ¿Hay suficiente información? ¿Está presentada de manera comprensible? ¿Es la condición suficiente para determinar la incógnita?, ¿es insuficiente?, ¿es redundante?, ¿es contradictoria?
- ¿Es un problema similar a algún otro que haya resuelto con anterioridad?

2. Elaborar un Plan.

Se centra en el cómo o, en otras palabras, qué estrategia se va a utilizar a fin de resolver el problema. Las estrategias pueden plantearse a partir de pruebas de ensayo y error o hasta desarrollar una táctica que permita llegar a la solución. Por ejemplo, idear un plan en matemáticas, en general, se traduce en usar una ley, una definición o un principio que permita obtener la respuesta del problema. La etapa de planificación debe relacionarse con problemas parecidos, del mismo tipo, y también con resultados útiles. Se debe determinar si se pueden usar problemas similares o sus resultados (problemas análogos).

- ¿Te has encontrado con un problema semejante?
- ¿Has visto el mismo problema planteado en forma ligeramente diferente?
- ¿Conocés algún problema relacionado?
- ¿Conocés algún teorema que te pueda ser útil?
- ¿Podrías enunciar el problema en otra forma?
- ¿Podrías plantearlo en forma diferente nuevamente?

¿Cómo resolver problemas?

3. Aplicar el plan.

Es la puesta en práctica de lo que el estudiante estableció en la etapa anterior. Se llevan adelante las etapas planteadas. Puede suceder que en el desarrollo se detecte que algo no es pertinente para la solución del problema, lo que implicará replantear la estrategia y volver a comenzar.

¿Se pueden ver claramente que los pasos son correctos?

¿Es posible demostrarlo?

4. Revisar y verificar la Solución.

Al final del trabajo, es necesario verificar el resultado obtenido y el proceso de solución desarrollado. Se realiza un abordaje metacognitivo.

¿Podés verificar el resultado?

¿Podés verificar el razonamiento?

¿Podés obtener el resultado en forma diferente?

¿Podés verlo de golpe?

¿Podés emplear el método en algún otro problema?

También a esta etapa se la conoce como visión retrospectiva. Para construir conocimiento que sirva en el futuro, es necesario observar y reflexionar sobre qué fue lo que se hizo. Así se obtiene un canal de retroalimentación valioso, que ayudará a resolver problemas futuros. Por otro lado, una vez obtenido el método que lleve a una solución correcta, también se procura ver si hay otros métodos alternativos que respondan al mismo problema. De esta manera, se obtienen distintas perspectivas que pueden aplicarse en el futuro.

¿Cómo resolver problemas?

Ejemplo de aplicación del método de Pólya:

1. Comprender la situación:

una persona compró [condición] medio kilogramo de carne [dato], tres cuartos de kilo de papas [dato] y un cuarto kilo de verduras [dato], ¿cuántos kilogramos transportó a su casa [incógnita]?

2. Elaborar un plan:

sumar lo que se compró y el resultado es la cantidad de kilogramos que la persona tuvo que transportar.

3. Ejecutar el plan:

$$(1 / 2) + (3 / 4) + (1 / 4) = 6 / 4 = 1,5 \text{ kg}$$

4. Revisar, verificar y comunicar:

La persona tuvo que transportar 1,5 kilogramos de comestibles hasta su casa.

¿Cómo resolver problemas?

Una ama de casa compró $\frac{1}{2}$ kg de carne, $\frac{3}{4}$ kg de papas y condición dato dato

$\frac{2}{3}$ kg de verduras. ¿Cuántos kilogramos tuvo que llevar?
dato incógnita

2) Conciba un plan

(Operación matemática - condición – incógnita)

Sumar lo que compró y el resultado es la cantidad de kilogramos que tuvo que llevar

3) Ejecute el plan

$$\frac{1}{2} + \frac{3}{4} + \frac{2}{3} = \frac{6+9+8}{12} = \frac{23}{12} = 1,91666 \dots = 1,92$$

