

Introducción al lenguaje de programación C

Introducción

1. Escribir una función que reciba el capital inicial, una tasa de interés y un número de años y devuelva el monto final a cobrar. La fórmula a utilizar es:

$$C_n = C \times \left(1 + \frac{x}{100}\right)^n$$

Donde C es el capital inicial, x es la tasa de interés y n es el número de años a calcular.

2. Escribir una función que convierta un valor en grados Fahrenheit a grados Celsius. La fórmula que los relaciona es $F = \frac{9}{5}C + 32$.
3. Escribir una función que convierta un valor en grados Celsius a grados Fahrenheit.
4. Escribir una función que dadas la hora, minutos y segundos devuelva el tiempo en segundos.
5. Escribir una función que dadas las coordenadas de un vector en \mathbb{R}^3 , (x, y, z) devuelva la norma del mismo dada por $\sqrt{x^2 + y^2 + z^2}$.
6. Teniendo el radio de un cilindro y su altura escribir funciones que calculen:
 - a. El diámetro del cilindro.
 - b. La circunferencia.
 - c. El área de la base.
 - d. El volumen del cilindro.
 - e. El área del cilindro.

(Notar que algunas funciones pueden resolverse llamando a las ya programadas.)

7. Escribir una función que dado un número entero devuelva 1 si el mismo es impar y 0 si fuera par.
8. Escribir una función que dado un número devuelva el primer número múltiplo de 10 inferior a él. Por ejemplo, para 153 debe devolver 150.

9. Escribí una función que compare (reciba) 2 números enteros `a` y `b`. La función debe retornar un número menor, igual o mayor que cero si `a` es, respectivamente, menor que, igual a, o mayor que `b`.
10. Escribir una función que reciba un número entero y devuelva 1 si el número es positivo y -1 si el número es negativo.
11. Escribir una función que reciba un número entero y devuelva 1 si el número es positivo y 0 si el número es negativo.
12. Escribir un programa que imprima, en 3 líneas separadas, su nombre, apellido y padrón. Por ejemplo:

```
Nombre: Nicasio
Apellido: Maciel
Padron: 123456
```

Programas sencillos: Ciclos

Lectura recomendada: [Los secretos de printf\(\)](#).

13. Escribir un programa que imprima la tabla de multiplicar del número 7.
14. Escribir un programa que imprima la tabla de multiplicar para los números del 1 al 10.
15. Escribir un programa que imprima la tabla de conversión de Celsius a Fahrenheit para los valores de 0°C a 100°C de 10 en 10 con 2 decimales.

❗ Nota

Utilizar la función programada en el ejercicio 3

16. Escribir un programa que imprima por pantalla todas las fichas del dominó, sin repetir.
17. Escribir funciones que calculen:

a. $n! = 1 \cdot 2 \cdot \dots \cdot (n - 2) \cdot (n - 1) \cdot n$,

b. $e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} \dots$,

c. $e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} \dots$.

18. Los coeficientes binomiales $\binom{n}{k}$ se pueden computar como $\binom{n}{k} = \frac{n!}{k!(n-k)!}$, escribir una función que los calcule.

19. La potencia de un binomio se computa como

$$(x + y)^n = \sum_{k=0}^n \binom{n}{k} x^{n-k} y^k.$$
 Escribir una función

`void desarrollar_binomio(int n);` que imprima los términos a calcular. Por ejemplo, para `desarrollar_binomio(3);` la misma debe imprimir:

```
(x + y)^3 = + 1.x^3.y^0 + 3.x^2.y^1 + 3.x^1.y^2 + 1.x^0.y^3
```

20. Escribir una función que dado un número entero lo imprima espejado. Por ejemplo: 9511 se imprime como 1159. *Sugerencia:* utilice los operadores módulo `%` y división `/`.

Programas sencillos: Decisiones

21. Escribir una función `es_primo` que dado un número entero devuelva `true` si el mismo es primo y `false` en caso contrario.

22. Escribir una función que dados dos números `x` e `y` imprima:

- `x es mayor que y`, si `x` es mayor que `y`,
- `x y y son iguales`, si los números son iguales.

23. Escribir una función que dado un número entero devuelva su valor absoluto.

24. Escribir una función que dada una dimensión imprima la matriz identidad de esa dimensión.

25. Escribir un programa que calcule las 2 raíces de un polinomio de segundo grado (de la forma $ax^2 + bx + c = d$). Validar las operaciones a realizar (no dividir por cero ni calcular la raíz de un número negativo, o imprimir la raíz compleja).

26. Escribir una función que dadas dos rectas definidas por su pendiente y su ordenada al origen devuelva la abscisa en la que se intersectan. Validar lo que considere necesario.

