

Universidad Central de Venezuela

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

Cálculo Raíces de Ecuaciones de Segundo Grado

Periodo 2020-3

16/11/2020

Prof. Gilberto R. Noguera

Alumno: José Páez C.I.: 24 311 351

Mediante uso de Resolvente

Para una ecuación cuadrática con coeficientes reales o complejos existen siempre dos soluciones, no necesariamente distintas, llamadas raíces, que pueden ser reales o complejas (si los coeficientes son reales y existen dos soluciones no reales, entonces deben ser complejas conjugadas). Fórmula general para la

obtención de raíces: $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

A continuación se presenta el código en python para resolver ecuaciones de segundo grado, dado los coeficientes de la ecuación, a,b,c:

```
In [7]: import math
import cmath
# a,b,c, en ese orden, son alojados en la variable "coeficientes" en fo
rma de tupla.
coeficientes= (5,20,8)
#Creo función que será la resolvente y arrojará las raíces de la ecuaci
ón.
def Resolvente(x):
    #Ubico la entrada de la variable coeficiente (tupla) en valores a,
b, c.
    a = x[0]
    b = x[1]
    c = x[2]
    #Creo condición donde si la ecuación es de grado menor (a=0) arrojé
el programa un mensaje comunicandolo-
    if a != 0:
        #Creo condición donde si el discriminante es negativo use funci
ones de raíz cuadrada en numeros complejos
        if (b*b - 4*a*c) >= 0 :
            #Resolvente para raíces enteras
            raiz1 = (-b + (math.sqrt(b*b - 4*a*c)))/(2*a)
            raiz2 = (-b - (math.sqrt(b*b - 4*a*c)))/(2*a)
            return raiz1, raiz2
        #Resolvente ppara raíces imaginarias
        else :
            raiz1 = (-b + (cmath.sqrt(b*b - 4*a*c)))/(2*a)
            raiz2 = (-b - (cmath.sqrt(b*b - 4*a*c)))/(2*a)
            return "Tiene raíces complejas:", raiz1, raiz2
    else :
        print ("No es Ecuación de segundo Grado")
print (Resolvente(coeficientes) )

(-0.4508066615170332, -3.5491933384829664)
```