

Trabajo Práctico Nivel 2

Descomposición de problemas - Modularización (funciones)

Fecha de Inicio 1/09/2025 - Fecha de Fin 8/09/2025

Ejercicio 1: Copiar el siguiente programa en el EID de Zinjay

```
1)      #include <stdio.h>
2)      #include<stdlib.h>
3)      int Cuenta_dig_pares(int);
4)      int main() {
5)          int NUM, CDIG;
6)          printf("\nIngrese un numero entero: ");
7)          scanf("%d",&NUM);
8)          NUM= abs(NUM);
9)          printf("\nMostrando rdo. de la función directamente: %d", Cuenta_dig_pares(NUM));
10)         printf("\n====> El mismo valor de otra forma <=====");
11)         CDIG= Cuenta_dig_pares(NUM);
12)         printf("\nCantidad de digitos pares: %d", CDIG);
13)         printf("\nFIN DEL PROGRAMA");
14)         return 0;
15)     }
16)     int Cuenta_dig_pares(int XNUM){
17)         int CD;
18)         CD=0;
19)         while(XNUM!=0){
20)             if((XNUM%2)==0) CD= CD+1;
21)             XNUM= XNUM/10;
22)         }
23)         return CD;
24)     }
```

Ejecutar el programa paso a paso con los siguientes valores NUM= -9781 y con NUM= 9713; revisar la salida de la ejecución. ¿Se podría realizar alguna mejora? Luego eliminar la línea 2) y volver a compilar, ¿nota algún cambio?

Recordar que se deben incluir las librerías que permiten realizar el cálculo solicitado en los siguientes ejercicios.

Ejercicio 2: Dados un número real A, se solicita crear un menú que permita calcular y mostrar: a)
El Seno de A.
b) El logaritmo natural de A.
c) El valor absoluto de A.

Ejercicio 3: Dados un número real A, se solicita mostrar la cantidad de cifras que posee su parte entera y la cantidad que posee su parte decimal.

Nota: La fórmula " $CD = \log_{10} A$ " puede indicar la cantidad de cifras que tiene, un número natural, ¿Cómo se podría convertir la parte decimal pura del número real A en un número natural?

Nota: recordar que en C la librería math.h tiene la función log() que representa al logaritmo natural.

Ejercicio 4: Dados dos naturales, X e Y, donde $X < Y$, se necesita generar aleatoriamente dos números reales A y B, pertenecientes a rango $[X, Y]$ y a continuación mostrar un menú que permita calcular y mostrar:

- a) La raíz cuadrada de A.
- b) La potencia de A^B .
- c) El logaritmo decimal de B.

Ejercicio 5: Dados dos números naturales a y b y dado un número natural N, mostrar N números naturales X generados aleatoriamente en el intervalo $[a, b]$ y que sean múltiplo de K.

Realizar un programa modular para los siguientes ejercicios (considerar la posibilidad de reutilizar módulos):

Ejercicio 6: Dados dos números naturales A y B, se desea calcular el producto de $A * B$ utilizando la definición del producto:

$$A * B = \underbrace{A + A + A + \dots + A}_{B \text{ veces}}$$

Nota: Crear una función que reciba A y B y que devuelva el resultado de $A*B$

Ejercicio 7: Escribir la cabecera e implementar módulos que realicen las siguientes operaciones:

- a) Calcular la cantidad de divisores naturales de un número natural X.
- b) Determinar si un número natural X es primo.
- c) Comparar dos números reales A y B, devolviendo 1 si $A > B$; 0 si $A = B$ o -1 si $A < B$.
- d) Invertir un número entero X.

Probar los módulos mediante un programa que presente un menú con las cuatro operaciones implementadas.

Ejercicio 8: Se necesita un programa en C, que simule el tiempo que esperan dos clientes, A y B, para realizar un trámite en el banco. El programa deberá mostrar cuál es el cliente que será atendido primero, considerando el menor tiempo de espera. (El tiempo de espera de cada cliente se asigna de forma aleatoria, siendo un número natural en el rango $[1, 30]$)

Nota: Si el rango es ingresado por el usuario, ¿cómo se modificaría la solución original?

Ejercicio 9: Dado un número natural X, determinar la cantidad de divisores naturales que posee. Si la cantidad de divisores es un número primo, mostrar el número equivalente de X en base 2, caso contrario, mostrar su equivalente a base 9.

Ejercicio 10: Dado un monto de dinero correspondiente a una deuda en pesos, la fecha de vencimiento de dicha deuda, y una tasa de interés diario, generar al azar una fecha posterior de pago y mostrar el monto a pagar en concepto de intereses.

Ejercicio 11: Dados una cantidad no determinada de números naturales, mostrar para cada uno de ellos si se trata de un número Deficiente, Abundante o Perfecto

Nota: Un número natural X se dice que es deficiente si la suma de sus divisores naturales (excepto el mismo número) es menor a X. Si la suma de sus divisores es mayor a X, se dice que el número es abundante y si es igual a X se dice que es perfecto.

Ejercicio 12: Dado un número natural N, generar al azar N números naturales X, de a lo sumo 6 cifras y determinar, para cada número X, si es capicúa, en caso de no serlo, generar un nuevo valor Y, que será igual al producto de todos los dígitos de X.

Ejercicio 13: Realizar un programa que genere al azar una lista de N números naturales aleatorios, en el rango [A, B] (con A y B naturales) y muestre aquellos números aleatorios que cumplan con no ser deficientes.

Nota: reutilizar la función generada en el ejercicio 11.

Ejercicio 14: Dadas N fechas ingresadas por el usuario, para cada una de ellas generar al azar una fecha mayor a ésta y luego mostrar la cantidad de días transcurridos entre la fecha ingresada y la fecha generada al azar. También mostrar la cantidad promedio de días transcurridos entre el total de los pares de fecha.

Ejercicio 15: Dada una lista de números naturales se solicita mostrar los números que están formados solo por dígitos pares. Además, mostrar el menor y el mayor de estos números (de los que cumplen la condición).

Ejercicio 16: Dada una lista de N caracteres, los cuales pueden ser dígitos y letras en minúsculas, y sin caracteres de puntuación, se solicita mostrar la cantidad de dígitos, de consonantes y de vocales se encontraron en la lista.

Nota: Crear una función que devuelva 1 si el carácter es un dígito, 0 si el carácter es una consonante o -1 si el carácter es vocal, es similar a la función del ejercicio 7.c). Activar la librería ctype.h para usar la función isdigit(char).

Ejercicio 17: Dada una lista de N caracteres, los cuales pueden ser espacios en blanco, dígitos y letras en minúscula, se solicita calcular la suma de los números contenidos en la lista. Considerar que un número está formado por una secuencia de dígitos Ejemplo:

Si la lista de 15 caracteres fuera **ad 78 c 12zy 11**

Entonces la suma sería 101

Nota: Utilizar la función creada en el ejercicio 16