4) Examine la solución obtenida

Tuvo que llevar 1,92 kilogramos

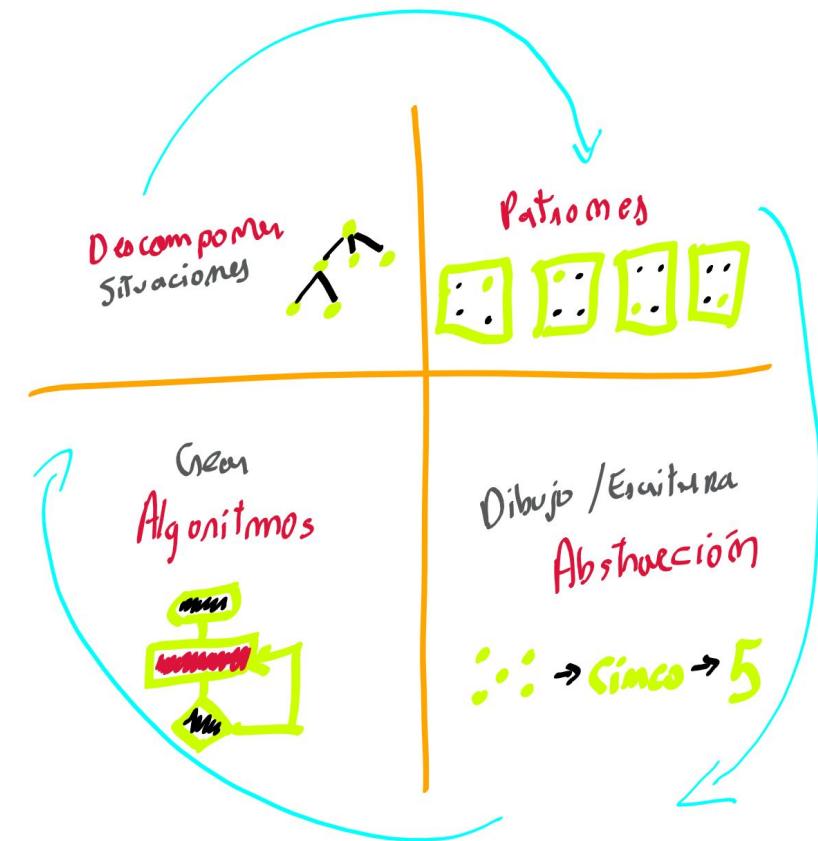
¿Qué es Pensamiento Computacional?

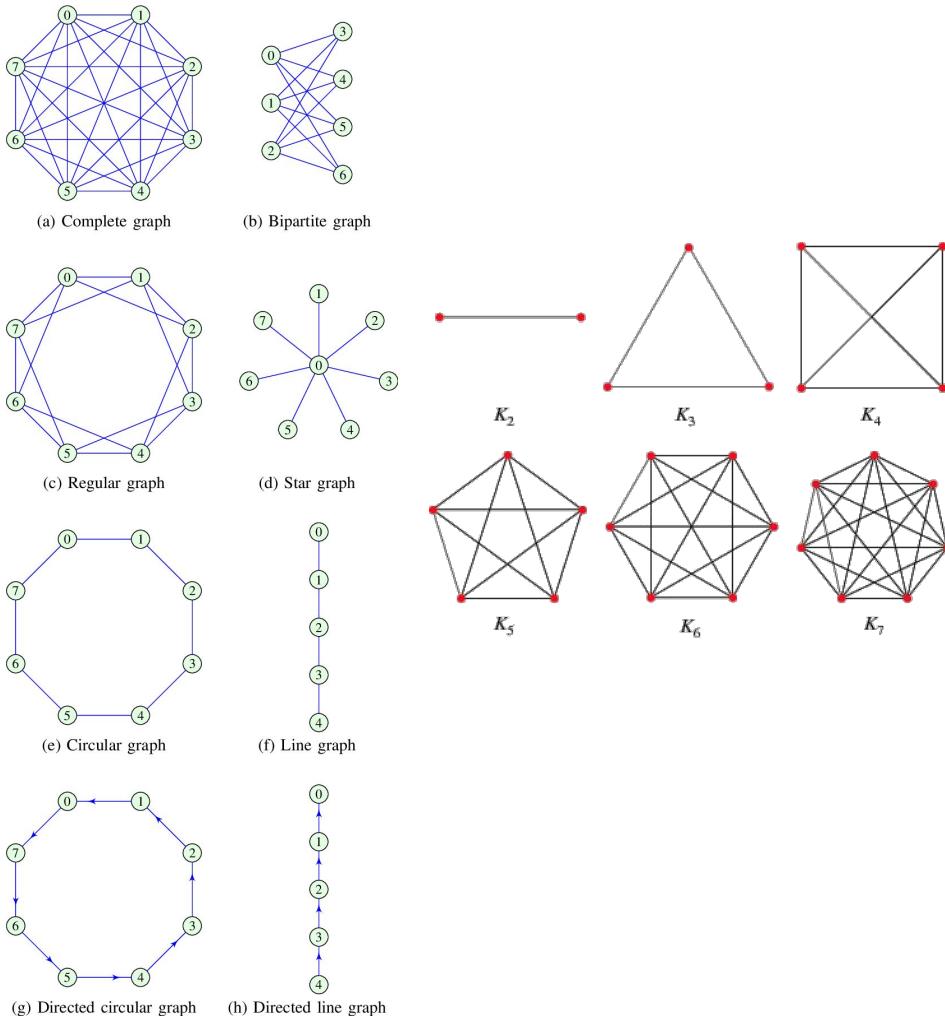
Se refiere a un **modo de procesamiento de información**, de un desarrollo cognitivo y creativo.

