

```
// Root Finder ~ Newton-Rhaphson Method
// James Saslow
// 3/8/2023
```

```
// Description:
// This Code calculates 1 root of a polynomial given an initial seed via the Newton Rhaphson Method
```

```
#include <iostream>
#include <cmath>
using namespace std;
```

```
// Prototyping a function f(x)
double f(double x);
```

```
// Prototyping f'(x)
double deriv(double x);
```

```
int main(){
    double x0; // Initial Guess
    double x1; // Updated Guess
    double eps = 0.0001; // Algorithm Tolerance
```

```
    // Algorithm variables
    double y;
    double dydx;
    double diff;
```

```
    // Prompting user to pick an initial seed for x0
    cout << "Initial Seed?" << endl;
    cout << "x0 = ? ";
    cin >> x0;
```

```
    //Implimenting Newton - Rhaphson Algorithm
    do{
        y = f(x0);
        dydx = deriv(x0);
        x1 = x0 - y/dydx; // Doing the algorithm
        diff = fabs(x1 - x0);
```

```
        x0 = x1; // Updating x0
    }while(diff>eps); // Tolerance Condition
```

```
    //Outputing the found root
    cout << "x = " << x0;
```

```
    return 0;
```

```
}
```

```
// Defining f(x)
double f(double x){
```

```
    // YOU CAN WRITE YOUR FUNCTION HERE
    return (x-1)*(x+6)*(x-5);
```

```
// YOU CAN WRITE YOUR FUNCTION HERE  
}  
  
// Defining f'(x)  
double deriv(double x){  
    double dx = 0.000001;  
    return (f(x+dx) - f(x))/dx;  
}
```