MC - Sheet 2

Exercises

1. Generate 10000 random numbers $r_i$ using Uniform Distribution function in the range [0,1] using C's default RN function, i.e. let $r_i$ be the random number drawn from a uniform distribution function lying between 0 and 1. Using inverse transformation method, find the relation between $r_i$ and $y_i$ such that $y_i$ be the random number drawn from a uniform distribution lying between lower limit "a" and upper limit "b". Using a=0 and b=10, plot the frequency distributions of both the $r_i$ and $y_i$ distributions (choose any suitable bin-width).

2. Generate 100000 random numbers $r_i$ using Uniform Distribution function in the range [0,1] using C's default RN function, i.e. let $r_i$ be the random number drawn from a uniform distribution function lying between 0 and 1. Using inverse transformation method, find the relation between $r_i$ and $x_i$ such that $x_i$ be the random number drawn from an exponential distribution exp(-x) lying between 0 and very large number. Plot the frequency distributions of both the $r_i$ and $x_i$ distributions (choose any suitable bin-width).

3. Generate 100000 random numbers $r_i$ using Uniform Distribution function in the range [0,1] using C's default RN function, i.e. let $r_i$ be the random number drawn from a uniform distribution function lying between 0 and 1. Using acceptance-rejection method, generate the frequency distribution for the function given as equation number 3.8 on page no. 43 of the notes circulated this week.

4. 1-dim Random Walk Simulation (Single Run): Generate "N" random numbers "$u_i$" drawn from a uniform distribution, lying in the range 0 and 1, i.e. U[0,1]. i=1,2,3….N

a. if u < 0.5 then x = x-h
b. else x=x+h; [ initial point of x can be taken as x0 = 0 ]
c. Repeat the above operations, (a) and (b), "N" times

problem#a:

d. Take h=1.0 and N=1000.
e. Plot the trajectory: x vs i (i.e. distance covered vs step taken),
f. Find the value of the actual distance traveled: d(N): xN -x0 (this will be one value only for this problem; I.e. x1000 - x0)

problem#b

g. vary N from 10 to 10000 (unit step);
h. h=1.0 and repeat problem#a.
i. plot d(N) vs N; (there will be one value of d(N) for each N),
j. Also plot $d^2$(N) vs N

problem#c

k. same as problem#b; for different values of h=0.1, 1.0, 2, 10, 50
l. plot d(N) vs N for different values of h
m. plot d^2(N) vs N for different values of h

problem#d

n. For h=0.1, and using problem # b i.e. N varying, execute multiple runs, say, n= 100; This means that for each value of N, you are running your code n=100 times.
o. For each value of “n” you will get a single value of d(N) for one N. Hence you will get“n=100” d(N) values for one “N”.
p. Take the average of d(N) over “n” values, and plot <d> vs N;
q. calculate standard deviation for n=100 values and plot it as an error bar of <d>.
r. Also plot $<d^2>$ and $<d>^2$

5. 2-dim Random Walk Simulation (Single Run): Generate “N” random numbers “$u_i$” drawn from a uniform distribution, lying in the range 0 and 1, i.e. U[0,1]. i=1,2,3….N

a. if u <=0.25 then x=x-h
b. else if 0.25<u<0.5 then x=x+h
c. else if 0.5<u<0.75 then y=y+k
d. else if 0.75<u<=1.0 then y=y-k

e. Repeat the above operations, (a)-(d), “N=1000” times

problem#a:

a. Take h=1.0 and k=1.0 and N=1000.
b. Plot the trajectories: x vs i, y vs i and x vs y (i.e. distance covered vs step taken),
c. Find the value of the actual distance traveled: (i) dx: xN -x0 (this will be one value only for this problem); (ii) dy=yN-y0; (iii) total =dx+dy; (iv) dr = sqrt(dx*dx + dy*dy)

problem#b

d. vary N from 10 to 10000 (unit step);
e. h=1.0 and k=1.0 and repeat problem#a.
f. plot (i) dx vs N, (ii) dy vs N, (iii) total vs N; (iv) dr vs N, and (v) $(dr)^2$ vs N

problem#c

g. same as problem #b; for different values of h,k=0.1, 1.0, 2, 10, 50
h. plot dr vs N for different values of h,k
i. plot $(dr)^2$ vs N for different values of h,k

problem#d

j. For h=0.1, k=0.1 and using problem # b i.e. N varying, execute multiple runs, say, n= 100; This means that for each value of N, you are running your code n=100 times.
k. For each value of “n” you will get a single value of dx, dy, tot,dr for one N. Hence you will get“n=100” dr values for one “N”.
l. Take the average of dx,dy,tot,dr over “n” values, i.e. <dx>,<dy>,<tot>,<dr>,<dr$^2$>,<dr>$^2$ and plot it vs N;
m. calculate standard deviation for n=100 values and plot it as an error bar of <dr>, <dr$^2$>,<dr>$^2$.