27. Escribir funciones que resuelvan los siguientes problemas:

- a. Dado un año indicar si es bisiesto.

! Nota

Un año es bisiesto si es un número divisible por 4, pero no si es divisible por 100, excepto que también sea divisible por 400.

- b. Dado un mes y un año, devolver la cantidad de días correspondientes.
- c. Dada una fecha (día, mes, año), indicar si es válida o no.
- d. Dada una fecha, indicar los días que faltan hasta fin de mes.
- e. Dada una fecha, indicar los días que faltan hasta fin de año.
- f. Dada una fecha, indicar la cantidad de días transcurridos en ese año hasta esa fecha.
- g. Dadas dos fechas (día1, mes1, año1, día2, mes2, año2), indicar el tiempo transcurrido entre ambas, en años, meses y días.

❗ Nota

En todos los casos, invocar las funciones escritas previamente cuando sea posible.

28. Escribir un programa que calcule la mínima cantidad de billetes (moneda peso argentino) con los que se puede obtener un monto.

Ejemplo: `2320 ARS = (2x1000 ARS + 3x100 ARS + 1x20 ARS)`.

29. Escribir una función que implemente el algoritmo de Euclides para calcular el máximo común divisor de dos números n y m , dado por los siguientes pasos:

1. Teniendo n y m , se obtiene r , el resto de la división entera de m/n .
2. Si r es cero, n es el MCD de los valores iniciales.
3. Se reemplaza $m \leftarrow n$, $n \leftarrow r$, y se vuelve al primer paso.

30. (Decisión múltiple) Escribir una función que dado un carácter imprima:

```
si el carácter es 'A':
    "House Arryn"
si el carácter es 'B':
    "House Baratheon"
si el carácter es 'F':
    "House Frey"
si el carácter es 'G':
    "House Greyjoy"
si el carácter es 'L':
    "House Lannister"
si el carácter es 'M':
    "House Martell"
si el carácter es 'S':
    "House Stark"
si el carácter es 'T':
    "House Targaryen"
si es otro carácter:
    "No lo conozco"
```

Entendiendo C

31. Dados los siguientes tipos

- *char*
 - *unsigned char*
 - *short*
 - *int*
 - *unsigned int*
 - *unsigned long*
 - *float*
 - *double*
- a. ¿Cuál es el tamaño, en bytes, de cada uno y cuáles son sus valores máximos y mínimos?
- b. ¿Cuáles, si alguno, pueden tener un tamaño diferente? ¿De qué depende el cambio?

32. Cuantificar la cantidad de memoria requerida para los siguientes problemas

! Nota

Los vectores y las matrices ocupan tanto como la suma de sus elementos

- a. Un sistema mide un valor de temperatura *float* por segundo. ¿Cuánta memoria se genera por hora de registros?
- b. Una pantalla 1080p tiene una resolución de 1920x1080 pixeles. ¿Cuánta memoria hace falta para generar una imagen si...?
- cada pixel es monocromo, es decir, ocupa 1 bit.
 - se utilizan 256 colores, es decir, 1 byte por pixel.
 - se utilizan 24 bits de color (SVGA).
- c. La cámara de fotos de un iPhone 11 tiene 12 megapíxeles. ¿Cuánta memoria ocupa una foto de 24 bits de profundidad de color?
- d. El audio digital se suele almacenar en 16 bits a distintas frecuencias de muestreo. ¿Cuánto ocupa un minuto de audio mono muestreado a...?
- 44.1 KHz (CD).
 - 192 KHz
 - un CD pero estéreo.

33. Implementá las siguientes funciones.

- `long sumar(int a, int b)`,
- `long restar(int a, int b)`,
- `float multiplicar(int multiplicando, int multiplicador)`,
- `double dividir(int dividendo, int divisor)`, y
- `mod()` : retorna el resto de una división.

- a. ¿por qué crees que se eligieron esos valores de retorno?
- b. Si ahora quisiéramos soportar operaciones en punto flotante, además de las operaciones con enteros ¿qué debería cambiar?

34. ¿Cuál es la diferencia entre los literales `6`, `'6'`, y `"666"`? Defina 3 constantes distintas para cada uno de ellos.

35. Definir las siguientes constantes:

- `M_PI` con el valor `3.14159265358979`,
- `STR_HOLA_MUNDO` con el valor `"Hola mundo"`,
- `NOTA_MAX` con el valor `10`,
- `NOTA_MIN_APROBADO` con el valor `4`, y
- `SEGxHORA` con el valor `3600`.

36. ¿Qué diferencia hay entre los operadores `=` e `==`?

37. ¿Es válida la sentencia: `M_PI = 3.14159;`, con `M_PI` definida como en el ejercicio 30?.