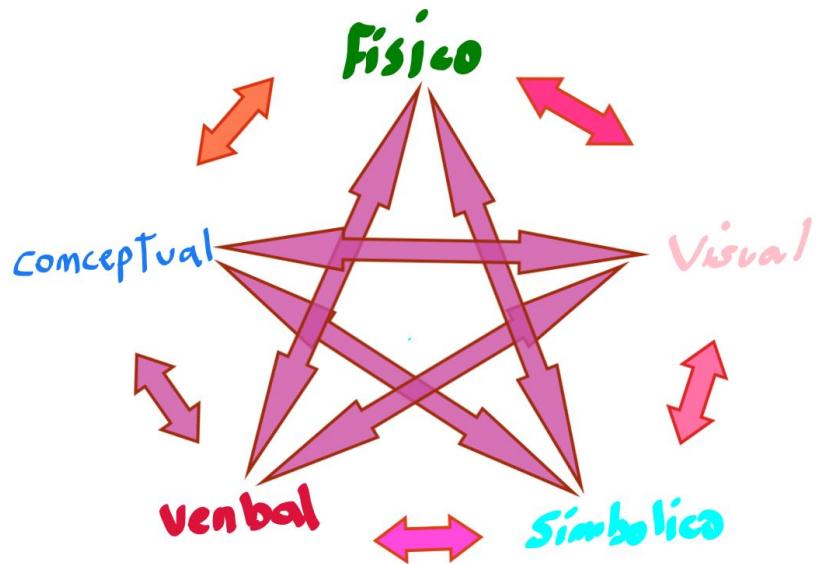


¿Qué es Pensamiento Computacional?

Es la capacidad de pensar en problemas que la computadora pueda resolver







¿Que es Pensamiento Computacional?

Grafo completo

Se transforma un problema a diferentes ámbitos

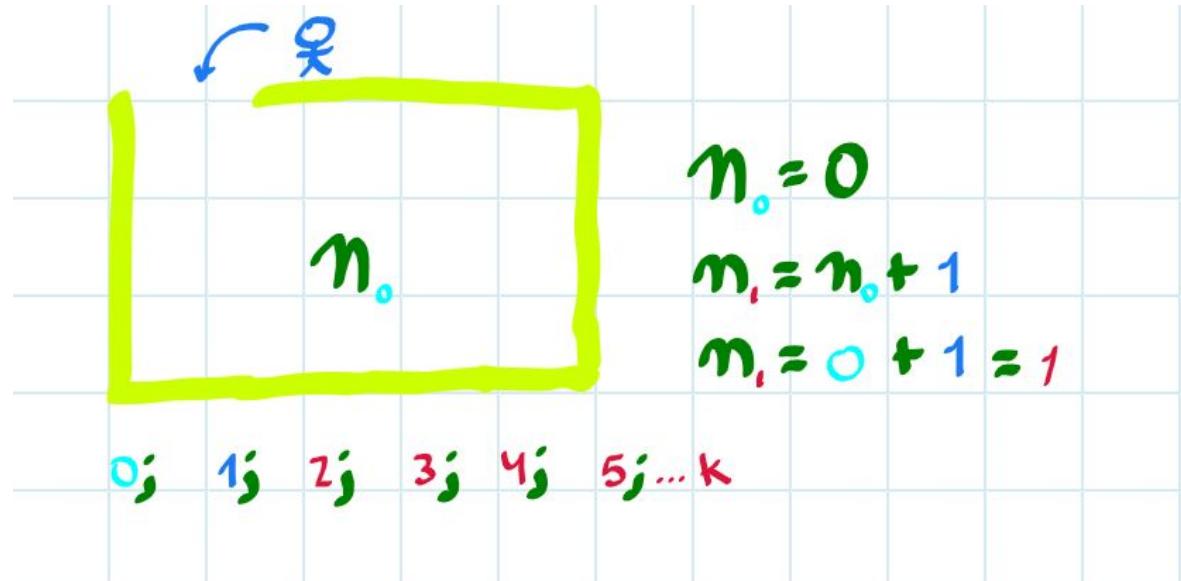
Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitación está vacía)



