

Parcial de Programación

Un número perfecto es un número entero positivo que es igual a la suma de sus divisores propios ; por ejemplo $6=1+2+3$. Diseñe un algoritmo en matlab para encontrar los primeros 4 números perfectos

```
function total_suma=divisores(x)
suma_d=0;
for k=1:x-1
    if mod(x,k)==0
        k;
        suma_d= suma_d+k;
    end
end
total_suma=suma_d;
end
```

```
% Hemos construido una función que calcula los divisores de un número
divisores(28)
```

```
ans =
28
```

```
% Utilizaremos la función creada anteriormente para poder calcular los
primeros 4 números perfectos
for numero_perfecto=1:10000
    if divisores(numero_perfecto)==numero_perfecto
        numero_perfecto
    end
end
```

```
numero_perfecto =
6
numero_perfecto =
28
numero_perfecto =
496
numero_perfecto =
8128
```

Punto 2

Un palíndromo es una frase que se puede leer igual al derecho y al revés. Ejemplo: “Dábale arroz a la zorra el abad”. Escriba un algoritmo que verifique si una frase es un palíndromo. Escriba la frase (sin espacios en blanco, sin tildes, sin mayúsculas) en un vector. Ejemplo: `x='dabalearrozalazorraelabad'`.

```
% Primero creamos la función para comprobar si es palindromo

function total_pal= hallar_pal(variable)
total_pal= true;
```

```

contador_a=1;
N = strlen(variable);
w= strlen(variable);

for k=1:N
    if variable(contador_a)==variable(w)
        %% Esta parte es muy importatne ya que recorre la palabra del
primer al ultimo caracter
        contador_a=contador_a+1;
        %% Recorre la palabra del ultimo al primer caracter
        w=w-1;
    else
        total_pal=false;
    end
end
end

%% Esta sección nos pregunta una palabra y usamos la función hallar_pal

evaluar= input("Compruebe si es palíndromo: ", "s")

```

```

evaluar =
'ana'

```

```

if hallar_pal(evaluar)
    disp('La palabra ingresada es un palíndromo')
else
    disp('La palabra ingresada es un palíndromo')
end

```

La palabra ingresada es un palíndromo

Punto 3

3a. Implementar una funcion en Matlab para calcular la funcion

$$\frac{\pi}{4} - \frac{2}{\pi} \sum_{k=1}^N \frac{\cos(2k-1)x}{(2k-1)^2} + \sum_{k=1}^N \frac{(-1)^{k+1}}{k} \sin kx$$

```

function z=suma_parcial(x, N)
suma_uno=0;
suma_dos=0;

%Este es el código correspondiente a la primera sumatoria
for n =1:N
    term_uno=cos((2*n-1)*x)/((2*n-1)^2);
    suma_uno=suma_uno+term_uno;
end

%Este es el código correspondiente a la segunda sumatoria
for m =1:N

```

```

    term_dos=(-1)^(m+1)/m*sin(m*x);
    suma_dos=suma_dos+term_dos;
end

%% Aca calculamos la expresión completa
z=pi/4-2/pi*suma_uno+suma_dos
end

```

```
suma_parcial(pi/2,10)
```

```

z =
1.6203
ans =
1.6203

```

Implementar una funcion para graficar $S(x, N)$ en un intervalo $[a, b]$ para un valor de N

```

%% En este apartado graficaremos la suma para cualquier intervalo ingresado
por el usuario

function graficar(a,b,N)
x = a:0.2:b;
y = suma_parcial(x,N);
plot(x,y)
end

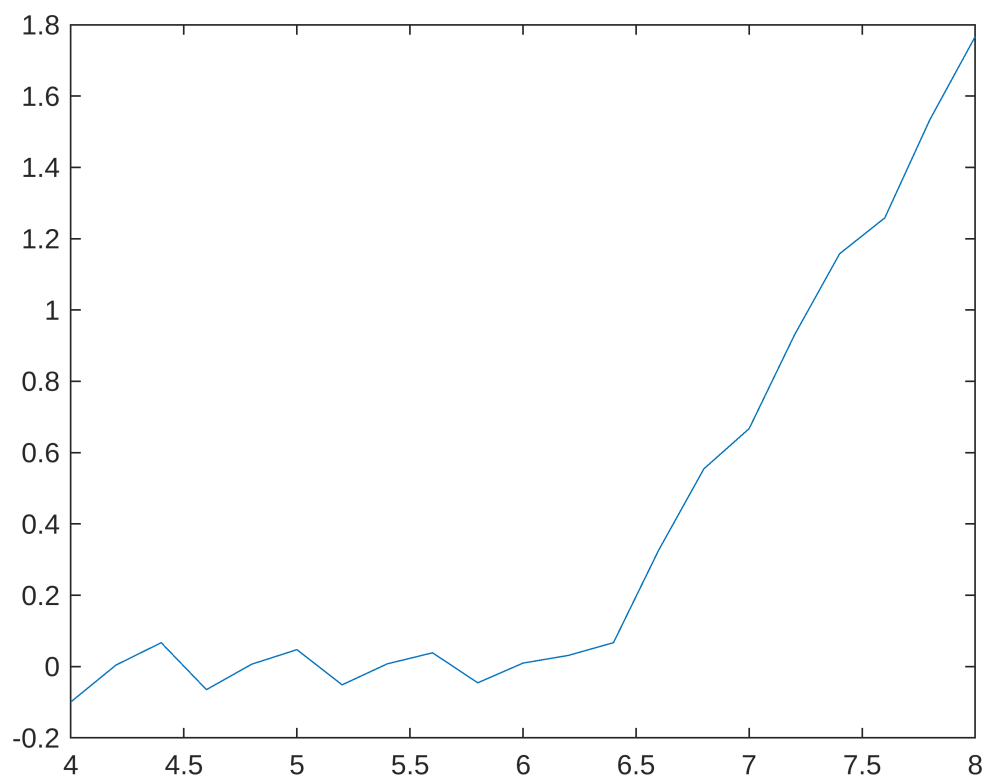
```

```
graficar(4,8,10)
```

```

z = 1x21
    -0.0992    0.0043    0.0670   -0.0647    0.0067    0.0475   -0.0512    0.0077 ...

```



Camilo Rafael Pérez Chaves

Parcial 1 - Programación