# Archivo:Logo Instituto Politécnico Nacional.png - Wikipedia, la enciclopedia libreArchivo:EscudoESCOM.png - Wikipedia, la enciclopedia libreINSTITUTO POLITÉCNICO NACIONAL

# ESCUELA SUPERIOR DE CÓMPUTO

Vargas Hernández Carlo Ariel

Análisis de algoritmos

3CM15

Ejercicios 04



Ejercicio 1

func **SumCuadratica3May**(**A,n**){

If(**A**[1]>**A**[2]&& **A**[1]>**A**[3]){

**m1**=**A**[1];

If(**A**[2]>**A**[3]){

**m2**=**A**[2];

**m3**=**A**[3];

}else{

**m2**=**A**[3];

**m3**=**A**[2];

}

}else if(**A**[2]>**A**[1]&& **A**[2]>**A**[3]){

**m1**=**A**[2];

If(**A**[1]>**A[**3]){

**m2=A**[1];

**m3**=**A**[3];

}else{

**m2**=**A**[3];

**m3**=**A**[1];

}

}else{

**m1**=**A**[3];

If(**A**[1]>**A**[2]){

**m2**=**A**[1];

**m3**=**A**[2];

}else{

**m2**=**A**[2];

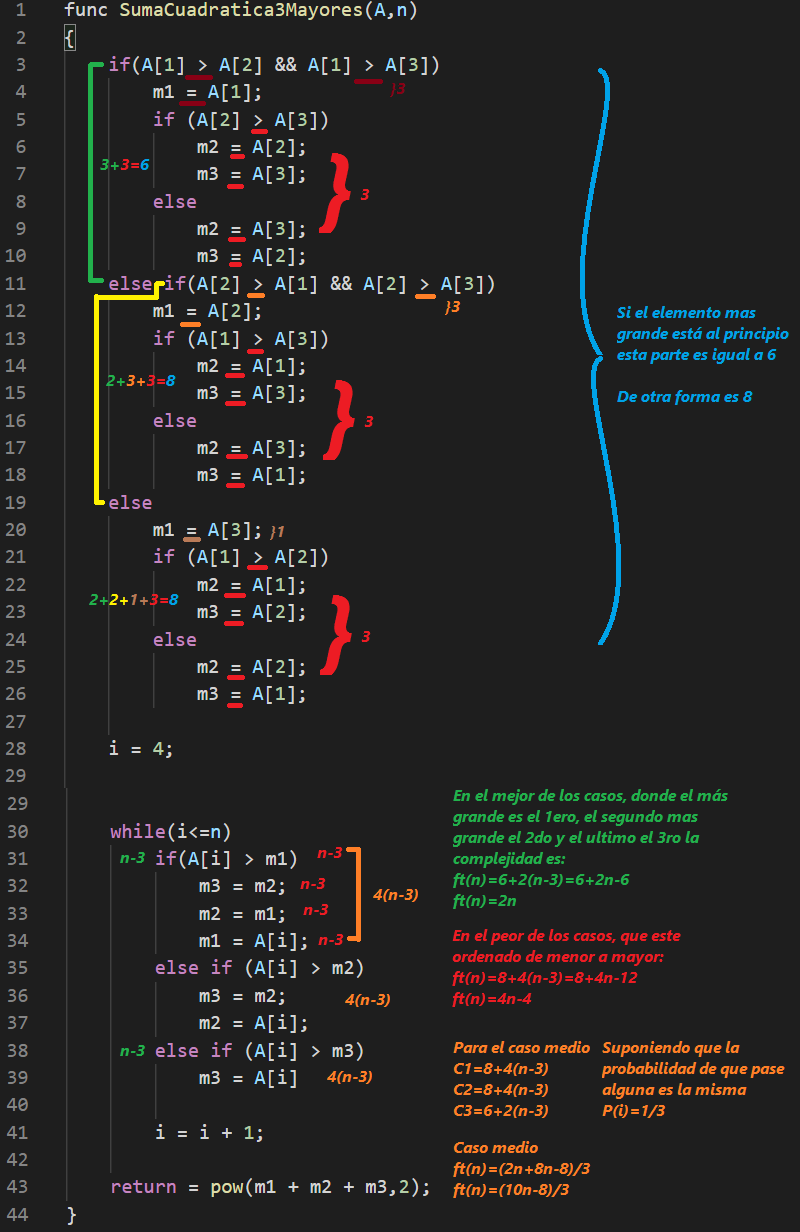
**m3**=**A**[1];

