Viveka Agrawal
Professor Veenstra
CSE13S, Winter 2023

CSE13S ASGN1 DESIGN DOC

Description of the program:

In this assignment, a bash script and gnuplot must be used to produce 2 plots of a provided C program. For this assignment, we are using the data of the monte_carlo.c program as points in the graphs to create Figures 2 and 3.

For Figure 2, we must graph a circle with a radius of 1 inside of a square. For this graph, we are using columns 3, 4, and 5 of the monte_carlo.c output. Columns 3 and 4 represent x and y respectively, and those points are plotted on the graph. The dots are red if column 5 equals 0 where 0 means that the point is outside the circle. The dots are blue if column 5 equals 1 where 1 means that the point is inside the circle. Here is Figure 2 that we are supposed to replicate:

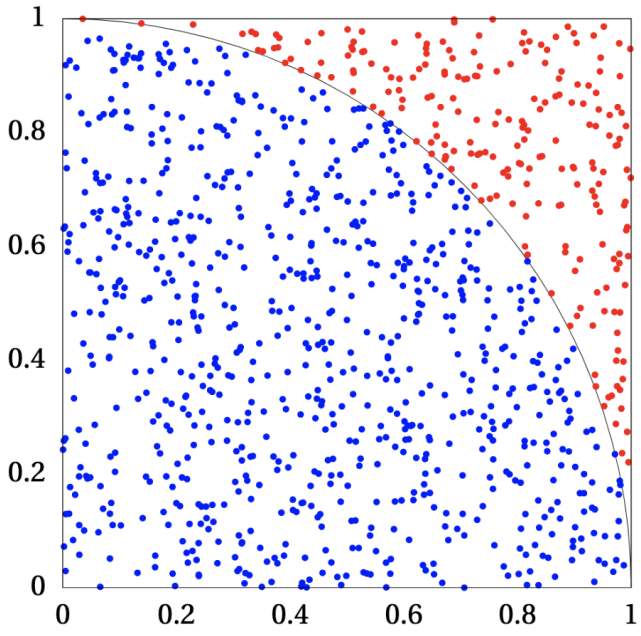

Figure 2: Blue points have distance less than or equal to 1, and hence belong both in the square and the circle, red points are a part of the square but not the circle.

For Figure 3, we must graph the difference between the estimated value of pi and pi. Initially, the difference will be large but with more iterations the difference will eventually decrease and either reach or almost reach 0. For this graph, we are using columns 1 and 2 of the monte_carlo.c output. Column 1 is the number of iterations of the monte_carlo.c program and column 2 is the estimated value of pi. Column 1 acts as the x value and column 2 minus pi acts as the y value, and those points are plotted on the graph. There needs to be 5 different lines, so 5 different monte_carlo data files must be created. The lines are also different colors to represent the different plotted points for each of the 5 different data files. Here is Figure 3 that we are supposed to replicate:

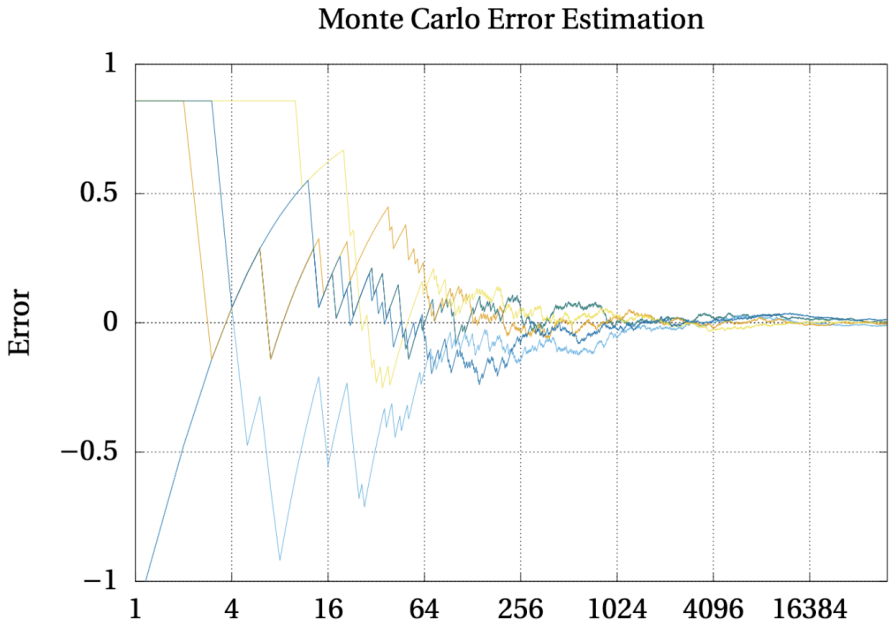

Figure 3: The value of the difference between the estimated π and pi gets closer to zero as we increase iterations. The different colors represent different seeds for the random number generator.

Files to be Pushed:

plot.sh : this is the bash script which produces the Monte Carlo method and has the plots that resemble Figures 2 and 3

monte-carlo.c : this file is provided and has the implementation of the Monte Carlo program

Makefile: this file is provided and directs the compilation process of the Monte Carlo program

README.md : uses Markdown syntax and describes how to use the script and Makefile. This also lists and explains command-line options accepted by the program

DESIGN.pdf : describes the design of the program using pseudocode (this document)

WRITEUP.pdf : includes the graphs produced from the bash script and explains which UNIX commands are used to create each plot and why they were chosen

Pseudocode:

plot.sh

Write the command that makes the system know that the program is a bash/shell script
Rebuild the monte_carlo executable using make
Place the data points from the monte_carlo executable into a data file

Figure 2
Start gnuplot
Use the command that ignores the first line in the monte_carlo data file
Produce a file in pdf form that will display the graph
Name the file Figure2.pdf
Set the title of the graph as Figure 2
Name the x axis as x
Name the y axis as y
Create a circle with the center being at the point of origin and set the radius of the circle to 1
Use the command that makes the length of the x-axis unit equal to the y-axis unit in order to make sure the circle is graphed properly
Plot the points in the 3rd and 4th columns (which are the x and y values respectively) of the monte_carlo data file and if the value in the 5th column is 0 then make the dots red and if its 1 then make the dots blue
End gnuplot

Figure 3
Create a for loop that creates 5 different data files of the monte_carlo.c program to be graphed and include the command that delays the time by 3 seconds to ensure that the data points in the 5 files are not the same
Start gnuplot
Use the command that ignores the first line in the monte_carlo data file
Produce a file in pdf form that will display the graph
Make a name for the file
Name the file Figure3.pdf

Set the title of the graph as “Monte Carlo Error Estimation”
Name the y axis as “Error”
Set the 0 axis in the center of the graph
Set the x axis to a logarithmic scale of 4
Include grid lines
Plot the values in the 1st column of the 1st generated monte_carlo data file as the x axis values and plot the values in the 2nd column of the 1st generated monte_carlo data file minus pi as the y axis values and make the line color brown
Plot the values in the 1st column of the 2nd generated monte_carlo data file as the x axis values and plot the values in the 2nd column of the 2nd generated monte_carlo data file minus pi as the y axis values and make the line color blue
Plot the values in the 1st column of the 3rd generated monte_carlo data file as the x axis values and plot the values in the 2nd column of the 3rd generated monte_carlo data file minus pi as the y axis values and make the line color green
Plot the values in the 1st column of the 4th generated monte_carlo data file as the x axis values and plot the values in the 2nd column of the 4th generated monte_carlo data file minus pi as the y axis values and make the line color orange
Plot the values in the 1st column of the 5th generated monte_carlo data file as the x axis values and plot the values in the 2nd column of the 5th generated monte_carlo data file minus pi as the y axis values and make the line color yellow
End gnuplot