A close up of a text

Description automatically generated

A black text on a white background

Description automatically generated

A white board with writing on it

Description automatically generatedA screenshot of a computer

Description automatically generated

If we use a larger terms, let’s say 40, the result:

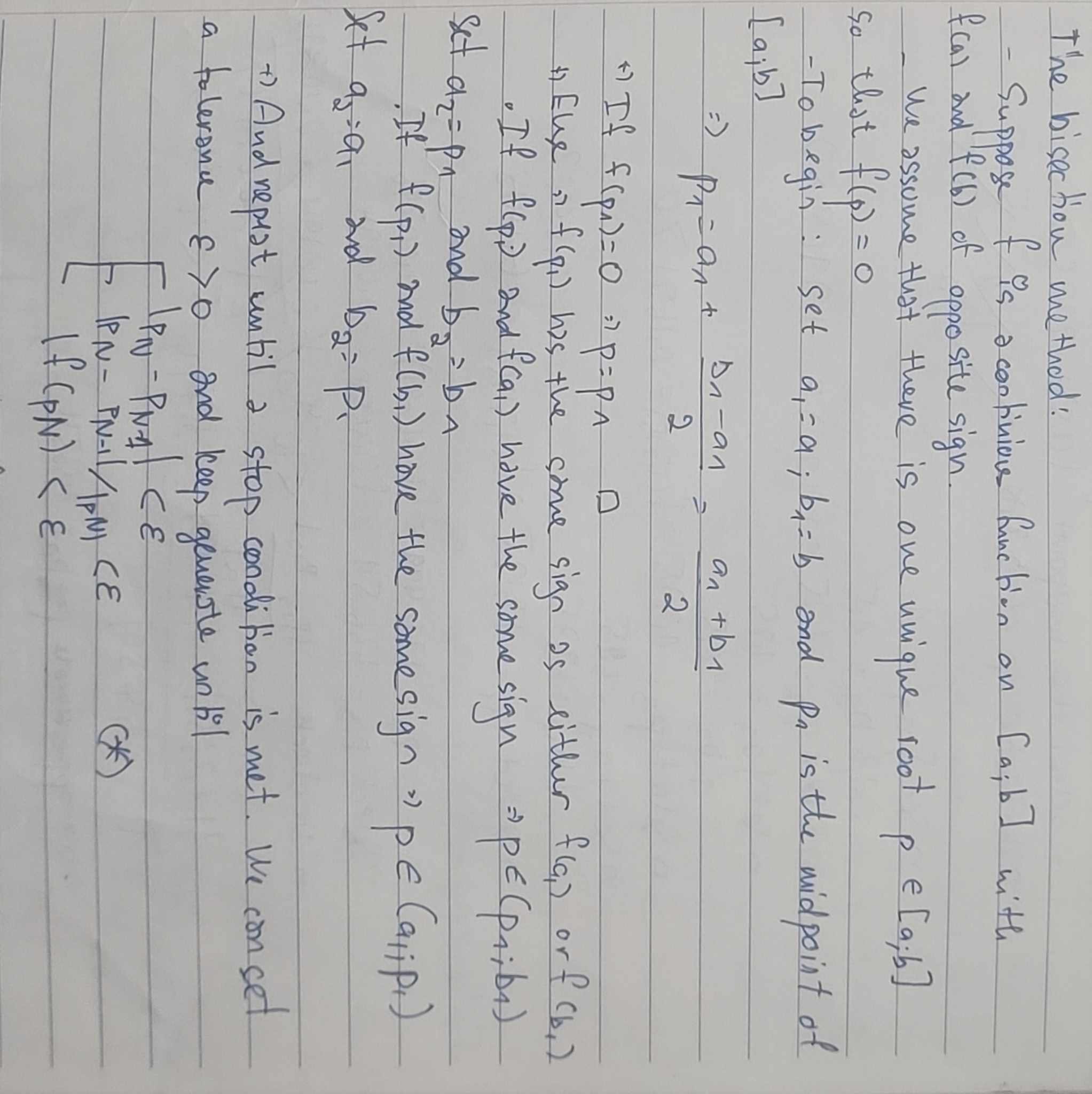
A screenshot of a computer

Description automatically generated

The results are much more accurate now.

