Proyecto Algoritmo de Comparación de Secretos

Daniela Severo Rico

Sherlyn Mendoza Reyes

La empresa Tesla, requiere encriptar su contraseña, que será 1402 a continuación veremos el proceso.

En esta primera fase que será

Encriptación

```
secreto = 1402;
k = 3;
n = 6;

partes = encriptar(secreto, k, n);
```

```
Coeficientes del polinomio:
        1402           2           5

Partes generadas (x, y):
           1        1409
           2        1426
           3        1453
           4        1490
           5        1537
           6        1594
```

```
disp('Las partes son:');
```

```
Las partes son:
```

```
disp(partes);
```

```
           1        1409
           2        1426
           3        1453
           4        1490
           5        1537
           6        1594
```

En el proceso anterios los coeficientes del polinomio se generan con el secreto (a0)

- Recorremos valores de x de 1 a n
- En cada paso calculamos y sumando cada término del polinomio
- Mostramos las coordenadas generadas

Lo cual hace que tengamos las coordenadas listas para su uso y podamos proceder

Ahora procedemos a

Desencriptar

```
puntos_seleccionados = partes(1:k, :);
```

```
secreto_recuperado = desencriptar(puntos_seleccionados);
```

```
El secreto reconstruido es:
        1402
```

En este punto se empieza con `secreto = 0` porque vamos a ir sumando los términos calculados para cada punto (x,y)

- Utilizamos fórmula de Lagrange para cada punto en donde usamos un `for`
- Para cada punto, sumamo yi*Li0 al secreto
- Redondeamos y mostramos el secreto

Ahora tomaremos los valores diferentes donde k=4 y n=10

```
secreto = 1402;
k = 4;

n = 10;

partes = encriptar(secreto, k, n);
```

```
Coeficientes del polinomio:
        1402          10           8          10

Partes generadas (x, y):
           1        1430
           2        1534
           3        1774
           4        2210
           5        2902
           6        3910
           7        5294
           8        7114
           9        9430
          10       12302
```

```
disp('Las partes son:');
```

```
Las partes son:
```

```
disp(partes);
```

```
           1        1430
           2        1534
           3        1774
           4        2210
           5        2902
           6        3910
           7        5294
           8        7114
           9        9430
          10       12302
```

Ahora se muestra la desencriptacion

```
puntos_seleccionados = partes(1:k, :);
```

```
secreto_recuperado = desencriptar(puntos_seleccionados);
```

```
El secreto reconstruido es:
        1402
```

Y así terminamos con el proyecto pedido por el cliente.

```
function partes = encriptar(secreto, k, n)

    coeficientes = [secreto, randi([1, 10], 1, k-1)];

    disp('Coeficientes del polinomio:');
    disp(coeficientes);

    for i = 1:n
        x = i;
        y = 0;
        for j = 1:k
            y = y + coeficientes(j) * x^(j-1);
        end
        partes(i, :) = [x, y];
    end

    disp('Partes generadas (x, y):');
    disp(partes);
end
```

```
function secreto = desencriptar(puntos)

    secreto = 0;
    k = size(puntos, 1);
    for i = 1:k
        Li = 1;
        for j = 1:k
            if i ~= j
                Li = Li * (-puntos(j, 1)) / (puntos(i, 1) - puntos(j, 1));
            end
        end
        secreto = secreto + puntos(i, 2) * Li;
    end
    disp('El secreto reconstruido es:');
    disp(round(secreto));
end
```