INSTITUTO POLITECNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO (ESCOM)

ANÁLISIS DE ALGORITMOS

NOMBRE DEL ALUMNO:

- SANTOS MÉNDEZ ULISES JESÚS

EJERCICIO 01:

- CALCULAR EL NUMERO DE IMPRESIONES

FECHA DE ENTREGA:

- 04/03/2022

GRUPO:

- 3CM14

Calcular el número de impresiones

- Determinar para los siguientes códigos el modelo matemático que determine el número de impresiones en términos de “n” que cada uno realiza de la palabra “Algoritmos” y comprobar empíricamente el resultado.
- Determine una función f(n) que modele el número de impresiones de la cadena “Algoritmos” de cada función
- Contraste sus funciones con la prueba empírica para los 20 valores de:

$$n = \{-1,0,1,2,3,5,15,20,100,409,500,593,1000,1471,1500,2801,3000,5000,10000,20000\}$$

Código 01:

01.c

```c
#include<stdio.h>

int main(void){
	int n,i;
	printf("Ingresa el valor de n: ");
	scanf("%d",&n);
	for(i=10;i<n*5;i*=2){
		printf("Algoritmos\n");
	}
	return 0;
}
```

Si se hace el modelo de una recta numérica además de las pruebas para valores chicos de n, se tiene un rango de valores de 0 para n=-1,0,1,2, el incremento de i hasta el límite de 5n estará dada por la expresión:

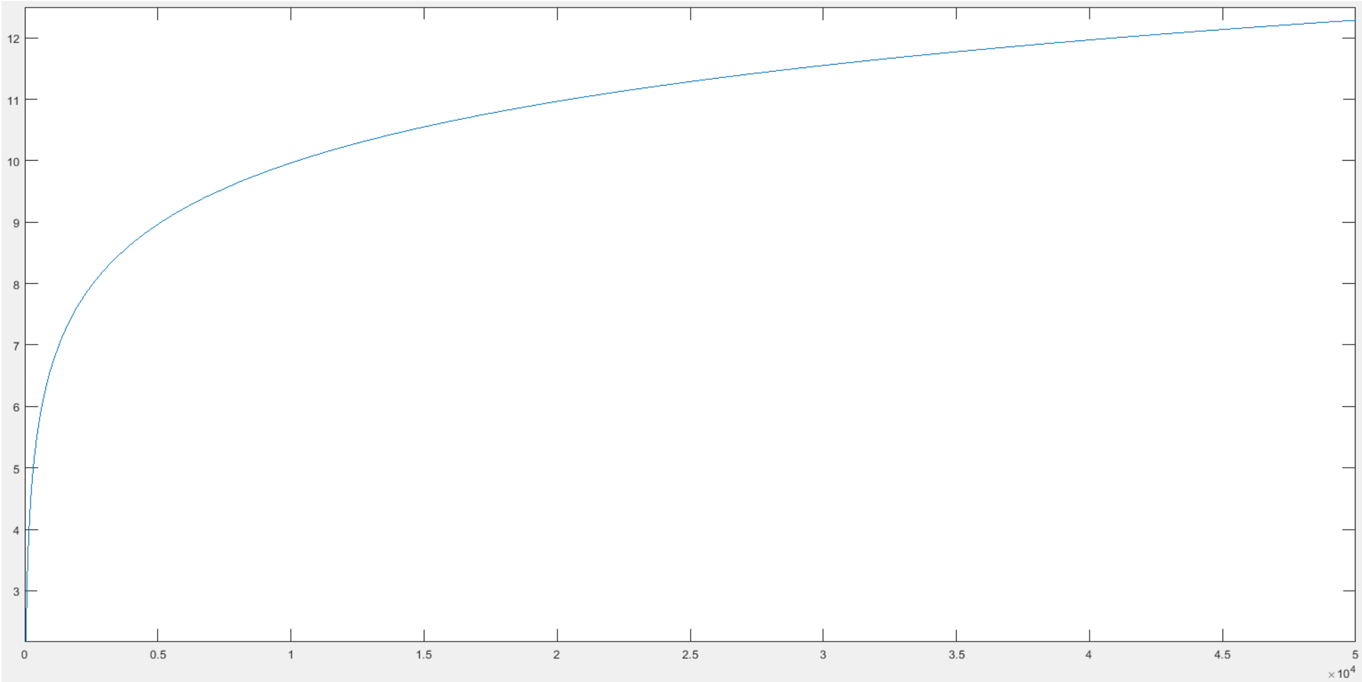
$$2^x = 5n$$

$$x = \log_2 5n$$

A su vez se tiene un salto de 10 en 10 que va incrementando de 10,20 posteriormente 40 y después 80 de forma sucesiva.

$$f(n) = \log_2 5n - \log_2 10$$

Se procede a graficar la función f(n) en Matlab



Se hizo una comparación de resultados teóricos y prácticos con el rango de valores que se nos fue dado:

n	Valor teórico	Valor empírico
-1	Indefinido	0
0	Indefinido	0
1	0	0
2	0	0
3	1	1
5	2	2
15	3	3
20	4	4
100	6	6
409	8	8
500	8	8
593	8	9
1000	9	9
1471	10	10
1500	10	10
2801	11	11
3000	11	11
5000	12	12
10000	13	13
20000	14	14

Código 02:

```
02.c
#include<stdio.h>

int main(){
    int n,j,i,cont=0;
    printf("Ingresa el valor de n: ");
    scanf("%d",&n);
    for(j=n;j>1;j/=2){
        if(j<(n/2)){
            for(i=0;i<n;i+=2){
                cont++;
                printf("%d.- Algoritmos\n",cont);
            }
        }
    }
    return 0;
}
```

Se tiene inicialmente el ciclo for donde parte de n y en cada salto o incremento el valor de j comenzará a dividirse entre dos sucesivamente tomando en cuenta que j=n entonces se expresaría como:

$$\frac{n}{2^1},\frac{n}{2^2},\dots,\frac{n}{2^x}=1$$

$$n=2^x$$

$$\log_2 n = x$$

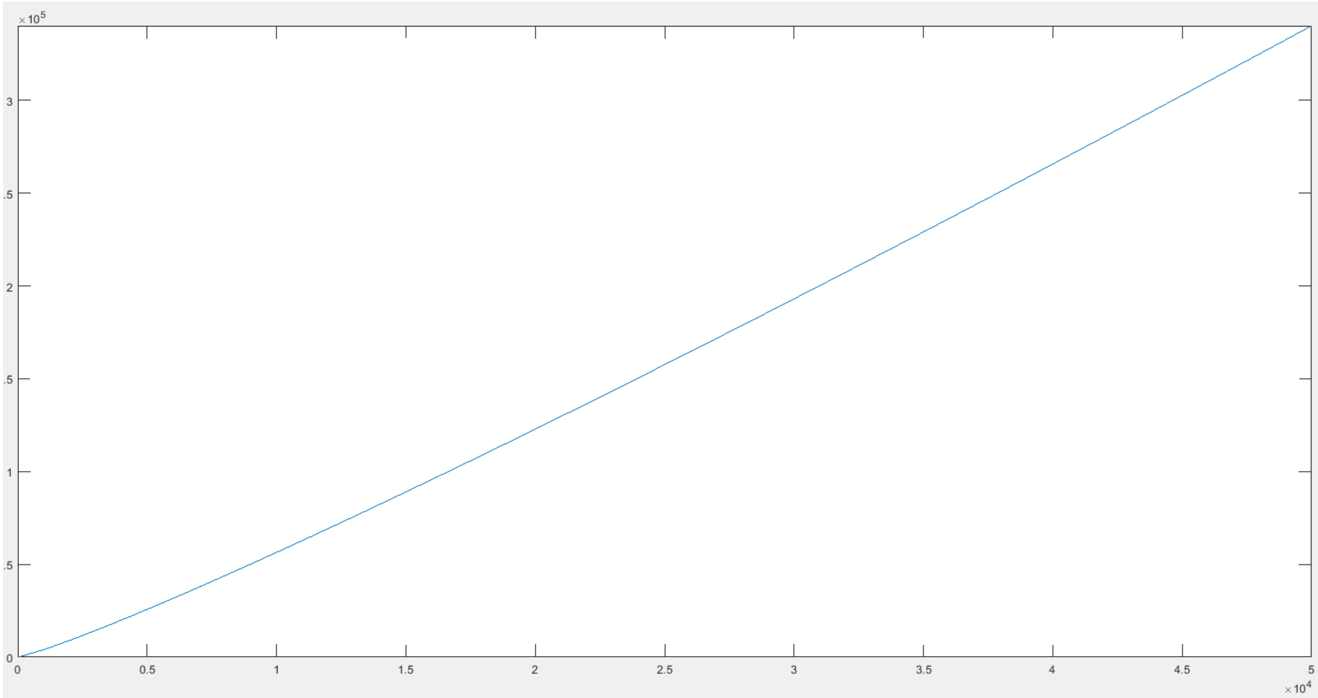
Considerando que la variable comenzara a ser menor que $\frac{n}{2}$ considerando que al entrar al for después del if y si este entra dos veces en el for por lo tanto se asume que se le deben restar dos para que así pueda entrar al if.