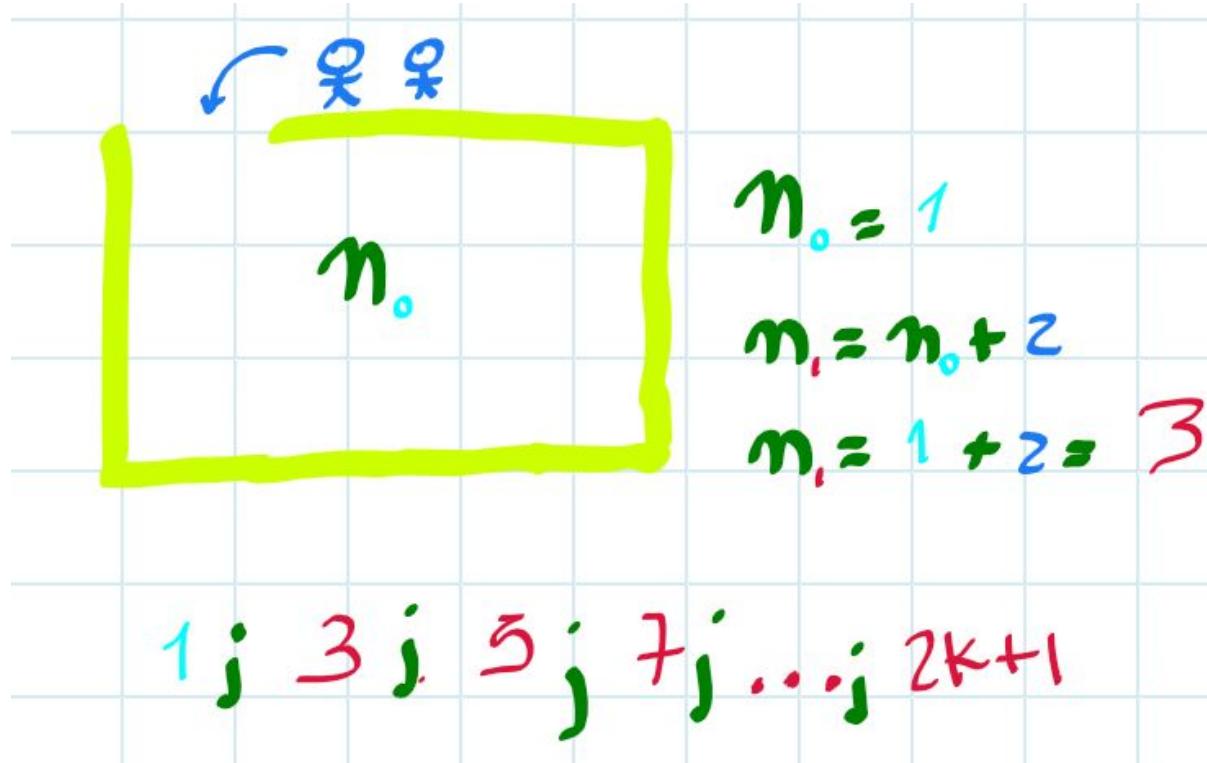
Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitación está vacía)



