```c
/*
This program is used to find the roots of an equation by using newton method. In this metho
d an initial guess X0 is taken and is iterated by the equation
                   Xn+1 = Xn - f(Xn)/df(xn-1)
Where df is the derivative of the function f(x) with respect to x.

Input : -An intital guess (X0)

Algorithm :
   A new approximation to the root can be found by using
   Xn+1 = Xn - (f(Xn)/df(Xn))

Output : - Root(s) of the equation f(x) = 0
*/


#include <stdio.h>
#include <stdlib.h>
#include <math.h>

//Function prototypes
float f(float x);
float df(float x);
float newton(float a);

int main()
{
  float root, a; // a is the initial guess
                 // root is the final root of the equation

  //Getting initial guess from the user.
  printf("Enter initial guess: ");
  scanf("%f", &a);

  root = newton(a); //Invoking the newton funtion
  printf("Root of the equation is %f\n", root);

  exit (0);
}

float newton(float a)
{
  float root, x, y;

  //Loop infinitely
  while(1)
  {
      //Iterating through the equation Xn+1 = Xn - (f(Xn)/f'(Xn))
      //Xn+1 is root
      //Xn is a
      root = a - (f(a)/df(a));

      //printf("Checking %f\n", root);

      //Checking if root is found
      if(f(root) == 0)
      {
          return root;
      }

      //Rounding a and root to 5 decimal accuracy and comparing
      x = round(root*100000)/100000;
      y = round(a*100000)/100000;

      //If they are equal, Implies the equation is converged
      if(x == y)
      {
          return root;
      }
      a = root; //Update the initial guess
  }
}

float f(float x)
{
  float ans;
//Calculating function value
  ans = x*log10f(x) - 1.2;

  //Checking if the result is a NAN
  if(ans != ans)
  {
      printf("Couldn’t proceed,..Try changing value\n");
      exit (2);
  }
  return ans;
}

float df(float x)
{
  float ans;

  //Calculating the derivative value
  ans = 1 + log10f(x);

  //Checking if the result is a NAN
  if(ans != ans)
  {
      printf("Couldn’t proceed,..Try changing value\n");
      exit (2);
  }
  return ans;
}
```