

```

1  //*****
2  //Program for implementation of Secant Method
3  //Coded by Ashwini K. Singh on 01-Feb-2021
4  //*****
5
6  #include<stdio.h>    // Inclusion of the input-output header file
7  #include<math.h>    // Inclusion of the math header file
8
9  //*****EVALUATION OF THE FUNCTION
10 float f(float x)
11 {
12     return ((x*x*x)-x-1);
13 }
14 //*****END OF FUNCTION f
15
16 float df(float x_, float x)
17 {
18     return (f(x)-f(x_))/(x-x_);
19 }
20 //*****END OF FUNCTION df
21
22 //*****STARTING MAIN FUNCTION*****
23 int main(void)
24 {
25     int i, N;                                //Variable declaration
26     float h, x_, x0, x1, e;                  //Variable declaration
27
28     FILE *input, *output;
29     input=fopen("inSecant.txt", "r");
30     output=fopen("outSecant.tsy", "w");
31
32     printf("\nPlease ensure x_ < x0\n");
33     // Reading inputs from the input file
34     fscanf(input, "%f,%f,%f,%d", &x_, &x0, &e, &N);
35
36     // Displaying inputs read from the input file on the console
37     printf("\nValues of x_, x0, allowed error and maximum iterations read from

```

```

38  'inSecant.txt' are: %f, %f, %f, %d\n", x_, x0, e, N);
39  // Writing the column-headers in the output file
40  fprintf(output, "Iteration\tx_\tx0\tx1\tabs(h)\n");
41
42  for (i=1; i<=N; i++)
43  {
44      h=f(x0)/df(x_, x0);
45      x1=x0-h;
46      fprintf(output, "%d\t%9.6f\t%9.6f\t%9.6f\t%9.6f\n", i, x_, x0, x1, fabs(h)); //
47      Writing the outputs in the output file
48      if (fabs(h) < e)
49      {
50          printf("\nAfter %d iterations, root = %9.6f\n", i, x1); // Displaying the
51          final output on console
52          return 0;
53      }
54      x_=x0;
55      x0=x1;
56  }
57  printf("\nThe required solution does not converge or iterations are insufficient\n");
58  return 1;
59  }
60  //*****END OF MAIN FUNCTION*****

```