

INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE INGENIERÍA ELÉCTRICA Y MECÁNICA UNIDAD  
ZACATENCO

PROGRAMACIÓN ORIENTADA A OBJETOS

REPORTE DE PRÁCTICAS NO.3

CARLOS ALBERTO HERRERA RANGEL

2AM2

# TABLA DE CONTENIDOS

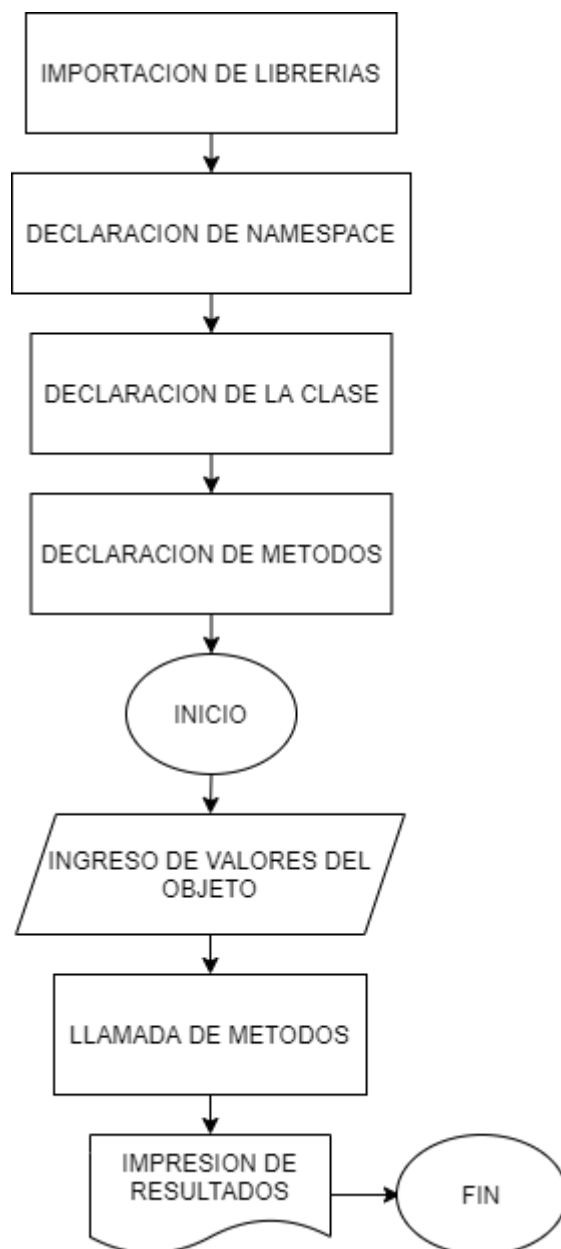
- TABLA DE CONTENIDOS
- INTRODUCCIÓN
- DESARROLLO
  - ALGORITMO
  - PSEUDOCODIGO
  - CODIGO
  - EJECUTABLE
- CONCLUSIONES
- BIBLIOGRAFIA

# INTRODUCCIÓN

En esta practica se realizara la creacion de una clase, con un constructor y un destructor, estos sirven para crear un objeto de una clase y cuando este ya no se ocupa liberar el espacio ocupado en memoria

# DESARROLLO

## ALGORITMO



## PSEUDOCODIGO

```
incluir <iostream>
incluir <string>
usar espacioNombre std;
clase ejemplo{
    cadena *ptr;
publico:
    ejemplo():ptr(nueva cadena){}
    ejemplo(constante cadena& str):ptr(nueva cadena(str)){}
    ~ejemplo(){eliminar ptr;}
    constante cadena& contenido(){retornar *ptr;}
};
entero principal(){
    ejemplo foo("foo");
    ejemplo bar("bar");
    salida<<"foo contenido: "<<foo.contenido()<<finLinea;
    salida<<"bar contenido: "<<bar.contenido()<<finLinea;
    retornar 0;
}
```

## CODIGO

```
#include <iostream>
#include <string>
using namespace std;
class ejemplo{
    string *ptr;
public:
    ejemplo():ptr(new string){}
    ejemplo(const string& str):ptr(new string(str)){}
    ~ejemplo(){delete ptr;}
    const string& contenido(){return *ptr;}
};
int main(){
    ejemplo foo("foo");
    ejemplo bar("bar");
    cout<<"foo contenido: "<<foo.contenido()<<endl;
    cout<<"bar contenido: "<<bar.contenido()<<endl;
    return 0;
}
```

## EJECUTABLE

```
$ cd "c:\Users\xXC4R\Desktop\Programacion\1er_parcial\03-02-22\" ; if ($?) { g++  
main.cpp -o main } ; if ($?) { .\main }  
foo contenido: foo  
bar contenido: bar
```

## CONCLUSIONES

En esta practica aprendimos a crear una clase con un constructor y destructor para eficientar el codigo



## BIBLIOGRAFIA

- **Ejercicios de clase**