
Taller 2: Estructuras de control

Haga diagramas de flujo para las siguientes situaciones, tenga presente que el programa debe pedir al usuario información precisa y el resultado o salida del programa debe ser lo mas explícita posible:

1. El factorial de un número natural.
2. Escriba un programa que pregunte una y otra vez si **desea continuar el programa**, salvo si se contesta exactamente **no** o **No** o **NO**.
3. La suma de los naturales hasta cierto número.
4. Escriba un programa que simule una alcancía. El programa solicitará primero una cantidad, que será la cantidad final de dinero que queremos ahorrar. A continuación, el programa solicitará una y otra vez las cantidades que se irán ahorrando, hasta que el total ahorrado iguale o supere al objetivo. El programa solo acepta cantidades positivas.
5. Cuando un usuario ingrese un número natural le calcule la suma de los números pares hasta ese número y los impares hasta ese número.
6. Todos los divisores de un número natural.
7. Tablas de multiplicar.
 - (a) Cuando un usuario ingrese un número natural entre 1 y 10 le arroje la tabla de multiplicar correspondiente.
 - (b) Todas las tablas de multiplicar (las tablas de multiplicar del 1 al 10).
8. El máximo común divisor de dos números enteros.
9. Una tienda tiene un descuento del 60% en todos sus productos, si el administrador del lugar desea programar todos los computadores de la tienda para que los usuarios puedan ingresar el precio de cada producto y el programa inmediatamente arroje el precio inicial, el valor del descuento aplicado y el precio final que deberá pagar. Todo esto después de haber ingresado la cantidad de productos que desea llevar. Por último, el usuario deberá ver en pantalla el valor total sin descuento de su compra, junto con el valor total descontado y el valor total final de su compra.

10. El promedio de una cantidad de números todos distintos de cero dados por un usuario. El usuario ingresará 0 para indicar que no se ingresarán más valores.
11. Dada la siguiente sucesión de números $-3, -1, 1, 3, 5, 7, 9 \dots$
- Hallar el término n -ésimo (a_n) de la sucesión.
 - El programa muestra la cantidad de términos de la sucesión que el usuario desea ver.
 - Determine el cociente entre los dos últimos números de la sucesión $\left(\frac{a_n}{a_{n-1}}\right)$ y determine si este cociente se acerca a algún número cuando la cantidad de términos de la sucesión es grande.
12. * Dada la siguiente sucesión de números $1, 1, 2, 3, 5, 8, 13 \dots$
- Hallar el término n -ésimo (a_n) de la sucesión.
 - El programa muestra la cantidad de términos de la sucesión que el usuario desea ver.
 - Determine el cociente entre los dos últimos números de la sucesión $\left(\frac{a_n}{a_{n-1}}\right)$ y determine si este cociente se acerca a algún número cuando la cantidad de términos de la sucesión es grande.
13. A través de la historia, grandes mentes han desarrollado diferentes esquemas de aproximaciones para el cálculo del número π . Consideraremos dos esquemas, uno dado por Leibniz (1646 - 1716), y uno dado por Euler (1707 - 1783). El esquema dado por Leibniz puede ser escrito como:

$$\pi \approx \sum_{k=0}^N \frac{8}{(4k+1)(4k+3)}.$$

Mientras que el esquema de Euler es:

$$\pi \approx \sqrt{\sum_{k=1}^N \frac{6}{k^2}}.$$

Si solamente los primeros N términos de cada suma son usados como aproximación de π , cada esquema aproximarán π con algún error.

Escribir un programa que tome a N como entrada y arroje el valor aproximado de Pi y el error con ambos esquemas. Por último, concluya cuál es el mejor esquema de aproximación.

- (a) * Encuentre el valor de N para el cual el error es menor a 0.000000001 en ambos esquemas.

*“Si estás trabajando en algo que te importa de verdad,
nadie tiene que empujarte: tu visión te empuja ”*
Steve Jobs.