Considerando la condicional y que dentro del if hay un for se imprimirá este $\frac{n}{2}$ veces.

$$f(n)=(\log_2 n-2)\left(\frac{n}{2}\right)$$

Considerando que el logaritmo lo tomaremos en su parte entera o piso.

Procedemos a realizar la gráfica en MATLAB



Se hizo una comparación de resultados teóricos y prácticos con el rango de valores que se nos fue dado:

n	Valor teórico	Valor empírico
-1	Indefinido	0
0	Indefinido	0
1	-1	0
2	-1	0
3	-1	0
5	0	0
15	7	8
20	20	20
100	200	200
409	1227	1230
500	1500	1500
593	2075	2079
1000	3500	3500
1471	5884	5888
1500	6000	6000
2801	12604	12609
3000	13500	13500
5000	25000	25000
10000	55000	55000
20000	120000	120000

Código 03:

```c
#include<stdio.h>

int main(){
    int n,j,i,k,cont=0;
    printf("Ingresa el valor de n: ");
    scanf("%d",&n);
    for(i=0;i<n*5;i+=2){
        for(j=0;j<2*n;j++){
            for(k=j;k<n;k++){
                cont++;
                printf("%d.- Algoritmos",cont);
            }
        }
    }
    return 0;
}
```

Analizando el primer ciclo for obtenemos que el primer ciclo inicia en $\frac{5n}{2}$, el segundo ciclo for corresponde a 2n y el tercer ciclo es n-j, inicialmente se hicieron diversas pruebas con valores pequeños, el segundo ciclo for entra 2n veces pero el tercer ciclo solo entra n veces obteniendo la siguiente función:

$$\left(\sum_{j=0}^{n}(n-j)\right)\left(\frac{5n}{2}\right)$$

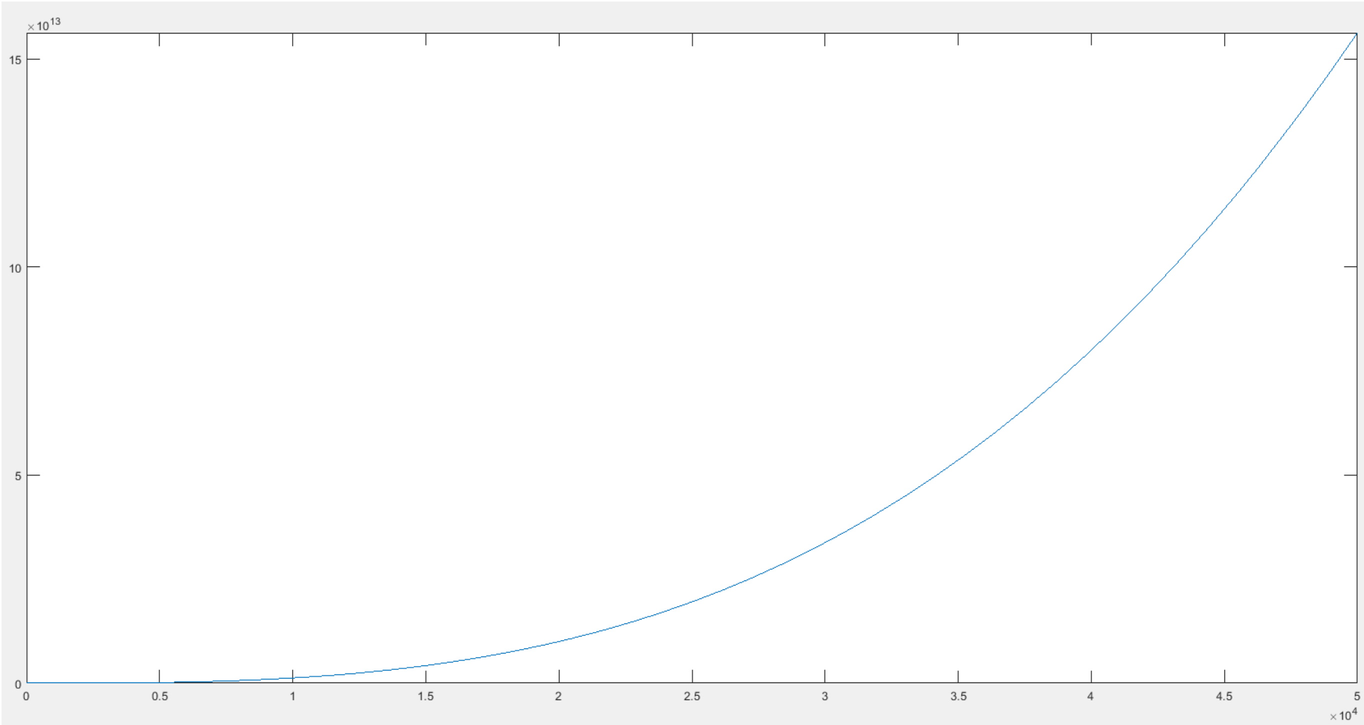
Con ayuda de una aplicación se obtiene la convergencia de la serie:

$$\sum_{j=0}^{n} n - j = n + \frac{n^2 - n}{2}$$

Nuestra función resultante es:

$$f(n) = \frac{5n^2(1+n)}{4}$$

Procedemos a generar la gráfica de la función en MATLAB



Se hizo una comparación de resultados teóricos y prácticos con el rango de valores que se nos fue dado:

n	Valor teórico	Valor empírico
-1	0	0
0	0	0
1	3	3
2	15	15
3	48	48
5	195	195
15	4560	4560
20	10500	10500
100	1262500	1262500
409	85773435	85773435
500	156562500	156562500
593	261187443	261187443
1000	1251250000	1251250000
1471	3982008768	3195852800
1500	4221562500	7340479600
2801	27481179603	1711375827
3000	33761250000	5984883680
5000	156281250000	1662427344
10000	1250125×10^{12}	2895168640
20000	100005×10^{13}	1816134912

Código 04:

```c
#include<stdio.h>

int main(){
    int n,j,i,cont=0;
    printf("Ingresa el valor de n: ");
    scanf("%d",&n);
    i=n;
    while(i>=0){
        for(j=n;i<j;i-=2,j/=2){
            cont++;
            printf("%d.- Algoritmos",cont);
        }
    }
    return 0;
}
```

Al realizar el análisis para este programa así como el teórico se observa que para todo valor nunca se cumple la condición del ciclo for esto haciendo que la condición se ejecute indeterminadamente ya que i siempre será mayor que 0 entonces:

$$f(n) = IND$$

Código 05:

```c
#include<stdio.h>

int main(){
    int n,j,i,cont=0;
    printf("Ingresa el valor de n: ");
    scanf("%d",&n);
    for(i=1;i<4*n;i*=2){
        for(j=i;j<5*n;j+=3){
            cont++;
            printf("%d.- Algoritmos",cont);
        }
    }
    return 0;
}
```

