

## Prueba Adriana Castillo

Se buscar encontrar la eficiencia de la generación de números pseudo-aleatorios a través de los métodos de cuadrados medios y congruencia lineal, para ello se debe seguir el siguiente proceso:

1. A través de la misma api generar una semilla diferente.
2. Encontrar el numero de iteraciones hasta que se repita uno de sus datos.
3. Generar 100 simulaciones con diferentes semillas.
4. Generar un histograma con el resultado obtenido por cada método.
5. Agregar sus conclusiones, opiniones y recomendaciones

## Método de cuadrados medios

1. A través de la misma api generar una semilla diferente.

[illegible]

1. Encontrar el numero de iteraciones hasta que se repita uno de sus datos.

```
In [2]: def cuadrados_medios(semilla):
    rep=1
    iteraciones = 1000
    digitos = 4
    val_xn=[]
    val_xn_m=[]
    val_ui=[]
    val_rn=[]
    div=[int(str(num).ljust(digitos+1, "0")) for num in [1]]
    ui=""
    izq=int(digitos/2)
    der=digitos-izq

    for i in range(iteraciones):
        val_xn.append(semilla)
        val_xn_m.append(semilla*semilla)
        numero = str(semilla*semilla)
        mitad=math.floor(len(numero)/2)
        for p in range(mitad-izq, mitad+der, 1): ui+=numero[p]
        if ui in val_ui:
            #print('Repetido', ui, ' pos:', rep)
            cuadrados[rep] = cuadrados[rep] + 1 if rep in cuadrados else 1
            break
        unicos.append(ui)
        val_ui.append(ui)
        val_rn.append(int(ui)/div[0])
        semilla=int(val_ui[-1])
        ui=''
        rep+=1
    #print(val_ui)
```

1. Generar 100 simulaciones con diferentes semillas

```
In [3]: cuadrados={}
for x in val_ui_semillas:
    #print(' ',ite,'o --> Semilla:', x)
    #ite+=1
    cuadrados_medios(int(x))


print(cuadrados)

{20: 1, 19: 1, 18: 1, 17: 1, 16: 1, 15: 1, 14: 1, 13: 1, 12: 1, 11: 1, 10: 1, 9: 1, 8: 1, 7: 1, 6: 1, 5: 1, 4: 1, 3: 1, 2: 1, 1: 1}
```

1. Generar un histograma con el resultado.

```
In [9]: import matplotlib.pyplot as plt

D = cuadrados
plt.bar(range(len(D)), list(D.values()), align='center')
plt.xticks(range(len(D)), list(D.keys()))
plt.show()
```



## Método de congruencias lineales

1. A través de la misma api generar una semilla diferente

```
In [5]: import pandas as pd, math
congruencias={}
iteraciones_cl =100
semilla_cl= 1 ##Semilla inicial
a_cl=3
b_cl=5
m_cl=19
val_xn_cl=[semilla_cl]
val_un_cl=['']

for i in range(iteraciones_cl): val_xn_cl.append((a_cl*val_xn_cl[-1]+b_cl)%m_cl) ; val_un_c
l.append(val_xn_cl[-1]/m_cl)

print('Semillas generadas: \n',val_xn_cl)

Semillas generadas:
[1, 8, 10, 16, 15, 12, 3, 14, 9, 13, 6, 4, 17, 18, 2, 11, 0, 5, 1, 8, 10, 16, 15, 12, 3, 14,
9, 13, 6, 4, 17, 18, 2, 11, 0, 5, 1, 8, 10, 16, 15, 12, 3, 14, 9, 13, 6, 4, 17, 18, 2, 11, 0,
5, 1, 8, 10, 16, 15, 12, 3, 14, 9, 13, 6, 4, 17, 18, 2, 11, 0, 5, 1, 8, 10, 16, 15, 12, 3, 1
4, 9, 13, 6, 4, 17, 18, 2, 11, 0, 5, 1, 8, 10, 16, 15, 12, 3, 14, 9, 13, 6]
```

1. Encontrar el numero de iteraciones hasta que se repita uno de sus datos.

```
In [6]: def congruencias_lineales(semillas):
        semilla=semillas
        rep_c=1
        iteraciones = 1000
        a=3
        b=9
        m=5
        val_xn=[semilla]
        val_un=['']

        for i in range(iteraciones):
            xn=(a*val_xn[-1]+b)%m
            if xn in val_xn:
                congruencias[rep_c] = congruencias[rep_c] + 1 if rep_c in congruencias else 1
                break

            val_xn.append(xn)
            val_un.append(val_xn[-1]/m)
            rep_c+=1
        #print(val_xn)
```

1. Generar 100 simulaciones con diferentes semillas.

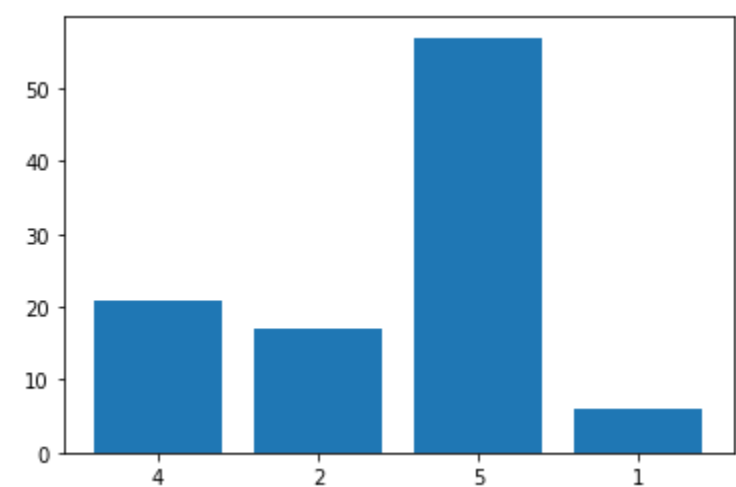
```
In [7]: for x in val_xn_c1:
         congruencias_lineales(int(x))
         print(congruencias)

(4: 21, 2: 17, 5: 57, 1: 6)
```

1. Generar un histograma con el resultado

```
In [8]: import matplotlib.pyplot as plt

D = congruencias
plt.bar(range(len(D)), list(D.values()), align='center')
plt.xticks(range(len(D)), list(D.keys()))
plt.show()
```



## Conclusiones, opiniones y recomendaciones

Como conclusión podemos observar que en el metodo de cuadrados medios en la iteración 5 ocurre el mayor numero de repeticiones y en las demas solo se repite una vez, mientras que en el metodo de congruencia lineal donde aparecen las repeticiones va variando y no unicamente en la iteración número 5.