

## Semilla

Alumno: Carlos Gabriel Morales Umasi

Curso: Algebra Abstracta

Codigo-explicación-github

---

```
#include <iostream>
#include <windows.h>
#include <stdio.h>
#include<time.h>
#include<chrono>
#include<ctime>
#pragma comment(lib, "user32.lib")
using namespace std;

int main()
{
    SYSTEM_INFO siSysInfo;

    GetSystemInfo(&siSysInfo); //funcion sacada de internet para datos del sistema
    cout << "INFORMACION DE HARDWARE: \n";
    cout << "  OEM ID: " << siSysInfo.dwOemId << endl;
    cout << "  Numero de procesadores: " << siSysInfo.dwNumberOfProcessors << endl;
    cout << "  Tamano de pagina: " << siSysInfo.dwPageSize << endl;
    cout << "  Tipo de procesador: " << siSysInfo.dwProcessorType << endl;
    cout << "  Minimum application address: " <<
siSysInfo.lpMinimumApplicationAddress << endl;
    cout << "  Maximum application address: " <<
siSysInfo.lpMaximumApplicationAddress << endl;
    cout << "  Active processor mask: " << siSysInfo.dwActiveProcessorMask << endl;
    unsigned tini, tfin;
    float a = 0.5;
    tini = clock(); //iniciando el tiempo
    tfin = clock(); //fin de tiempo
    for (a; a < 1000000000;)
    {
        a *= 1.25;
    } //incrementamos el valor de a hasta que sea menor al parametro(10 digitos)
    //utilizando funciones para calcular el tiempo
    double time = (double(tfin - tini) / 1000000);
    auto start = chrono::high_resolution_clock::now(); //tomamos el tiempo
    auto finish = chrono::high_resolution_clock::now(); //tomamos el tiempo
    cout << chrono::duration_cast<chrono::nanoseconds>(finish - start).count() << "
microsegundos"; //tomando en cuenta 1 microsegundo=1e-6
}
//funciones de clock son sacadas de google para ayudarme en hallar tiempos
```

---

En el código se genera números random respecto al tiempo de los valores, el cual hago variar con un  $a=0.5$  dando así un inicio y multiplicarlo hasta que sea menor al valor asignado (1000000000) llamando y usando funciones buscadas desde internet ya que no tengo conocimiento de ellas, pero si se pudo recompilar datos e información sobre la librería `<ctime>`, `<time.h>`, `<chrono>` y `<Windows.h>`, donde esta última llamando a `GetSystemInfo`, compila los datos de la computadora con las distintas funciones ya escritas en el código mostrado; proponiendo número random con `<chrono>` que imprime datos en nanosegundos (con la fórmula ya escrita en código) y con `high_resolution_clock` que cumple la función de dar mediciones de duración de inicio y final; es importante resaltar que en se pone la función en nanosegundos para su implementación en código ya que da números de 3 dígitos, mientras que, por ejemplo, milisegundos imprime de 1 dígito.

Como otra propuesta, se podría agrandar la cantidad de dígitos, “juntando” los de la primera compilación con los de las  $n$  compilaciones que uno quiera, esto ya se podría ver con ZZ ya que serían mayor o igual a 6 dígitos.

---

Github: <https://github.com/CarlosGabrielMoralesUmasi/ALGEBRA-ABSTRACTA-PARTE2>