A white background with black text

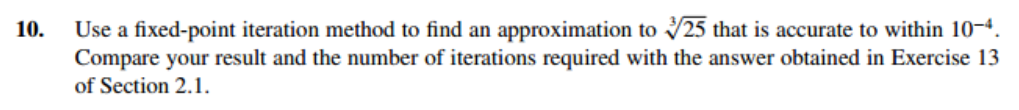
Description automatically generated



Perform the algorithm on all the given ranges, we result:

A computer screen with white text

Description automatically generated



A white board with writing on it

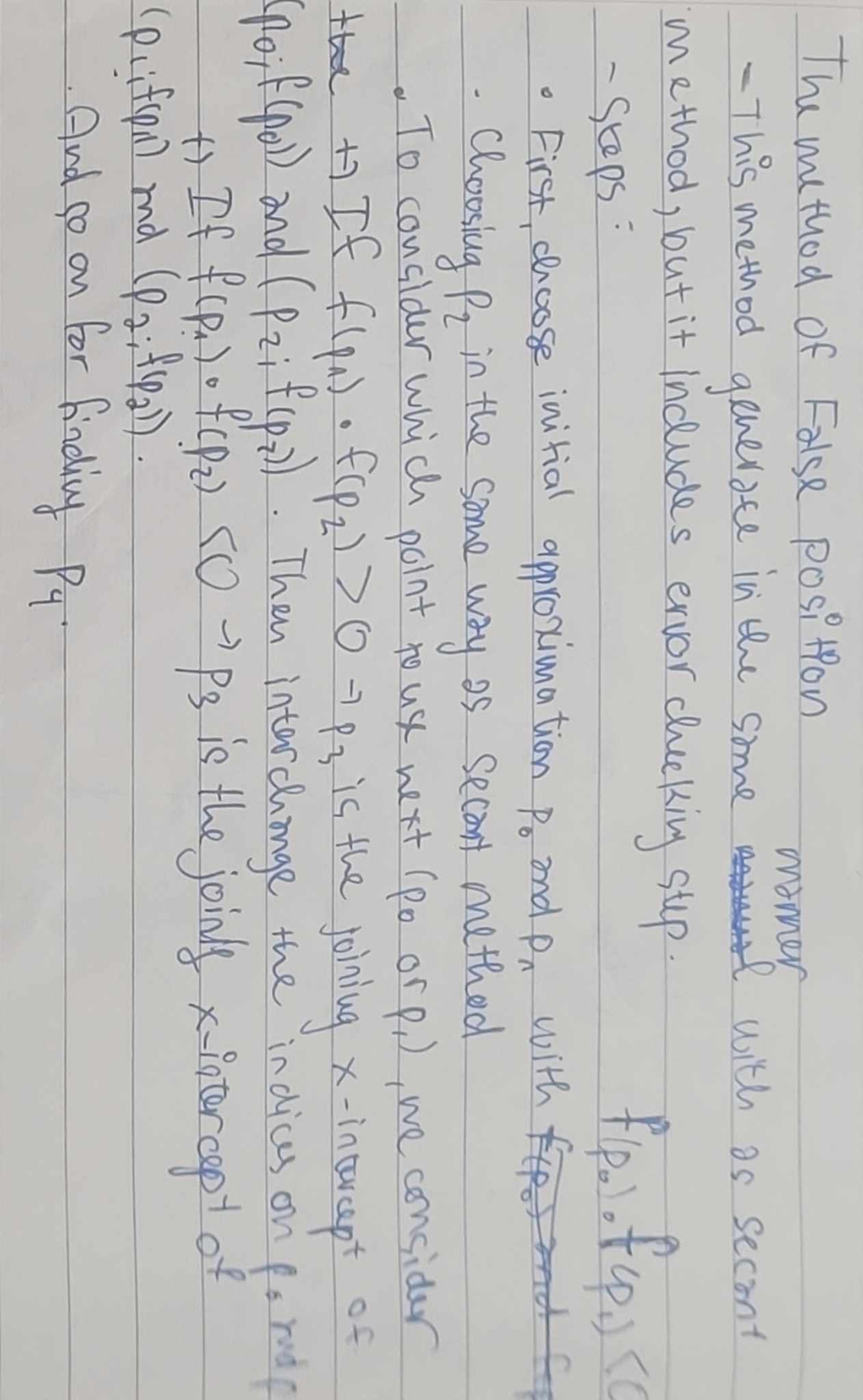
Description automatically generated

The code will randomly choose a number and start iterating till the final value:

A screenshot of a computer

Description automatically generated

A math equations on a white background

Description automatically generated

Calculating by hand:

A white board with writing on it

Description automatically generated

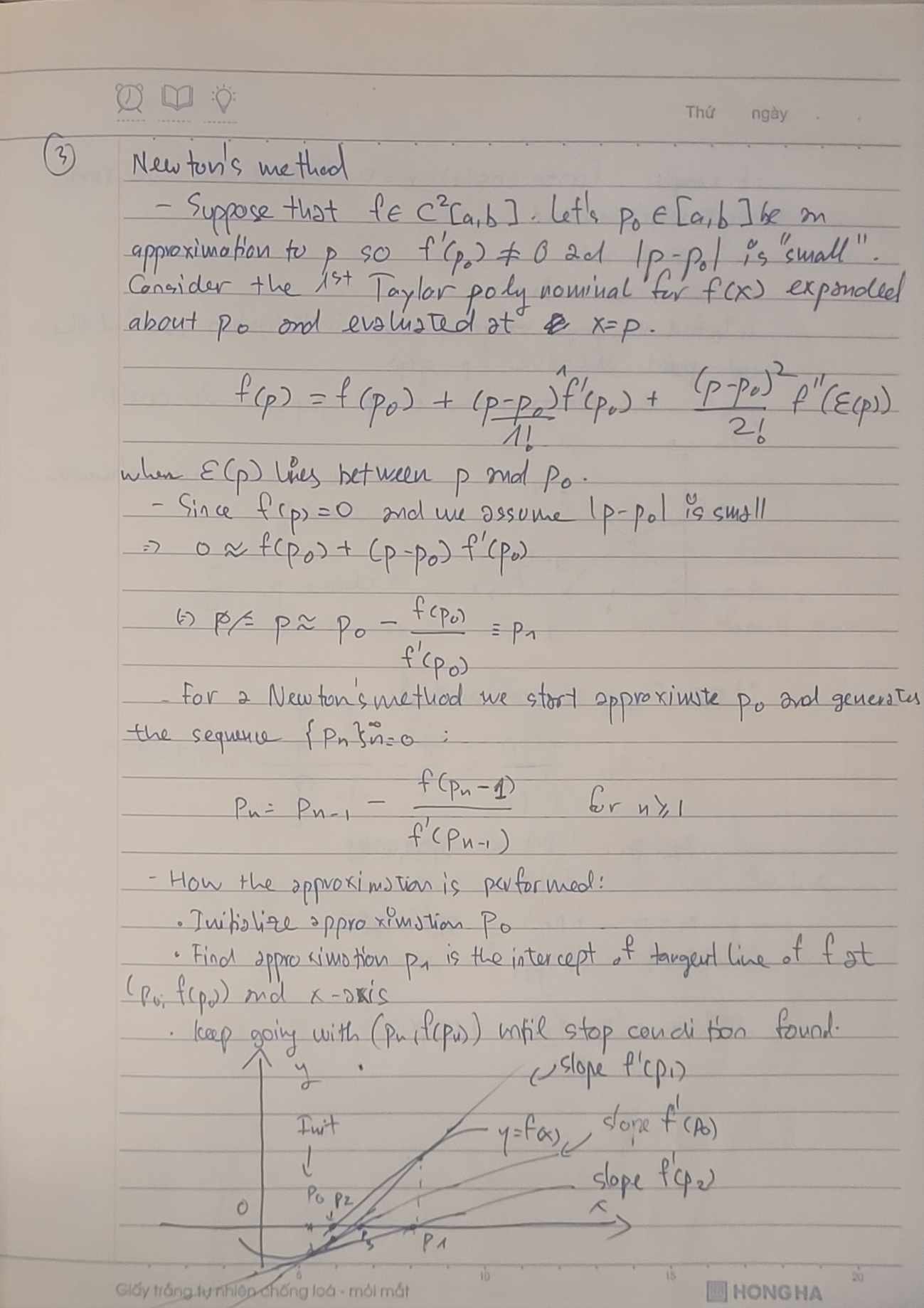
Use code:

A screenshot of a computer program

Description automatically generated

A math equations and numbers

Description automatically generated with medium confidence



Calculating by hand:

A white board with math equations

Description automatically generated

Calculating by code with some other p0:

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generatedA computer screen with white text

Description automatically generatedA screenshot of a computer program

Description automatically generated

A math equations on a white background

Description automatically generated

Let’s denote: A number and number written on a white board

Description automatically generated with medium confidence

We next write A into a Lower matrix multiply to the Transposed of that Lower matrix:

A whiteboard with writing on it

Description automatically generated

Perform calculating and result in the coefficents on the left

Finally, we calculate A = L \* L^T and L\*y = B, giving the result:

A whiteboard with blue writing

Description automatically generated

Runnign the code yeilding the same result:

A computer screen shot of a code

Description automatically generated

A screenshot of a paper

Description automatically generated

A white board with math equations and numbers

Description automatically generated

Using code, we find a = 13.146500, and b = -0.002255

A graph with blue dots

Description automatically generated

A math equations on a white background

Description automatically generated

A piece of paper with writing on it

Description automatically generated

A paper with writing on it

Description automatically generated

To determine whether to use 3 points endpoint, 3 points midpoint, or the 5 points fomula, we should look at the index of the value in the table.

End point fomula suits value at the start or end (if the case is at the end, simply lets h<0) of the table, while mid points is for the middle ones.

Applying the code on table (a), we yield:

A black background with white numbers

Description automatically generated

A screenshot of a calculator

Description automatically generated

A white board with blue writing on it

Description automatically generated

A white background with black text

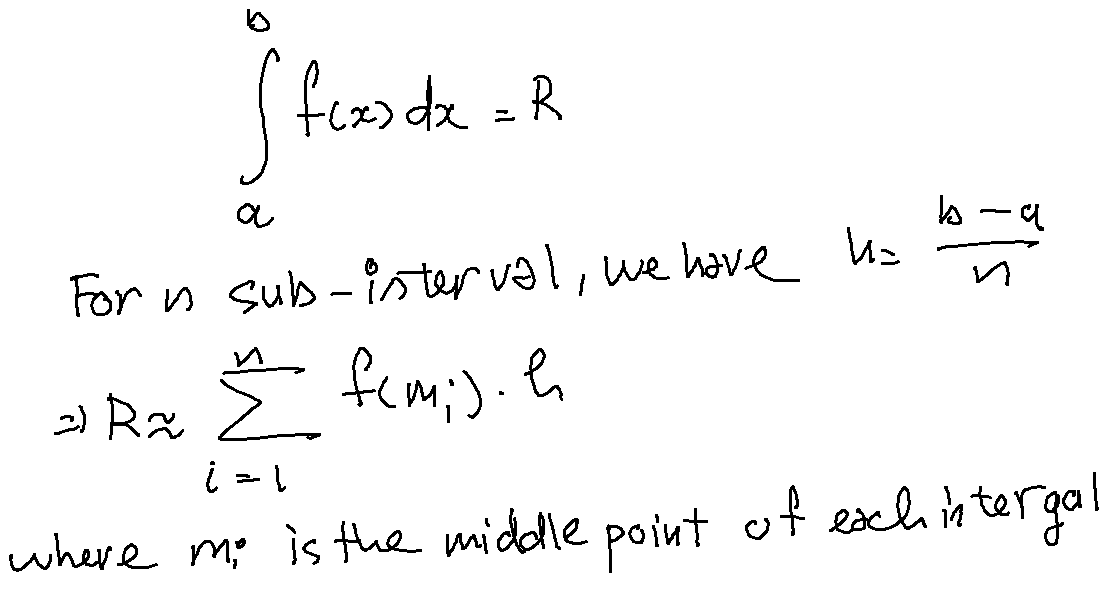
Description automatically generated

Midpoint rule (rectangular rule but we take the middle points instead of the starting ones), is a method to calculate the intergration by calculating the rectangulars’s size. The following figure illustrates how this method works.

