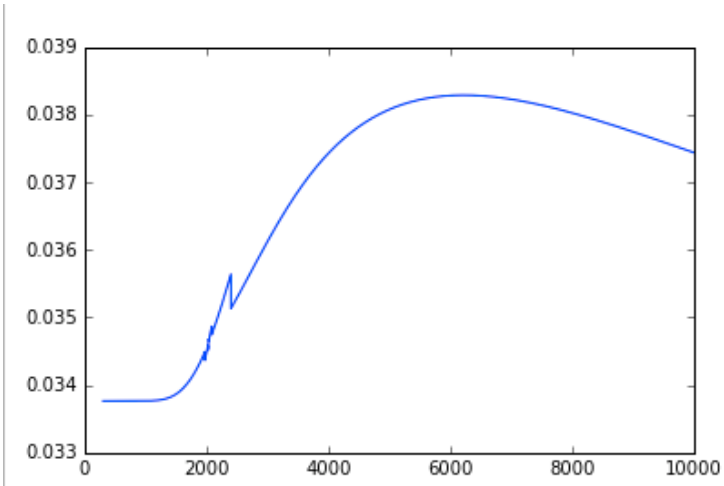
For this question, Part a) I tried to use the integration. Quadrature in scipy, but the value returned out to be 0. So I instead used the Romberg code which was written in HW 2 but it will introduce a gap image like this




```
Maximumvalue happend at 6221.637336684891 with efficency= 0.0382861948433
```

It turns out to be caused by the error turn since in some operation the integral will break out of the loop but lies far from the last integral. So I remove the error term

```
I1=Ii
for m in range(1,i+1):      #find value from last column same row and last column last row
    z[i][m]=z[i][m-1]+(z[i][m-1]-z[i-1][m-1])/(4**m-1)
    error=abs((z[i][m-1]-z[i-1][m-1])/(4**m-1))
    if error<1e-6:                  #when error reached the threshold , print last row and print the final estimate
        return z[i][m]   #terminate funtion (avoid double break)
```

and change it thus it will produce the end of triangle at column number 10. A smooth curve fitting requirement was given.

For question B, just directly use the golden ratio search method, and since from part a we can see the maxima lies near 6000, I set the starting point and the end point to be 5000 and 7000 and find the value.