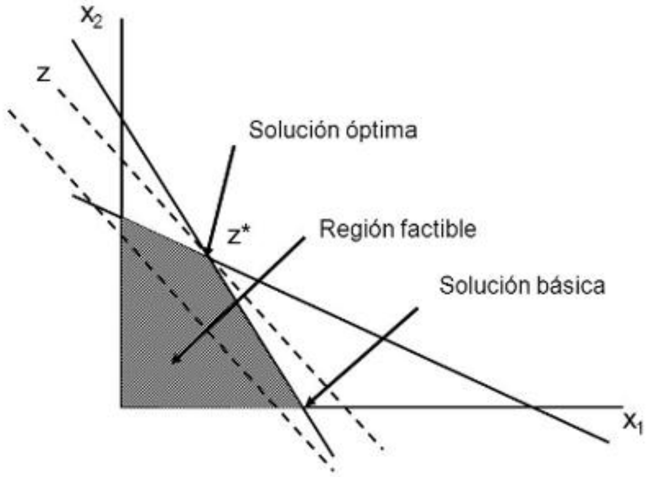
# PROBLEMA DE OPTIMIZACIÓN





**Trabajo realizado por: Ángel Conde Díaz-Dávila y Alejandro Díaz García**
**2-BACH-B**

## Índice:

# 1.- Introducción.

En esta práctica hemos desarrollado los conocimientos adquiridos sobre PYTHON que es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.

## 2.Enunciado del problema.

Se quiere construir un depósito de chapa abierto superiormente con forma de prisma recto de base cuadrada, de 1000m3 de capacidad, lo más económico posible. Sabiendo que: El coste de la chapa usada para los laterales es de 100 euros el metro cuadrado El coste de la chapa usada para la base es de 200 euros el metro cuadrado ¿Qué dimensiones debe tener el depósito? ¿Cuál es el precio de dicho depósito?

## 3.- Resolución del problema

La ecuación fx es la ecuación coste deducida de la ecuación volumen, ya que V=1000m**3
Llamando 'x' al lado de la base e 'y' a la altura, la ecuaciuón del volumen es la siguiente:
V(x,y)= x*2 y=1000 #aquí deducimos la 'y' para reducir la ecuación a una sola variable(x)
fy=1000/(x**2)
Lamamos fx a la ecuación coste, quedando así una vez que hemos puesto la ecuación con una sola variable:
f(x) =((200*x**2) +( 400000/x)).

In [33]:

1. REPRESENTAMOS LA FUNCIÓN COSTE

```
from matplotlib.pyplot import *
from numpy import *
from pandas import *
def f(x):
    return ((2*x**2) +( 4000/x))
def d(x):
    return 4*x - 4000/x**2
grid()
ylabel('f(x)')
xlabel('x')
title(" $f(x)=((200*x**2) +( 400000/x)) y su derivada")
x1= arange(-100, 100, 4)
plot(x1, f(x1))
plot(x1, d(x1))

show()
```

```
C:\Users\Alumno_14\Desktop\WPy-3701\python-3.7.0.amd64\lib\site-packag
es\ipykernel_launcher.py:8: RuntimeWarning: divide by zero encountered
in true_divide
```

```
C:\Users\Alumno_14\Desktop\WPy-3701\python-3.7.0.amd64\lib\site-packag
es\ipykernel_launcher.py:10: RuntimeWarning: divide by zero encountere
d in true_divide
  # Remove the CWD from sys.path while we load stuff.
```

In [3]:

```python
from sympy import *

x = Symbol('x')
fx =((200*x**2) +( 400000/x))
dx= diff(fx, x) # derivamos la ecuación coste
dx   #representamos el valor de dx
```

Out[3]:

```
400*x - 400000/x**2
```

In [4]:

```python
simplify(dx)
```

simplificamos el resultado cuando hay varios bloques de paréntesis

Out[4]:

```
400*x - 400000/x**2
```

In [5]:

```python
simplify(diff(dx,x))  #calculamos la segunda derivada
```

Out[5]:

```
400 + 800000/x**3
```

In [8]:

```python
solve(400*x - 400000/x**2)
```

Out[8]:

```
[10, -5 - 5*sqrt(3)*I, -5 + 5*sqrt(3)*I]
```

Como x=10, igualamos las segunda derivada con x=10 para comprobar que sea un mínimo.

In [11]:

```python
(diff(dx,x)).subs(x, 10)  # valor numérico de una derivada con x=10
```

Out[11]:

```
1200
```

Como la derivada segunda es mayor a 0, decimos que hay un mínimo cuando x=10

In [15]:

```python
from sympy import *

x = Symbol('x')
fy =1000/(x**2)
```

In [16]:

```python
(fy).subs(x, 10)
```

Out[16]:

```
10
```

In [ ]:

```
Por lo tanto, el lado es 10m y la altura es 10m, para que tenga el cos
t6e mínimo
```

In [12]:

```
(fx).subs(x, 10)
```

Out[12]:

```
60000
```

Por tanto el coste mínimo para que tenga 1000m**3 es de 60000 euros

# 4.-Concusión

.

Nos ha parecido dificultoso realizar este problema de optimización con phyton, pero nos ha servido para conocer mejor el programa. Hemos encontrado este problema resulto en internet pero no estaba bien del todo y por ello ha sido tan dificultoso.