A graph of a graph

Description automatically generated with medium confidence

The theory:



For (a) and (b) the code (n = 10) yields resspectively:

A black screen with white text

Description automatically generatedA screenshot of a black screen

Description automatically generated

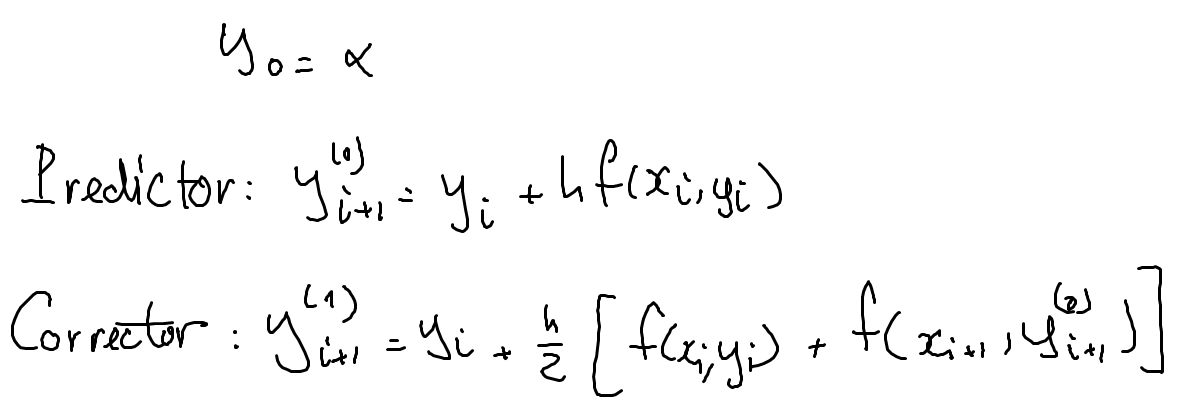
The truth value of (a) and (b) are 0.489971 and 0.0525, as we can see the errors are quite large.

A math problem with black text

Description automatically generated

Heun’s method is a method to approx the differential equations. This method will predict a value (predictor) beforehand, then use that prediction to approximate a better (corrector) value.

In general:



Applying code for exercise (a), we have:



Try with a smaller h (let’s say, h = 0.05)

A screen shot of a computer

Description automatically generated

The error doesn’t noticably decrease.

A math problem with black text

Description automatically generated with medium confidence

Euler method is defined:

A close-up of a blue handwritten text

Description automatically generated

Which is simple. Use code to solve the (a):

A screenshot of a computer program

Description automatically generated

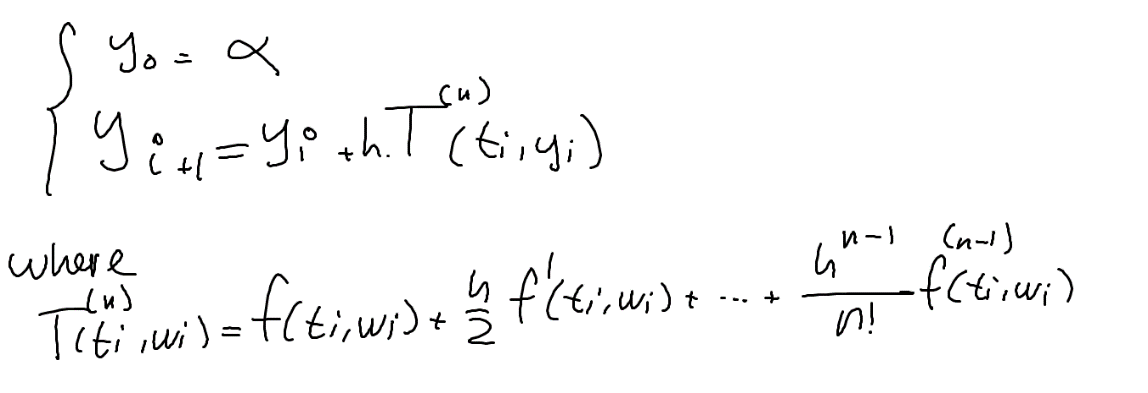
Now, apply Lagrange interpolation into the aboves data set, we have:

* f(1.052) = -0.9480945175525641
* f(1.555) = -0.6241495493175295
* f(1.978) = -0.4776453139246011

A math problem with equations

Description automatically generated with medium confidence

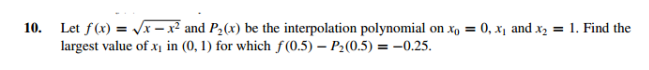
Taylor method high order is defined:

