Justin Santer

Professor Yuefan Deng

AMS 326

February 14, 2020

Homework 1

Problem 1.1

Problem Description:

Using the given equation, find the area between a heart-shaped curve and outside a circle of the maximum area within the heart.

$$x^2 + \left(y - \sqrt{|x|}\right)^2 = 2$$

Algorithm Description:

I first found the maximum area of the circle analytically (A = 3.350827). Next, I found the area of the heart numerically using the Monte Carlo method. I decided to use a sample size of 1 billion. Looping through the sample size, I generated two random numbers to use as coordinates. In order to be more accurate and save running time, I used the right half of the heart as my sample area. Then, I put the random x value into the given heart equation to get a lower and upper value of y. If the random value of y was between the lower and upper values, I increased a counter of "valid points," (meaning they were within the heart). After finishing the loop, I divided the valid point counter by the sample size to get proportion of points within the heart. I then multiplied this rate with the overall sample area, which was the x-range * y-range and equal to 4.923179, to get the area of half the heart. Finally, I doubled the half heart area and subtracted the max circle area to get the desired remaining area.

Results and Comments:

The run-time of the program averaged slightly over 6 minutes on my machine, but I tested it on other machines where I found it to be as fast as 2 minutes. I ran the program three times and got the average of the attempts for better accuracy.

1st attempt: 2.932107

2nd attempt: 2.932124

3rd attempt: 2.932103

The average came out to be **2.932111**, which is my final solution to the problem. As for the estimated amount of floating-point operations, I calculated that each loop would need 20 operations. After a billion loops, there would be 20 billion operations.

Problem 1.2

Problem Description:

Find the roots of the given function using a numerical method.

$$f(x) = 2.020^{-x^3} - x^3\cos(x^4) - 1.984$$

Algorithm Description:

I decided to use the secant method to calculate the roots of this function using 6 iterations. My program just followed the formula:

$$x_{n+1} = x_n - \frac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$$

All I had to do was find an initial x_n and x_{n-1} for each root I wanted to find. I analyzed the graph that was given and found there were 6 roots and got the corresponding initial x-values I needed.

Results and Comments:

This time, the program runs through within milliseconds, because there is no extreme sample size. Here are the results:

1st root using values -0.80 and -0.85: **-0.824286**

2nd root using values 1.25 and 1.30: **1.269196**

3rd root using values 1.40 and 1.45: **1.414293**

4th root using values 1.65 and 1.70: **1.695594**

5th root using values 1.80 and 1.85: **1.806723**

6th root using values 1.90 and 1.95: **1.948278**

Six iterations of the secant method will get more than 4 digits of accuracy; therefore, these are the roots within the given range of the function.

Final Note: The source code I created and used for all calculations mentioned will be included in the .zip file that this report is in. I also collaborated with classmate Gino Giacoio when discussing general algorithms to use for the problems.