Ejercicios propuestos para las pruebas:
PEI1 y PEI2 de Informática

1. Escribe un programa que calcule la siguiente expresión matemática para un valor dado de x:
$$\frac{3x^3+1}{\cos(x^2)+5\ log|x|}$$

2. Escribe un programa que dados tres números reales: a, b y c ($a \neq 0$) y calcule el módulo máximo de las raíces de la ecuación de segundo grado:
$$ax^2+bx+c=0$$

3. Escribe un programa que calcule la suma de los N primeros números primos.

4. Escribe un programa que calcule la suma de los N primeros números perfectos.

5. Escribe un programa que permita obtener la representación de un entero en base dos. El resultado debe ser una cadena de caracteres.

6. Escribe un programa que permita obtener la representación de un entero en base dieciséis. El resultado debe ser una cadena de caracteres.

7. Escribe un programa para calcular la suma de los N primeros términos de la serie numérica
$$S_N=\sum_{n=1}^{N}\frac{1}{n!}$$

8. Escribe un programa para calcular la suma de infinitos términos de la siguiente serie numérica:
$$S=\sum_{n=1}^{\infty}\frac{1}{n^2}$$
con precisión ϵ. Sabiendo que el resultado es $\pi^2/6$, sumar mientras $S-S_N>\epsilon$ y calcular el número mínimo de términos que hay que sumar para alcanzar esa aproximación.

9. Escribe una función que reciba un número natural N y devuelva una lista con los números primos menores o iguales que el dado. Nota: Para $N<2$ devolverá una lista vacía.

10. Escribe una función que dado un número entero N, devuelva una lista con los primeros N números perfectos. Nota: Un numero es perfecto si es la suma de todos sus divisoresde sin contar el mismo coincide con dicho número. Ej 6 es un número perfecto $6=1+2+3$; 28 es un número perfecto $28=1+2+4+7+14$.

11. Escribe una función que reciba dos listas y que devuelva dos listas: una con los elementos comunes y otra con los elementos diferentes.

12. Escribe una función que reciba dos cadenas de caracteres y que devuelva una lista con las letras que aparecen en ambas cadenas. Por ejemplo:

    - Para 'melón' y 'tomate' devolverá ['m', 'e'] (Distingue entre vocales con o sin tilde).
    - Para 'Violín' y 'lava' devolverá ['v', 'l'] (No distingue entre mayúsculas y minúsculas).

13. Escribe una función que elimine el elemento segundo mas pequeño de una lista. La función deberá devolver una lista con el elemento eliminado.

14. Escribe una función que cuente los elementos de una lista de números que están entre $[a, b]$ siendo a y b dos números reales. La función devolverá el número de elementos de la lista que cumplan esa condición.

15. Escribe una función que determine si una cadena de caracteres es un palíndromo. Un palíndromo es una palabra o frase que se lee igual en un sentido que en otro. Ej. "somos o no somos" es un palíndromo.

16. Escribe una función que determine si una cadena de caracteres es un palíndromo. Completa la función anterior teniendo en cuenta que los caracteres en mayúscula y las vocales acentuadas se consideran iguales para determinar si es palíndromo. Ej: "Dábale arroz a la zorra el abad" es un palíndromo.

17. Escribe una función que cuente el número de vocales que aparecen en una cadena de caracteres.

18. Escribe una función que elimine los elementos duplicados de una lista.

19. Escribir una función de nombre: `divisors` de devuelva una lista de los divisores de un número N.

20. Escribir una función de nombre: `mcd` que calcule el máximo común divisor de los números introducidos mediante el uso de la función anterior.

21. Dado el siguiente texto:

```
Hit the road Jack and don't you come back
No more, no more, no more, no more
Hit the road Jack and don't you come back no more
What you say?
Hit the road Jack and don't you come back
No more, no more, no more, no more
Hit the road Jack and don't you come back no more
Old woman, old woman, don't treat me so mean
You're the meanest old woman that I've ever seen
I guess if you said so
I'll have to pack my things and go (that's right)
Hit the road Jack and don't you come back
No more, no more, no more, no more
```

Determinar:

(a) El número de palabras diferentes. NOTA: las comas y los paréntesis no se consideran palabras. Tampoco son palabras los retornos de carro del final de línea.

(b) El número de repeticiones de cada palabra.

(c) La palabra con mayor número de repeticiones.

22. Escribe una función que determine si una subcadena `s` pertenece a una cadena de caracteres `S` dada. La función deberá devolver una lista con todos los índices de `S` donde empiece la subcadena `s`. Ej: S = "1234561234", find s="34". Return [2, 8]

23. Escribe una función que calcule una partición equespaciada del segmento $[a, b]$ de la recta real en N trozos. Devolverá una vector x cuyas componentes serán los nodos x_i de la partición con $N+1$ componentes.