}

}

}

Para esta sección de código la complejidad la calcule de la siguiente manera:



while(i<=n){

if(**A**[i]>**m1**){

**m3**=**m2**;

**m2**=**m1**;

**m1**=**A**[i];

}else if(**A**[i]>**m2**){

**m3**=**m2**;

**m2**=**A**[i];

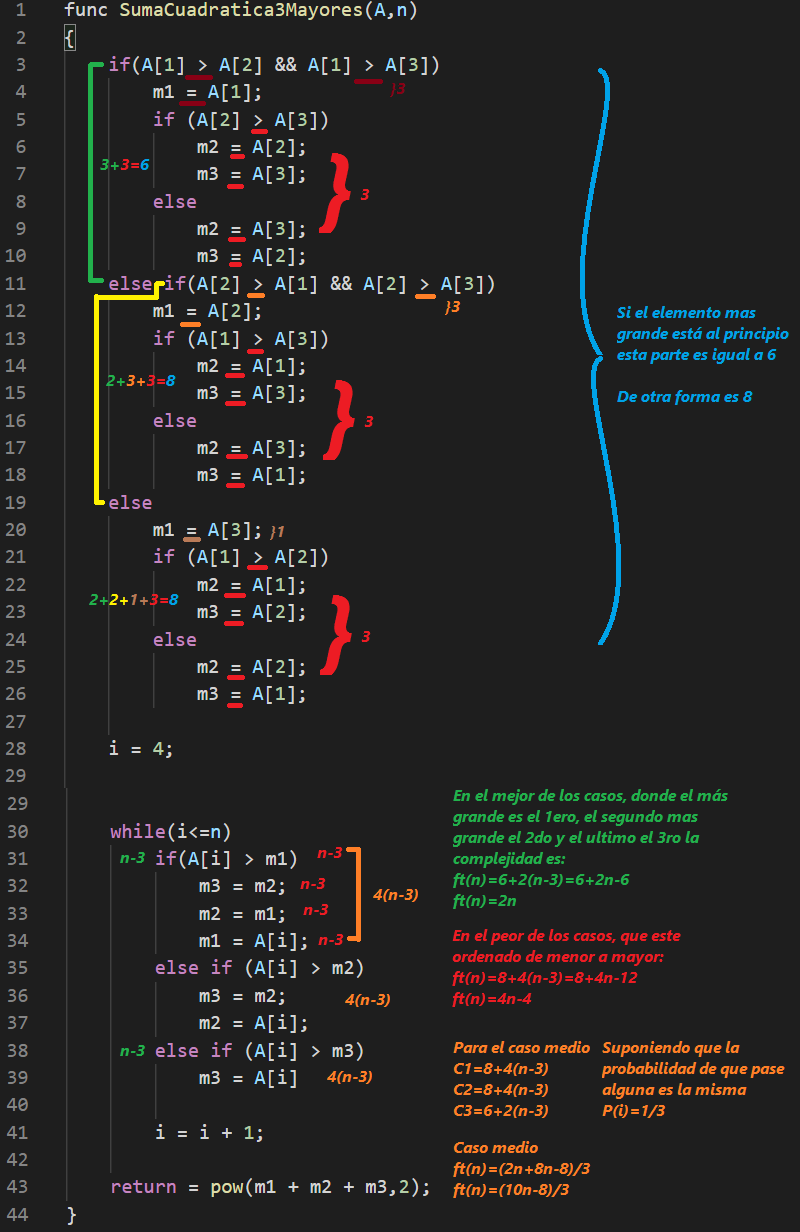
}else if(**A**[i]>**m3**){

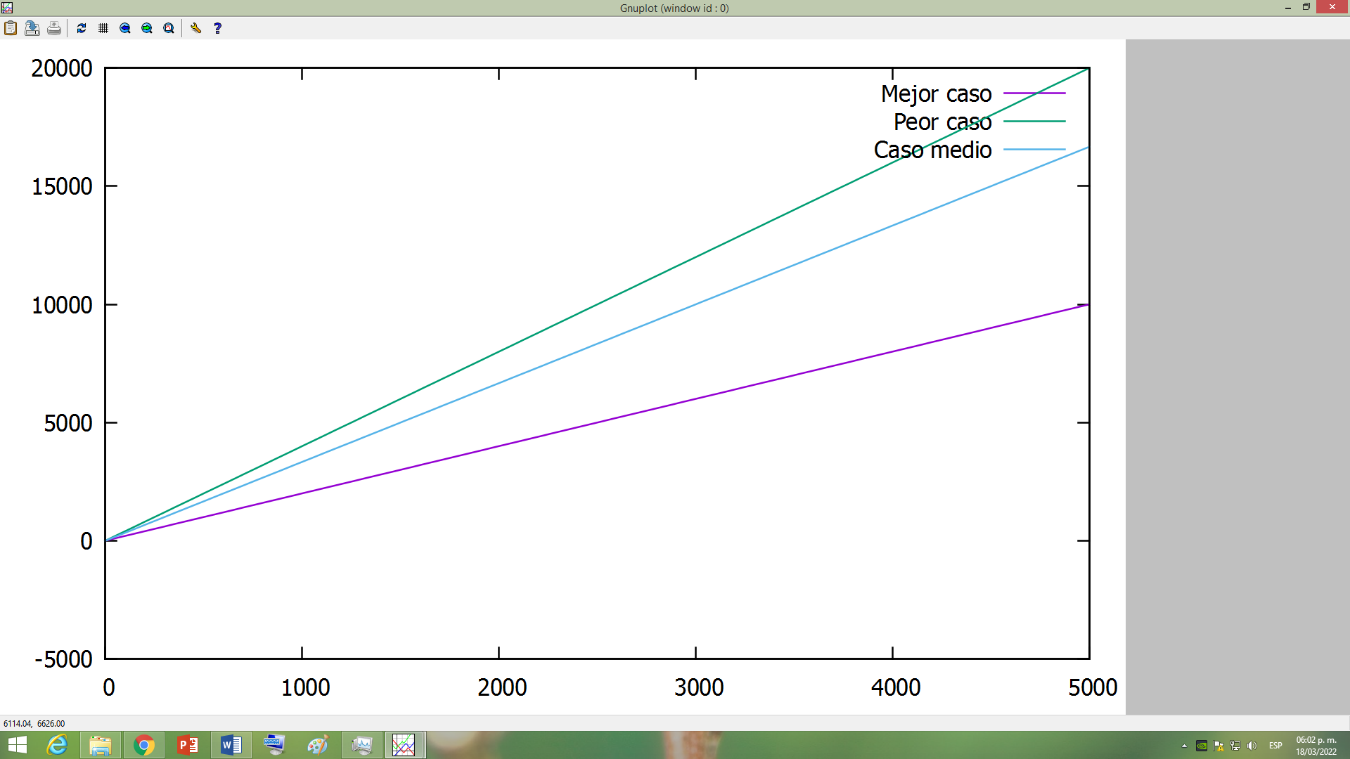
**m3**=**A**[i];

}

i++;