38. Siendo `NOTA_MAX` una constante definida y `nota` una variable

- a. ¿Es válida la sentencia `NOTA_MAX == nota;`?
- b. ¿Qué valores puede tomar esa sentencia?

39. Escribir un programa que muestre por pantalla el valor de tres variables de tipo entero, real y carácter que hayan sido definidas pero no inicializadas. Compilar y ejecutar el programa. Analizar la salida que se produce.

40. Corregir los errores en cada una de las siguientes sentencias. *Nota:* puede haber más de un error en cada una:

a. `printf ("El producto de %d y %d es %d"\n, x, y);`

b. `a + b = suma;`

c. `*/ Programa para obtener el máximo */`

d. `if (numero => maximo)
 maximo == numero;`

e.

```
printf ("El resto de dividir %d por %d es\n",
      numerador,
      denominador,
      numerador % denominador);
```

f.

```
if (x = y)
    print ("%d es igual a %d\n", x, y);
```

g.

```
print ("La suma da %d\n," x + y);
```

41. (Todo es memoria 1)

- Escribir un programa que imprima el carácter `'A'`.
- Escribir un programa que imprima, con valor decimal, los caracteres `'0'` y `'j'`. *Nota:* utilice `printf()`.

42. (Todo es memoria 2) Respóndase las siguientes preguntas, y luego verifíquelas escribiendo un programa que se las responda:

- ¿Qué diferencia hay entre `0` y `'0'`? ¿Es válida la operación `'6' - '0'`?
- ¿Es válida la operación `'X' - 'A'`? ¿Qué obtiene al hacerla? ¿A qué tipo de variable corresponde?

43. Escribir un programa que imprima por pantalla los valores de las constantes estándar `INT_MIN`, `INT_MAX`, `RAND_MAX` y `UINT_MAX`. ¿En qué encabezados se encuentran estas constantes?

44. Escribí una función que imprima tu nombre, apellido y padrón. La misma no recibe argumentos.

```
Nombre: Nicasio
Apellido: Maciel
Padron: 123456
```

- Utilice 3 llamadas a `puts()`,
- Utilice 1 llamada a `puts()`,
- Utilice 3 llamadas a `printf()`.

45. (`printf()`) Dado un número flotante imprimirlo:

- Con 2 decimales,
- Con 3 decimales.

- c. Con signo, sea positivo o negativo.
 - d. Con 4 decimales, en ancho fijo de 10 caracteres, con signo y alineado a derecha.
46. En C, se *define* que el valor entero 0 (cero) representa el valor de verdad FALSO, mientras que un valor distinto de 0 representa el valor de verdad VERDADERO. A partir del estándar ISO C99 se encuentra disponible el tipo `bool`, incluyendo el encabezado `stdbool.h`. Sin embargo, éste no se incluye en código ANSI C compatible (el estándar anterior).
- a. Escriba la sentencia para definir un tipo enumerativo que cumpla con la definición del estándar y utilice los símbolos VERDADERO (o TRUE) y FALSO (o FALSE),
 - b. Escriba la sentencia para definir una variable del tipo definido,
 - c. Escriba una porción de código donde se imprima por pantalla "VERDADERO" o "FALSO" según el valor de la variable declarada.
 - d. ¿Cómo convertiría de este enumerativo al tipo `bool`?
47. Defina el tipo enumerativo `status_t` con los símbolos `ST_OK`, `ST_ERR_VALOR_NEGATIVO` y `ST_ERR_VALOR_NULO`. Escriba un programa que le pida al usuario un número positivo, valide la entrada e imprima por pantalla "Gracias, vuelvan pronto." si el usuario ingresa un valor apropiado o, en caso de error, imprima un mensaje acorde.
48. Defina el tipo `byte_t` a partir de `unsigned char`.
- a. Si aún no lo sabe, respóndase: ¿Qué es un byte?
49. El estándar C99 define varios tipos de enteros ¹ distintos en el encabezado `stdint.h` (e `inttypes.h`).
- a. Busque qué tipos se encuentran definidos en dicho encabezado y lístelos.
 - b. ¿Qué utilidad cree que pueden tener dichos tipos?

[1] : además de los enteros, se definen otros tipos en el encabezado `stdint.h`.

Interactuando con el usuario

50. Escribir un programa que calcule el inverso y la raíz de un número ingresado por teclado. Realice las validaciones necesarias. *Nota:* puede elegir NO calcular la raíz de números negativos o mostrar un valor complejo.