24. Crear una función que calcule las imágenes de la función $f(x)$ en los puntos de la partición. La función tendrá dos argumentos: la función y una lista que contega la partición donde se calculan las imágenes. Devolverá una lista con los valores las imágenes.

25. Crear una función que calcule las imágenes de la función $f(x)$ en los puntos de la partición. La función tendrá dos argumentos: la función y un vector que contenga la partición donde se calculan las imágenes. Devolverá un vector cuyas componentes sean las imágenes.

26. La expresión general de un polinomio $p(x)$ de grado N es:

$$p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_N x^N$$

Se pide crear una función que tenga como argumentos de entrada una lista con los coeficientes a_i del polinomio y el valor x donde se pretende evaluar. La función deberá devolver el valor del polinomio evaluado en el punto x. La función debe ser válida para un número arbitrario de coeficientes y el argumento de entrada x es un valor real.

27. Escribir una función que calcule las imágenes $y_i = f(x_i)$ de las componentes de un vector x mediante la función $f(x)$ genérica.

28. Hacer aplicación de la función anterior cuando

$$f(x) = \begin{cases} \cos(\pi x) & x \leq 0 \\ 1 + \sin(\pi x) & x > 0 \end{cases}$$

29. Escribe una función de nombre gaussian que reciba como argumento una variable real x, otra variable real m cuyo valor por defecto sea 0 otra variable real s cuyo valor por defecto sea 1. Devolverá el resultado de la operación:

$$f(x) = \frac{1}{\sqrt{2\pi} s} \exp\left[-\frac{1}{2}\left(\frac{x-m}{s}\right)^2\right]$$

30. Una partícula se mueve en el plano describiendo un movimiento circular de radio R centrado en $(x, y) = (0, 0)$. Escribir una función que reciba el radio de giro R y el instante temporal t como variables reales y y que devuelva la posición: $[x(t), y(t)]$ como una tupla real usando la fórmula:

$$\begin{aligned} &f : \mathbb{R} \to \mathbb{R}^2 \\ &\vec{r}(t) : t \to (R\cos(t), R\,\mathrm{sen}(t)) \end{aligned}$$

31. Escribir una función con dos argumentos vectoriales de $\mathbb{R}^N$, que representan la posición de dos puntos, y que calcule a distancia euclídea entre dichos puntos.

32. Escribir una función de nombre: `is_inside(x, R, c)` que determine si un punto $x \in \mathbb{R}^N$ está en el interior de una hiperesfera de radio R y centro $c \in \mathbb{R}^N$. La función devolverá un valor lógico True o False dependiendo de la posición del punto.

33. Escribir una función de nombre derivative(x, h, funcion), que recibe como argumentos dos valores reales x, h y una función real de variable real. Devolverá el valor de la derivada de funcion en x de acuerdo con la fórmula:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

34. Escribir una función que dado un vector x, una función $f(x)$ y la función `derivative` del ejercicio anterior calcule las derivadas de $f(x)$ en las componentes de x

35. Escribe una función de nombre `Taylor_seno(` x`, tol)` que aproxime mediante un desarrollo en serie de potencias la función `seno(x)` con una toleracia dada: `tol` en un punto `x` dado. La función deberá implementar la aproximación mediante el polinomio de Maclaurin.

$$\sin(x) \sim T(x) = \sum_{n=1}^{M} \frac{(-1)^{n+1} x^{2n-1}}{(2n-1)!}$$

El número de términos en el sumatorio vendrá determinado por la variable tol de manera que se cumpla: $|\sin(x) - T(x)| <$ `tol`.

36. Series de potencia. Aproximar mediante un desarrollo truncado en serie de Taylor de la forma:

$$f(x) = \sum_{k=0}^{M} a_k \ (x - x_0)^k, \qquad\qquad a_k = \frac{f^{(k)}(x_0)}{k!},$$

para las funciones $f : \mathbb{R} \to \mathbb{R}$, siguientes:

(a) $f(x) = e^x$,

(b) $f(x) = \sin(x)$,

(c) $f(x) = \cosh(x)$,

(d) $f(x) = \dfrac{1}{1-x}$.

Para cada desarrollo en serie se pide:

(a) Calcular el desarrollo para un valor de M genérico ($M = 1, 2, 3, 4, 5, \ldots$) mediante un bucle desde $k = 0$ hasta $k = M$.

(b) Implementar un desarrollo para infinitos términos que para de sumar cuando se alcance la máxima precisión. Realizar la implementación mediante una sentencia de control `while`.

