Case 1 example:

s(n+2) = as(n+1) + bs(n) + cn + d

Enter values of recurence relation

a = 2

b = 3

c = 1

d = 1

Real component of s0: 0

Complex component of s0: 1

Real component of s1: 1

Complex component of s1: 0

The generating function is:

(̲(̲1̲-̲2̲j̲)̲x̲^̲3̲ ̲+̲ ̲(̲-̲1̲+̲5̲j̲)̲x̲^̲2̲ ̲+̲ ̲(̲1̲-̲4̲j̲)̲x̲ ̲+̲ ̲1̲j̲)

(1 - 3.000x)(1 - -1.000x)(1 - x)^2

sn = (0.438 + 0.250j)(3.000)^n + (-0.188 + 0.750j)(-1.000)^n + -0.250 + -0.250n

s0 = 1.000j

s1 = 1.000

s2 = (3.000 + 3.000j)

s3 = (11.000 + 6.000j)

s4 = (34.000 + 21.000j)

s5 = (105.000 + 60.000j)

s6 = (317.000 + 183.000j)

s7 = (955.000 + 546.000j)

s8 = (2868.000 + 1641.000j)

s9 = (8609.000 + 4920.000j)

Enter any value of n you want to check. To stop, enter a negative integer.

n = 21

s21=(4576404521.000 + 2615088300.000j)

n = 14

s14=(2092545.000 + 1195743.000j)

n = -1

>>>

== RESTART: C:\Users\winte\OneDrive\Desktop\CSCI2202\Math2113 Final Project.py =

Case 2 example:

s(n+2) = as(n+1) + bs(n) + cn + d

Enter values of recurence relation

a = 4

b = -4

c = 2

d = 1

Real component of s0: 0

Complex component of s0: 1

Real component of s1: 1

Complex component of s1: 0

The generating function is:

(̲(̲2̲-̲4̲j̲)̲x̲^̲3̲ ̲+̲ ̲(̲-̲1̲+̲9̲j̲)̲x̲^̲2̲ ̲+̲ ̲(̲1̲-̲6̲j̲)̲x̲ ̲+̲ ̲1̲j̲)

(1 - 2.000x)(1 - 2.000x)(1 - x)^2

sn = ((-5.000 + 1.000j))(2.000)^n + ((2.000 + -1.000j))(n)(2.000)^n + 5.000 + 2.000n

s0 = 1.000j

s1 = 1.000

s2 = (5.000 + -4.000j)

s3 = (19.000 + -16.000j)

s4 = (61.000 + -48.000j)

s5 = (175.000 + -128.000j)

s6 = (465.000 + -320.000j)

s7 = (1171.000 + -768.000j)

s8 = (2837.000 + -1792.000j)

s9 = (6679.000 + -4096.000j)

Enter any value of n you want to check. To stop, enter a negative integer.

n = 12

s12 = (77853.000 + -45056.000j)

n = 21

s21 = (77594671.000 + -41943040.000j)

n = -1

>>>

== RESTART: C:\Users\winte\OneDrive\Desktop\CSCI2202\Math2113 Final Project.py =

Case 3 example:

s(n+2) = as(n+1) + bs(n) + cn + d

Enter values of recurence relation

a = 4

b = -3

c = 1

d = -2

Real component of s0: 2

Complex component of s0: 1

Real component of s1: 0

Complex component of s1: -1

The generating function is:

(̲(̲-̲5̲-̲5̲j̲)̲x̲^̲3̲ ̲+̲ ̲(̲1̲6̲+̲1̲1̲j̲)̲x̲^̲2̲ ̲+̲ ̲(̲-̲1̲2̲-̲7̲j̲)̲x̲ ̲+̲ ̲(̲2̲+̲1̲j̲)̲)

(1 - 3.000x)(1 - 1.000x)(1 - x)^2

sn = (-1.375 + -1.000j)(3.000)^n + (3.375 + 2.000j) + 1.000n + -0.250n^2

s0 = (2.000 + 1.000j)

s1 = -1.000j

s2 = (-8.000 + -7.000j)

s3 = (-33.000 + -25.000j)

s4 = (-108.000 + -79.000j)

s5 = (-332.000 + -241.000j)

s6 = (-1002.000 + -727.000j)

s7 = (-3009.000 + -2185.000j)

s8 = (-9026.000 + -6559.000j)

s9 = (-27072.000 + -19681.000j)

Enter any value of n you want to check. To stop, enter a negative integer.

n = 12

s12 = (-730752.000 + -531439.000j)

n = 19

s19 = (-1598109585.000 + -1162261465.000j)

n = -1

>>>

== RESTART: C:\Users\winte\OneDrive\Desktop\CSCI2202\Math2113 Final Project.py =

Case 4 example:

s(n+2) = as(n+1) + bs(n) + cn + d

Enter values of recurence relation

a = 2

b = -1

c = 3

d = 1

Real component of s0: 1

Complex component of s0: 1

Real component of s1: 0

Complex component of s1: 2

The generating function is:

(̲(̲6̲+̲1̲j̲)̲x̲^̲2̲ ̲+̲ ̲(̲-̲4̲-̲2̲j̲)̲x̲ ̲+̲ ̲(̲1̲+̲1̲j̲)̲)

(1 - 1.000x)(1 - 1.000x)(1 - x)^2

sn = (1.000 + 1.000j) + (-0.500 + 1.000j)n + -1.000n^2 + 0.500n^3

s0 = (1.000 + 1.000j)

s1 = 2.000j

s2 = 3.000j

s3 = (4.000 + 4.000j)

s4 = (15.000 + 5.000j)

s5 = (36.000 + 6.000j)

s6 = (70.000 + 7.000j)

s7 = (120.000 + 8.000j)

s8 = (189.000 + 9.000j)

s9 = (280.000 + 10.000j)

Enter any value of n you want to check. To stop, enter a negative integer.

n = 13

s13 = (924.000 + 14.000j)

n = 22

s22 = (4830.000 + 23.000j)

n = -1

>>>