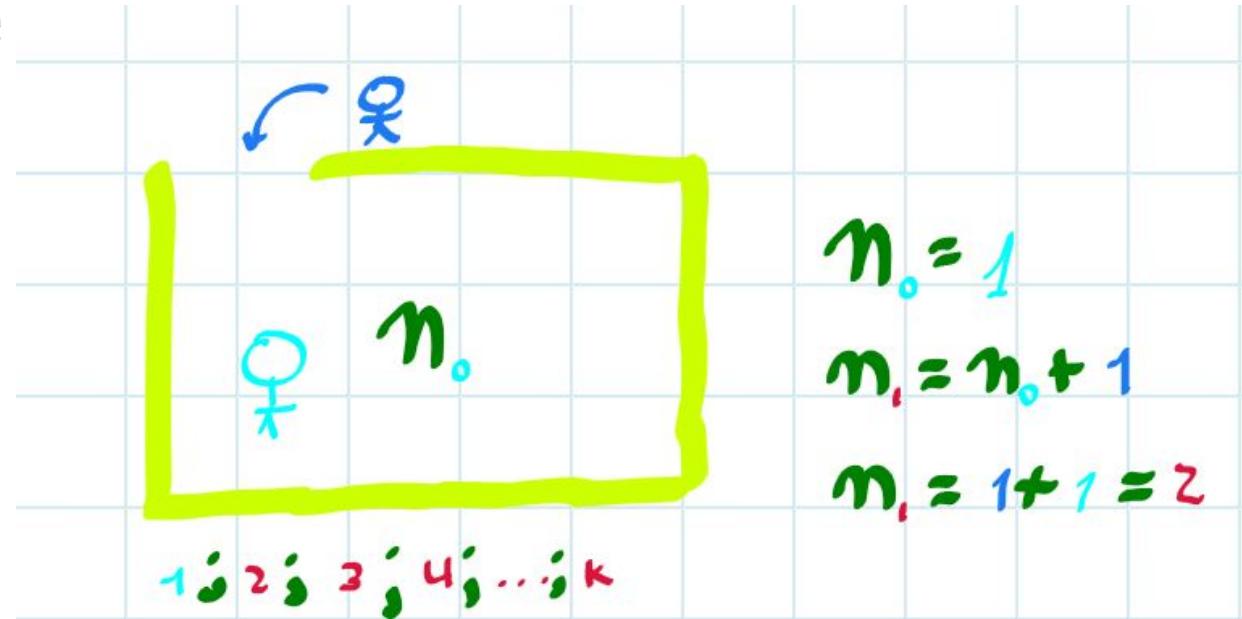
Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitación está vacía)



