--- Cheybesev Method ---

1.quadratic.c

f(x) = x^2 + 5x + 2

Input: guess = 4
Ouput: root = -0.438447

Input: guess = -197
Ouput: root = -4.561553

2.cos.c

f(x) = 3x - cosx - 1

Input: guess = 5
Output: root = 0.607102

3.log.c

f(x) = xlogx - 1.2

Input: guess = 152
Output: root = 2.740569

--- Newton Method ---

1.quadratic.c

f(x) = x^2 + 5x + 2

Input: guess = 165
Ouput: root = -0.438447

Input: guess = -198
Ouput: root = -4.561553

2.cos.c

f(x) = 3x - cosx - 1

Input: guess = 765
Output: root = 0.607102

3.log.c

f(x) = xlogx - 1.2

Input: guess = 236
Output: root = 2.740649

--- Regula Falsi Method ---

1.quadratic.c

f(x) = x^2 + 5x + 2

Input: a = -10, b = 10
Ouput: root = -0.438447

Input: a = -7, b = -4
Ouput: root = -4.561553

2.cos.c

f(x) = 3x - cosx - 1

```
Input: a = -10, b = 10
Output: root = 0.607102

3.log.c

f(x) = xlogx - 1.2

Input: a = 998837, b = 13
Output: root = 2.740649

--- Muller Method ---

Input in Muller method is the set of approximates to the root.

1.quadratic.c

f(x) = x^2 + 5x + 2

Input: -1, -3, -4
Ouput: root = -0.438447

Input: 1, 2, 3
Ouput: root = -4.561553

2.cos.c

f(x) = 3x - cosx - 1

Input: 1, -1, 2
Output: root = 0.607111

3.log.c

f(x) = xlogx - 1.2

Input: 2.1, 2.7, 2
Output: root = 2.740649

--- Secant Method ---

1.quadratic.c

f(x) = x^2 + 5x + 2

Input: a = -10, b = 10
Ouput: root = -0.438447

Input: a = -7, b = -4
Ouput: root = -4.561553

2.cos.c

f(x) = 3x - cosx - 1

Input: a = -10, b = 10
Output: root = 0.607102

3.log.c

f(x) = xlogx - 1.2

Input: a = 98127, b = 1836
Output: root = 2.740646
```