En el primer ciclo for se observan los saltos que dan así como el incremento teniendo una razón:

$$2^x = 4n$$

$$x = \log_2 4n$$

Dentro de este ciclo for se tiene otro ciclo anidado en donde se considera una sumatoria que va incrementando en pares siendo esto generar una expresión para que genere los pares que se van a sumar:

$$\sum_{i=1}^{n}\left(\frac{5n}{2} - (2i-2)\right)$$

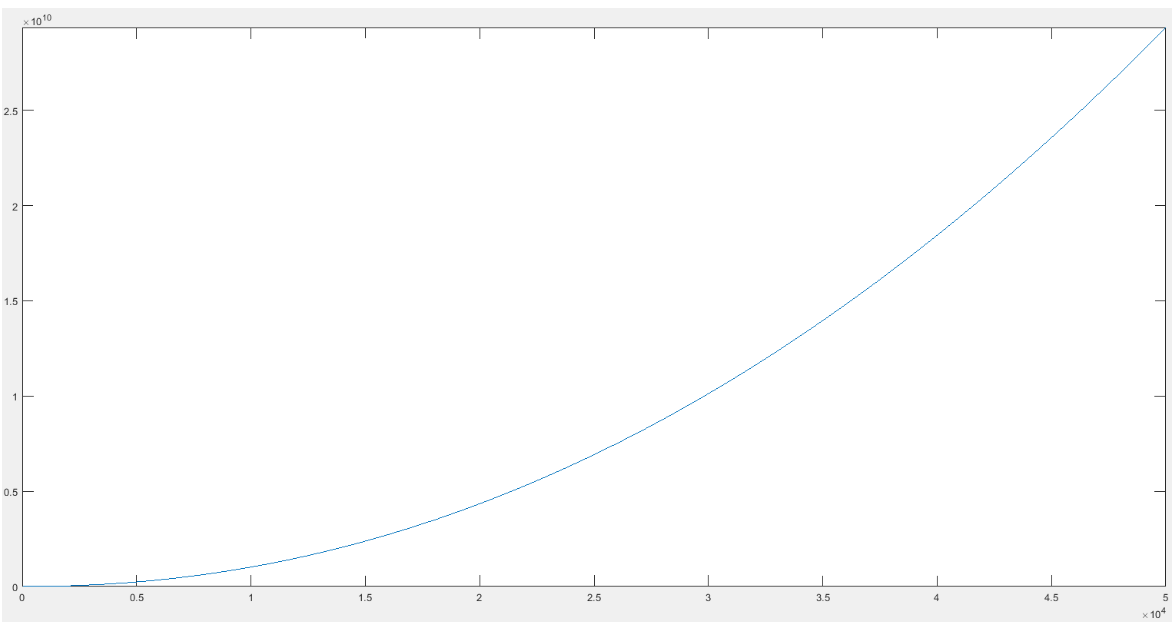
Se obtuvo la convergencia con ayuda de una aplicación

$$\sum_{i=1}^{n} \frac{5n}{3} - \left(2i - 2\right) = -n^2 + n + \frac{5n^2}{3}$$

Simplificando tenemos la función de la iteración de los ciclos for:

$$f(n) = (\log_2 4n)\left(\frac{2n^2}{3} + n\right)$$

Procedemos a graficar la función en MATLAB



Se hizo una comparación de resultados teóricos y prácticos con el rango de valores que se nos fue dado:

n	Valor teórico	Valor empírico
-1	0	0
0	0	0
1	3	3
2	14	8
3	36	17
5	87	32
15	990	132
20	1720	192
100	60900	1333
409	1231226	6820
500	1838833	8486
593	2585282	10497
1000	7989155	18639
1471	18083004	29146
1500	18844946	29776
2801	70395596	59898
3000	81345132	64546
5000	238199978	114000
10000	1019333702	244827
20000	4343715722	522979

En esta situación los resultados no son los mismos por lo que hay una gran diferencia en el valor teórico y el valor empírico.