```c
//************************************************
//Program for implementation of Newton-Raphson Method
//Coded by Ashwini K. Singh on 27-Jan-2021
//************************************************

#include<stdio.h>   // Inclusion of the input-output header file
#include<math.h>    // Inclusion of the math header file

//*******EVALUATION OF THE FUNCTION
float f(float x)
{
    return ((exp(-x))-x);
}
//*******END OF FUNCTION f

//*******EVALUATION OF THE DERIVATIVE OF FUNCTION
float df (float x)
{
    return (-((exp(-x))+1));
}
//*******END OF FUNCTION df

//*************STARTING MAIN FUNCTION*********
int main(void)
{
    int i, N;                                   //Variable declaration
    float h, x0, x1, e;                         //Variable declaration

    printf("\nF(x) = ((exp(-x))-x)\n");
    printf("\ndF(x) = (-((exp(-x))+1))\n");

    FILE *input, *output;
    input=fopen("inNewtonRaphson.txt","r");
    output=fopen("outNewtonRaphson.tsv","w");

    // Reading inputs from the input file
    fscanf(input,"%f,%f,%d", &x0, &e, &N);

    // Displaying inputs read from the input file on the console
    printf("\nValues of x0, allowed error and maximum iterations read
from 'inNewtonRaphson.txt' are: %f, %f, %d\n",x0,e,N);
    // Writing the column-headers in the output file
    fprintf(output,"Itration\tx0\tx1\n");
    printf("\nItr\tx0\t\tf(x0)\t\tdf(x0)\t\th\t\tx1\n");

    for (i=1; i<=N; i++)
    {
        h=f(x0)/df(x0);
        x1=x0-h;
        fprintf(output,"%3d\t%9.6f\t%9.6f\n", i, x0, x1);    // Writing
the outputs in the output file

printf("%3d\t%9.6f\t%9.6f\t%9.6f\t%9.6f\t%9.6f\n",i,x0,f(x0),df(x0),h,x1)
;
        if (fabs(h) < e)
        {
            printf("\nAfter %3d iterations, root = %8.6f\n", i, x1);
// Displaying the final output on console
            return 0;
```

```
        }
        x0=x1;
    }
    printf("\nThe required solution does not converge or iterations are
insufficient\n");
    return 1;
}
//****************END OF MAIN FUNCTION**********
```