



# Práctica 2: Diseño de algoritmos con diagramas de flujo

---

## 1. Objetivos

- Utilizar una metodología de análisis de problemas para el diseño de algoritmos.
- Representar operaciones algebraicas y lógicas como expresiones algorítmicas.
- Plantear algoritmos representados como diagramas de flujo.
- Desarrollar algoritmos que tengan entrada y salida de datos, condicionales y ciclos.

## 2. Marco teórico

Un algoritmo es un conjunto de pasos, que deben ser computables, finitos y precisos, para resolver un problema. A pesar de que no existe una receta para encontrar el algoritmo que soluciona un problema, resulta de gran ayuda seguir una metodología que permita estructurar la información que se tiene del problema y así identificar más fácilmente las estrategias de solución. A partir del método de Polya, podemos entonces plantear siempre, al menos las siguientes preguntas durante el proceso de diseño de un algoritmo:

- ¿Cuáles son los datos de entrada?
- ¿Cuáles son los datos de salida?
- ¿Qué variables auxiliares ayudan a construir el algoritmo?
- ¿Qué proceso algorítmico debe seguirse para solucionar el problema planteado o los subproblemas?

Seguir una metodología de diseño de algoritmos no garantiza, por sí solo, que se obtendrá el algoritmo buscado. Lo único que se busca es estructurar la información y las ideas para llegar más fácilmente al algoritmo que en todo caso dependerá en gran medida, de la creatividad, la habilidad y la experiencia de quien está afrontando la tarea.

## 3. Tareas a realizar

El trabajo a realizar en esta práctica consiste en el desarrollo de una serie de algoritmos para los problemas planteados. Analice cada problema y desarrolle el algoritmo solución con un diagrama de flujo. Utilice el software Raptor (<http://raptor.martincarlisle.com>) para probar el funcionamiento de sus algoritmos a la vez que para entenderlos mejor. Cada ejercicio está marcado de acuerdo a su nivel de dificultad relativa, donde \* quiere decir fácil y \*\*\* difícil.

1. [\*] Dado el valor del lado de un triángulo equilátero, haga un algoritmo que calcule su perímetro, su altura y su área.
2. [\*] Haga un algoritmo que convierta de pulgadas a centímetros.
3. [\*] Determinar el área y volumen de un cilindro, dada su altura y su radio.
4. [\*\*] Los empleados de una fábrica trabajan en dos turnos (de 8 horas), diurno y nocturno. Teniendo en cuenta las siguientes reglas para el pago del salario:
  - La tarifa de las horas diurnas es de 6 euros
  - La tarifa de las horas nocturnas es de 10 euros
  - En caso de ser domingo, la tarifa se incrementará en 200 euros el turno diurno y 300 euros el turno nocturno

Calcule el costo de la nómina mensual (costo total de salarios) de la fábrica con la siguiente distribución de trabajadores:

- Diurno en semana: 120 trabajadores
  - Diurno domingo: 20 trabajadores
  - Nocturno en semana: 95 trabajadores
  - Nocturno domingo: 5 trabajadores
5. [\*] Teniendo en cuenta que un carro necesita cambio de aceite cada 6000 Km de movimiento, haga un algoritmo que calcule cuántos cambios de aceite ha tenido un carro según el total de kilómetros que ha recorrido.

#### Test

A continuación se muestran algunos valores de test para verificar el algoritmo:

Kilómetros recorridos	Cambios de aceite
5500	0
18000	3
26500	4

**Ayuda:** Utilice el operador módulo "%" o la división entera "//"

6. [\*] Diseñe un algoritmo que lea un número entero y determine si es par o impar.
7. [\*] Haga un algoritmo que muestre en pantalla las soluciones de la ecuación  $ax^2+bx+c=0$ , dados valores para los coeficientes  $a$ ,  $b$  y  $c$ .
8. [\*\*] Una máquina dispensadora de productos requiere un algoritmo para calcular las devueltas en monedas. El objetivo del algoritmo es que dada una cantidad a devolver, se debe calcular la combinación que genere la mínima cantidad de monedas, utilizando denominaciones de \$1000, \$500, \$200, \$100 y \$50. Si es imposible lograr la cantidad exacta, el sistema deberá decir lo que resta para lograrla.

#### Test

A continuación se muestran algunos valores de test para verificar el algoritmo:

Monedas	38780	1543	850	167
\$1000	38	1	0	0
\$500	1	1	1	0
\$200	1	0	1	0
\$100	0	0	1	1
\$50	1	0	1	1
Resto	30	43	0	17

#### Ayuda

- Utilice división entera "//" y la operación módulo "%"

9. [\*\*] Se necesita un algoritmo para sumar los tiempos de una carrera por etapas. El algoritmo solo debe sumar dos tiempos que incluyen horas, minutos y segundos. El problema es que los tiempos llegan al algoritmo así: por ejemplo 74532, queriendo decir 7 horas, 45 minutos y 32 segundos. El algoritmo recibirá dos tiempos expresados de esa manera y deberá calcular su suma.
10. [\*\*\*] Haga un algoritmo que diga si un número ingresado es perfecto o no. Un número perfecto es un número que es igual a la suma de sus divisores propios positivos. De esta forma, 6 es un número perfecto porque sus divisores propios son 1, 2 y 3; y  $6 = 1 + 2 + 3$ .
11. [\*\*\*] Haga un algoritmo que muestre el siguiente patrón en la pantalla:

```
#
###
#####
#####
#####
```

El tamaño del patrón estará determinado por un número entero impar que ingrese el usuario. En el ejemplo mostrado, el tamaño de la figura es 9.

#### Ayuda

- Consulte cómo mostrar un carácter en la pantalla y utilice esa instrucción en uno o varios ciclos para mostrar todos los caracteres que requiera.
- Consulte cómo se puede "dar enter" con una instrucción en el algoritmo para poder cambiar de línea.

12. [\*\*\*] Haga un algoritmo que muestre el siguiente patrón en la pantalla:

```
      *           *
    ***         ***
  *****     *****
*****
```

El tamaño del patrón estará determinado por un número entero impar que ingrese el usuario. En el ejemplo mostrado, el tamaño de la figura es 7.

## 4. Evaluación

Envíe a través de la plataforma Moodle todos los archivos de Raptor desarrollados en un solo archivo comprimido. Los algoritmos enviados deberán sustentarse ante el profesor