}

Para esta sección de código las complejidades temporales para el caso medio, mejor caso y peor caso quedan de la siguiente forma:



Ejercicio 3

func **SumaCuad3Mv2**(**A**,**n**){

for(i=0;i<3;i++){

for(j=0;j<n-1-i;j++){

if(**A**[j]>**A**[j+1]){

**aux**= **A**[j];

**A**[j]= **A**[j+1];

**A**[j+1]=**aux**;

}

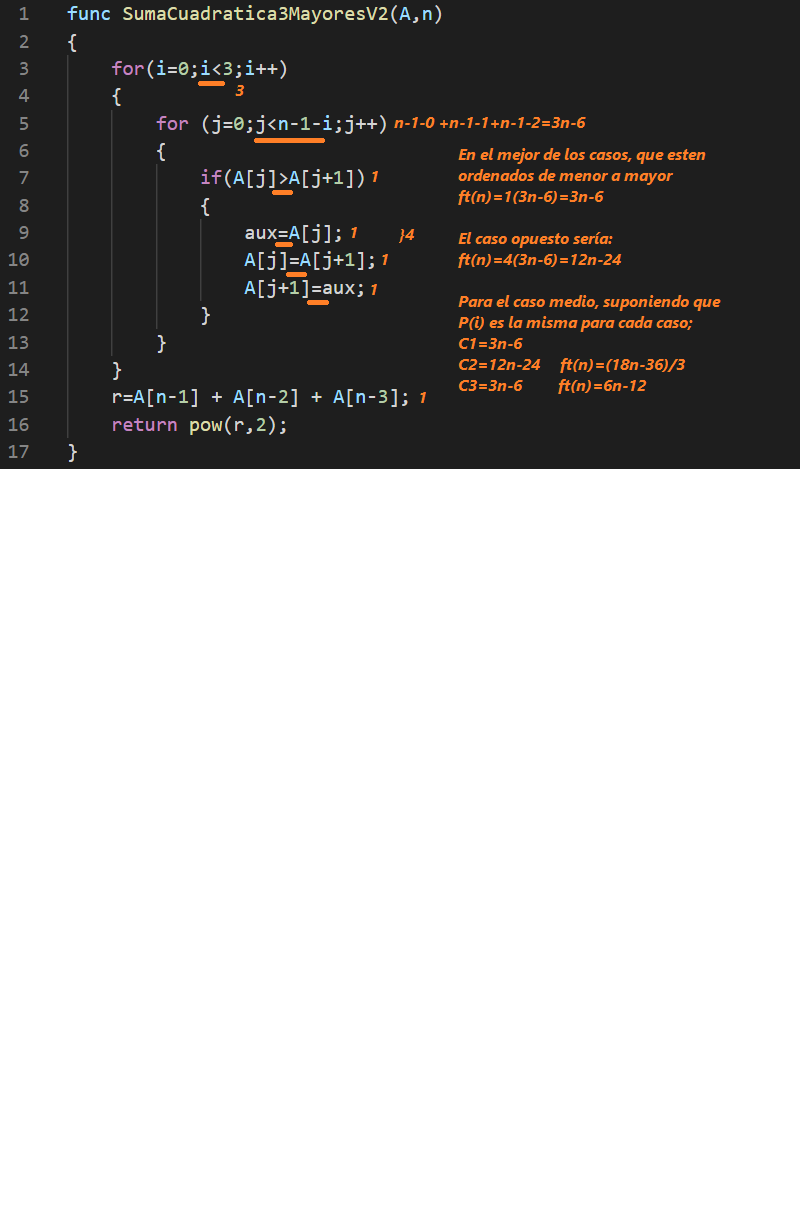
}

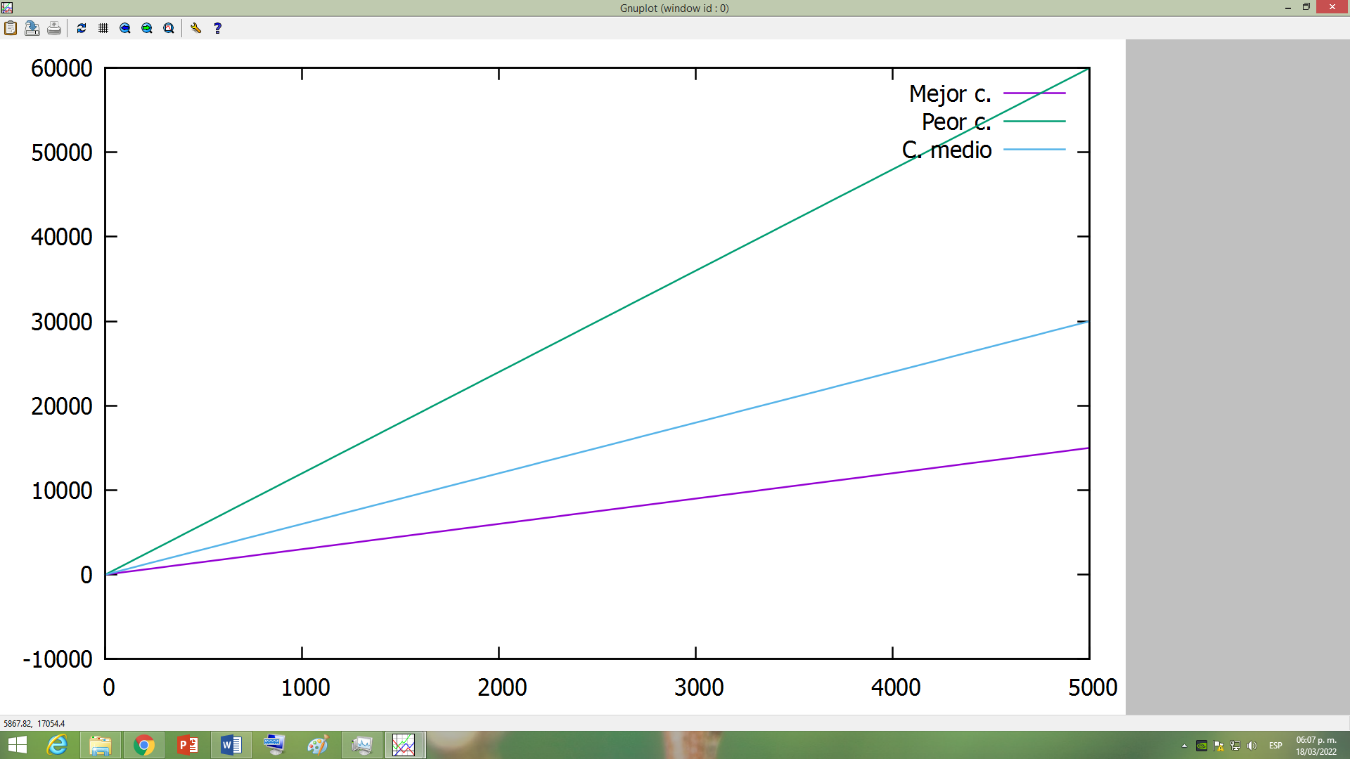
}

…..

}

Para este código, que es la versión 2 del primero la complejidad se obtendría de la siguiente forma:





Ejercicio 4:

Mientras i<=n

**ntemp**=**A**[i]

j=1

**ftemp**=0

Mientras j<=n

Si **ntemp**=**A**[j]