51. Escribir un programa que dado un día del año (1 a 366) ingresado por el usuario, imprima el día de la semana que le corresponde. Debe asumir que el año comenzó, por ejemplo, un domingo. Por ejemplo: si se ingresa '5', imprime 'jueves', si se ingresa '10' imprime 'martes', si se ingresa '294' imprime 'sabado'.
52. Escribir un programa que convierta una cantidad de segundos ingresada por teclado a horas, minutos y segundos. Realice las validaciones necesarias.
53. Escribir un programa que cuente la cantidad de líneas ingresadas por `stdin` hasta el ingreso de EOF. Al finalizar, muestre el resultado por pantalla.

! Nota

si el archivo del código fuente del programa se llama g01e44.c, ejecute el programa del siguiente modo: `cat g01e44.c | ./cuenta_lineas`

54. Escribir un programa que cuente la cantidad de espacios, tabuladores, caracteres de nueva línea y letras ingresados por `stdin` hasta el ingreso de EOF. Al finalizar, muestre el resultado por pantalla discriminando cada cuenta.

! Nota

si el archivo del código fuente del programa se llama g01e45.c, ejecute el programa del siguiente modo: `cat g01e45.c | ./cuentas`

55. Escribir un programa que pida números al usuario en tanto no se dispare la marca de EOF (en las terminales de GNU/Linux se dispara con Control+D). Al recibir la finalización, imprima los valores máximo, mínimo, promedio del intervalo. Escriba 3 versiones del mismo: una con un ciclo do-while, otra con un ciclo while y una tercera con un ciclo for. *Nota:* no es necesario almacenar todos los valores ingresados por el usuario.
56. Escribir un programa que lea de `stdin` una cantidad indefinida de calificaciones, hasta que se ingrese una calificación inválida. Una calificación inválida es aquella que se encuentra fuera del rango válido (por ejemplo, [0,100]). Al finalizar, debe imprimir un mensaje por `stdout` que indique el porcentaje de alumnos aprobados y los siguientes mensajes:
 - a. `Esto es raro` si el porcentaje es 100%,
 - b. `Muy bien` si el porcentaje es mayor al 80%,
 - c. `No está mal` si el porcentaje se encuentra entre 60% y 80%,

- d. Algo no está bien si el porcentaje está entre 40% y 60%, y
- e. Algo está mal si el porcentaje es menor.

57. Modificar el ejercicio 47 de forma tal que:

- a. la cantidad de calificaciones sea definida y la ingrese el usuario, y
- b. si se ingresa una calificación inválida, sea ignorada.

58. Escribir un programa que pida al usuario los extremos de un intervalo y luego lea una cantidad indefinida de números. Finalizar cuando se ingrese el número 0, sin importar si está contenido en el intervalo. Al finalizar imprimir por pantalla:

- a. La cantidad de números ingresados.
- b. La cantidad de números ingresados que pertenecen al intervalo, en valor absoluto, relativo y porcentaje.
- c. Los valores mínimos y máximos ingresados.

59. **(Rendimiento del auto)** Los automovilistas se encuentran preocupados por el consumo y rendimiento de su vehículo. Algunos de ellos llevan un registro con varias cargas de combustible, el costo del mismo y la cantidad de kilómetros recorridos con esa cantidad.

Escriba un programa que permita al usuario ingresar dicha información y muestre el rendimiento (en litros / 100 km) del vehículo cada vez. Al finalizar, imprima el rendimiento promedio, el mínimo, el máximo, el total de kilómetros recorridos y el dinero utilizado.

Lista 1 Ejemplo de ejecución

```
Ingrese los kilometros recorridos (-1 para terminar): 486
Ingrese el precio del combustible por litro: 21.38
Ingrese el monto total utilizado en este compra: 930.24
Rendimiento [l/100km] de este tanque >> 8.95

Ingrese los kilometros recorridos (-1 para terminar): 606
Ingrese el precio del combustible por litro: 23.57
Ingrese el monto total utilizado en este compra: 980.04
Rendimiento [l/100km] de este tanque >> 6.86

Ingrese los kilometros recorridos (-1 para terminar): 594
Ingrese el precio del combustible por litro: 24.89
Ingrese el monto total utilizado en este compra: 900.16
Rendimiento [l/100km] de este tanque >> 6.09

Ingrese los kilometros recorridos (-1 para terminar): -1

Rendimiento promedio [l/100km] >> 7.1919
Mejor rendimiento [l/100km] >> 6.09
Peor rendimiento [l/100km] >> 8.95
Distancia total recorrida [km] >> 1686
Combustible total consumidos [l] >> 121.25
Costo total de su vehiculo [$] >> 2810.44
```

📌 Nota

Según el algoritmo con el que se calcule el rendimiento promedio variará el resultado, por ejemplo, otro valor posible es: 7.3008.