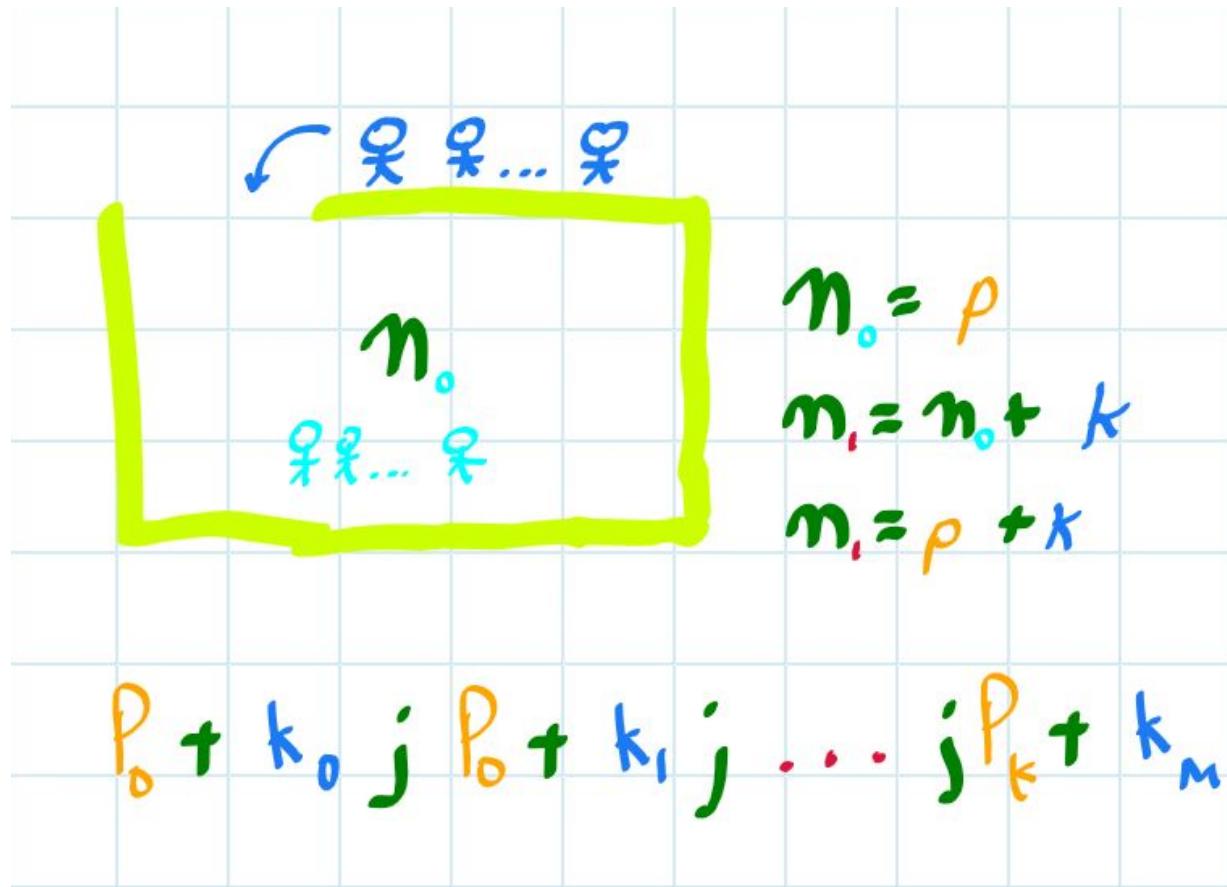
Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitación está vacía)



Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitación está vacía)



Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitación está vacía)

Algoritmo Contador

Cont0 <- 0

Cont1 <-0

A <-0

Escribir "Ingrese el número inicial de personas en la habitación"

Leer Cont0

Escribir "Ingrese el número de personas que están ingresando en la habitación"

Leer A

Cont1 <- **Cont0** + A

Escribir "El número de personas es:",Cont1

Fin Algoritmo

Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitaciones está vacía)

```
#include <iostream>

using namespace std;

int main()
{
    int Cont0=0,Cont1=0,c;

    cout<<"Ingrese el número inicial de personas en la habitación";
    cin>>a;

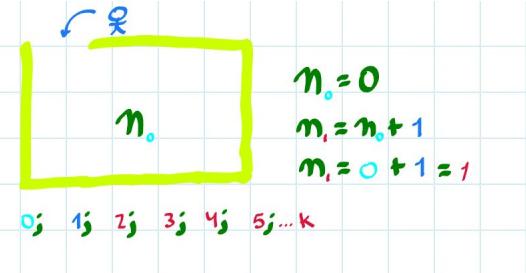
    cout<<"Ingrese el número de personas que están ingresando en la habitación";
    cin>>b;
    c=a+b;

    cout<<"La respuesta es"<<c;

    return 0;
}
```

Problema explicativo 01

Crear un contador, que a medida que entre alguna persona, cuente el número de personas que se encuentra en la habitación.(La habitación está vacía)



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

int main()
{
    int Cont0=0,Cont1=0,c;

    cout<<"Ingrese el número inicial de personas en la habitación";
    cin>>a;

    cout<<"Ingrese el número de personas que están ingresando en la
    habitación";
    cin>>b;
    c=a+b;

    cout<<"La respuesta es"<<c;
    return 0;
}
```



Algoritmo Contador

```
Cont0<- 0
Cont1<-0
A<-0
```

habitación"

```
Escribir "Ingrese el número inicial de personas en la habitación"
Leer Cont0
Escribir "Ingrese el número de personas que están ingresando en la
habitación"
Leer A
Cont1<-Cont0 + A
Escribir "El número de personas es: ",Cont1
```

Fin Algoritmo