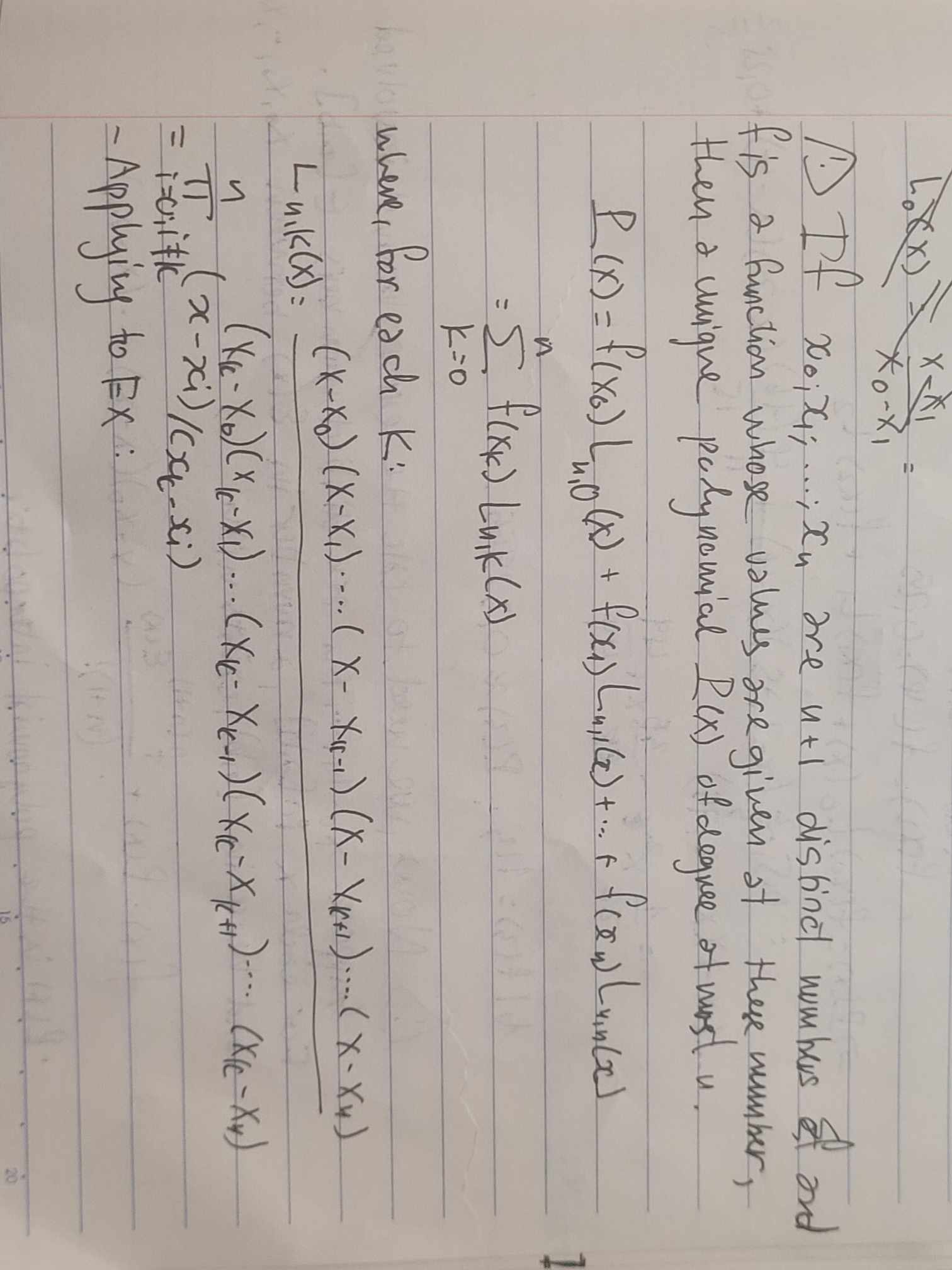


Applying into the function:

A screenshot of a computer program

Description automatically generated



To do this exercise, first we archive a Lagrange polynomial.Ththan

Next, we will do some coding trick to guess and choose the t1 value so that it is as correct as possible. After 100000 steps, the program concludes the following value.

A screenshot of a computer error

Description automatically generated

More detail about how we handle the task can be seen in the code file.