(c) Comparar con las funciones intrínsecas correspondientes.

(d) Discutir los resultados obtenidos.

37. Vectores y matrices.

Sean los vectores $V, W \in \mathbb{R}^N$ y la matriz de Vandermode $A \in \mathcal{M}_{N\times N}(\mathbb{R})$:

$$\{v_i = 1/i^2, \quad i = 1, \dots N\} \qquad \{w_i = (-1)^{i+1}/(2i+1), \quad i = 1, \dots N\}$$

$$\{a_{ij} = (i/N)^{j-1}, \quad i = 1, \dots N, \quad j = 1, \dots N\}.$$

Escribir un programa que realice las operaciones siguientes y que muestre los resultados por pantalla:

(a) Sumar todas las componentes del vector V.
(b) Sumar todas las componentes de la matriz A.
(c) Sumar las componentes del vector V mayores que cero.
(d) Sumar las componentes de la matriz A menores que cero.
(e) Producto escalar de los vectores V y W.
(f) Producto escalar del vector V y la columna N de la matriz A.
(g) Multiplicar la matriz A por el vector V.
(h) Matriz traspuesta de la matriz A
(i) Obtener el valor máximo de la matriz A y su posición.

38. Dada la matriz de Vandermonde $A_M \in \mathcal{M}_{M\times M}(\mathbb{R})$ del hito anterior, calcular las siguientes operaciones:

(a) Suma de las trazas de A_M para matrices de diferente dimensión.

$$S_1 = \sum_{M=1}^{10} trace(A_M)$$

(b) Suma de las trazas del cuadrado de A_M para matrices de diferente dimensión.

$$S_2 = \sum_{M=1}^{5} trace(A_M^2)$$

(c) Traza de la suma de potencias de A_M con $M = 8$.

$$S_3 = trace\left(\sum_{k=0}^{5} A_8^k\right)$$

Implementar la operaciones anteriores mediante un módulo llamado **Algebra** que contenga la función **trace** que calcula la traza de una matriz, la función **Vandermonde** que crea la matriz $A_M \in \mathcal{M}_{M\times M}(\mathbb{R})$ y la función **power** que permite calcular la potencia k–ésima de cualquier matriz cuadrada.

39. Sentencias de entrada y salida. Crear los ficheros de datos con nombres `data1.txt` e `data2.txt` con la información siguiente: Contenido del fichero de entrada `data1.txt` :

```
VARIABLES=t, x, y, z, u
1.2,      3.4,     6.2,    -14.0,    0.1
-25.2,    -8.6,    5.1,     9.9,    17.0
-1.0,     -2.0,    -5.4,    -8.6,    0.0
3.14,     -11.9,   -7.0,    -12.1,   9.2
6.66,    5.32,    0.001,   0.2,     0.001
```

Contenido del fichero de entrada `data2.txt` :

```
VARIABLES=t, x1, x2, x3, x4, x5, x6
1.2,      3.4,     6.2,    -14.0,    0.1,      4.89,   7.54
25.2,    -8.6,    5.1,    12.0,     9.9,      12.24,  17.0
0.0,      34.5,    -1.0,   -2.0,     -43.04,   -8.6,    0.0
3.14,     -11.9,   71.0,   7.0,     17.0,      -12.1,   9.2
 6.66,     5.32,    0.001,   0.2,     0.001,    0.008,   -0.027
54.0,     77.1,    -9.002,  -13.2,   0.017,    65.53,   -0.021
 23.04,    -51.98,  -34.2,   9.99,    5.34,     8.87,    3.22
```

Escribir una función `load_matrix` que permita crear una matriz A con las dimensiones del fichero de datos y que almacene los valores de dicho fichero mediante asignaciones del tipo:

```
A = load_matrix('data1.txt')      A = load_matrix('data2.txt')
```

La función `load_matrix` deberá permitir la carga de datos con cualquier formato. Una vez cargados los valores de la matriz A, crear dos vectores U y V que contengan la primera y segunda columnas de la matriz A.

40. Escribe una función de nombre collatz (N) que reciba como argumento un numero natural N y devuelva una lista con la sucesión $\{a_1, a_2, \dots, a_i, \dots\}$ generada a partir de $a_1 = N$ usando la fórmula: Con:

$$a_{i+1} = f(a_i)$$

$$f(x) = \begin{cases} \frac{x}{2} & \text{si x es par} \\ 3x + 1 & \text{si x es impar} \end{cases}$$

Devolverá la lista cuando se alcance $a_i = 1$
Ej collatz $(7) = [7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1]$