**ftemp**=**ftemp**+1

FinSi

j++

FinMientras

Si **f**<**ftemp**

**f**=**ftemp**

**num**=**ntemp**

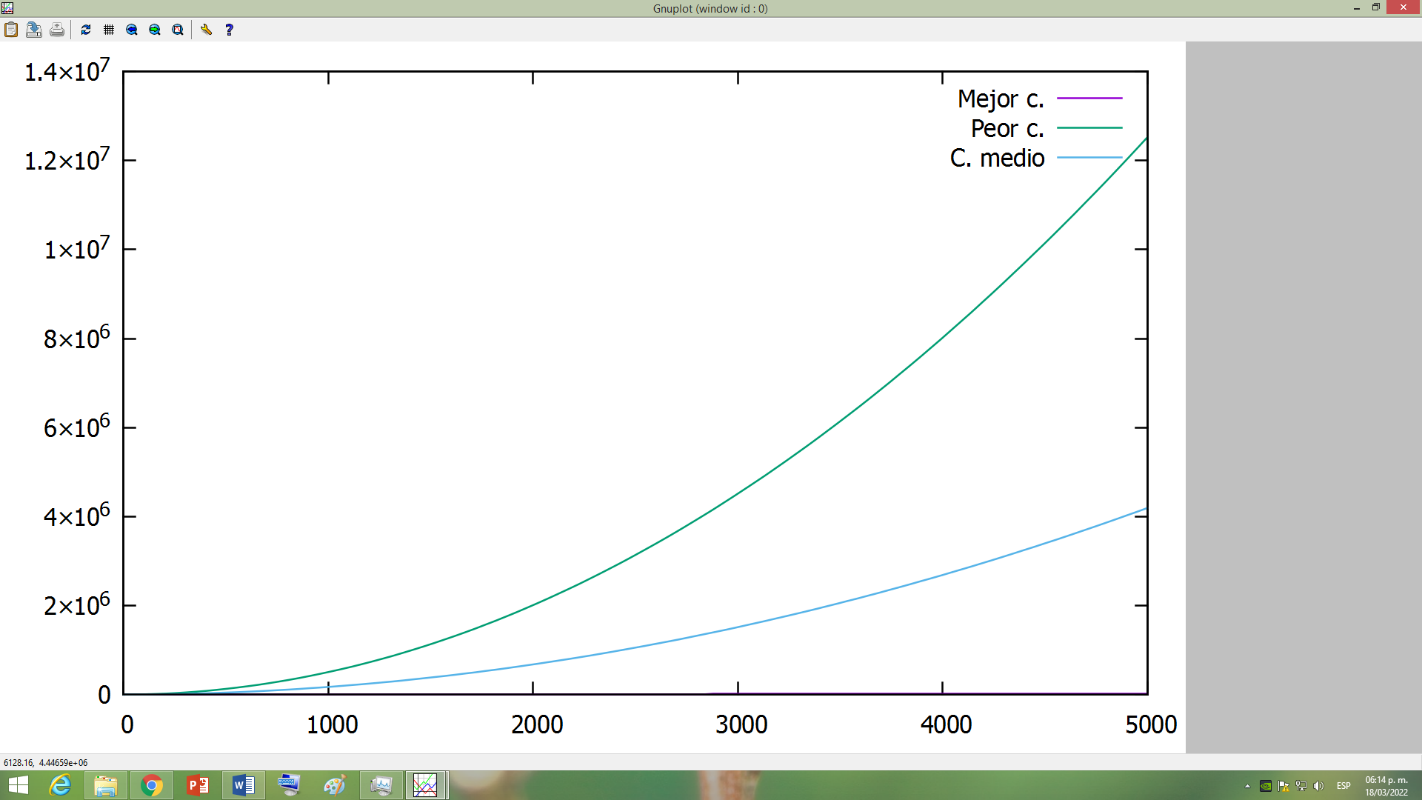
FinSi

i++

FinMientras

En este pseudocódigo hay 3 ciclos while de los cuales dos están anidados, el primero es el que llena el arreglo con los números y los otros dos los que se usan para buscar los números que más se repiten. La complejidad quedaría de la siguiente manera:



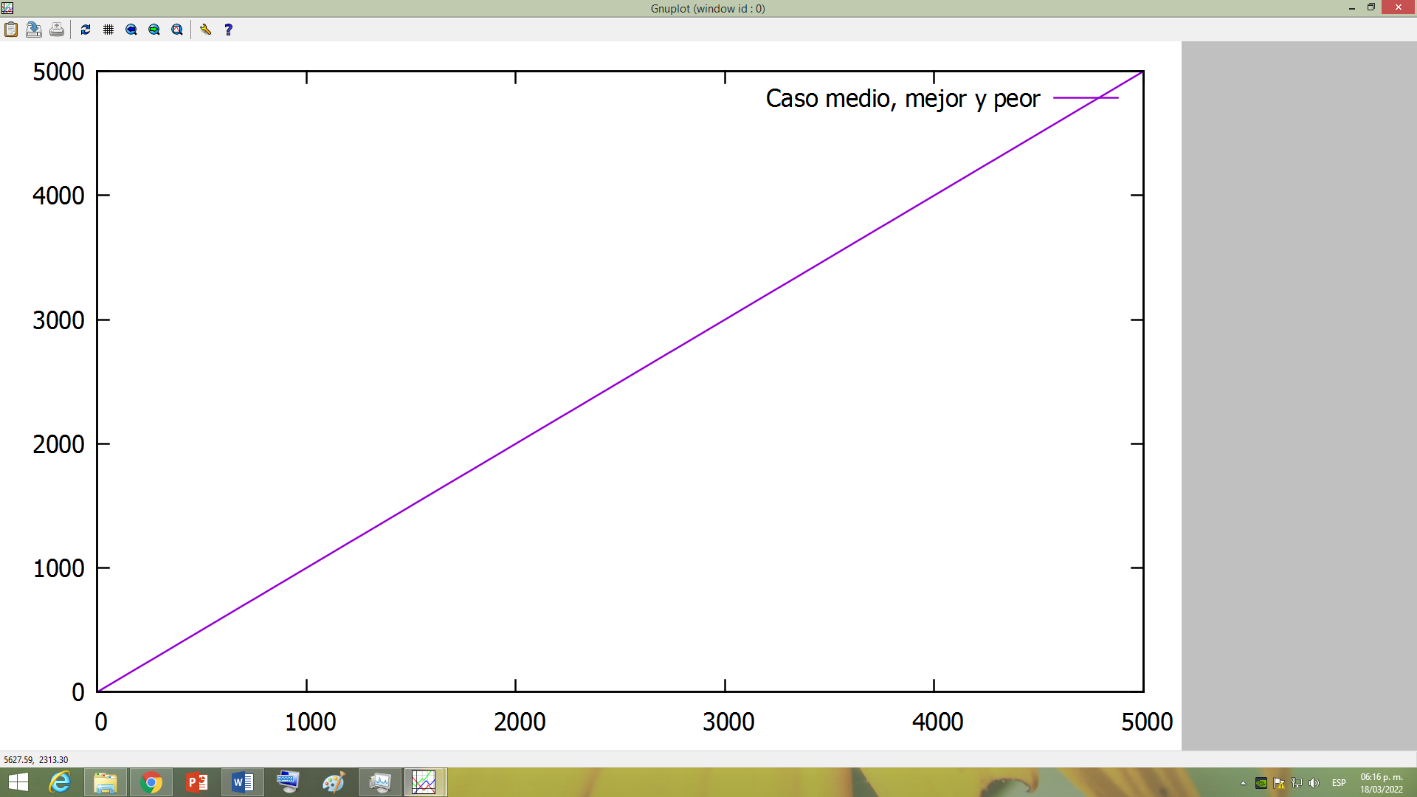


Ejercicio 5:

for(i=0;i<3;i++){

**polinom**=**polinom**\***z**+**A**[n-i];

}

Este código es lineal por lo que puedo ver, para resolver polinomios.