1 Task 1

```
function plott = task1(steps, start, left_prob, no_step_prob)
    num = steps; %number of steps
    sum = start; %starting position of the person
    plott(1) = start; %the steps-position plot
    step_direction = rand(num, 1); %random number which decides his movement

    for i = 2:num
        if step_direction(i) < no_step_prob
            sum = sum;
        elseif step_direction(i) < no_step_prob + left_prob
            sum = sum - 1;
        else
            sum = sum + 1;
        end
        plott(i) = sum;
    end

    %figure (1);
    %c = plot(1:num, plott, '-rx');
    %set(c, 'color', 'blue');
    %grid on;
    %title('random walk in 1 dimension')
    %xlabel('steps');
end
```

```
function avgstep = task1Avg(start, steps, left_prob, no_step_prob, simulations, step_needed)
    avgplott(1) = start;
    all_simulations = zeros(simulations, steps);
    for i = 1:simulations
        all_simulations(i,:) = task1(steps, start, left_prob, no_step_prob);
    end

    for j = 2:steps
        sum = 0;
        for i = 1:simulations
            sum = sum + all_simulations(i, j);
        end
        avgplott(j) = sum/simulations;
    end
    avgstep = avgplott(step_needed);

    figure (1);
    c = plot(1:steps, avgplott, '-rx');
    set(c, 'color', 'blue');
    grid on;
    title('avg random walk in 1 dimension')
    ylabel('steps');
end
```

2 Task 2

```
function task2(steps, distance_apart, left_prob1, left_prob2, no_step_prob1, no_step_prob2)
    start1 = 0;
    start2 = 0 + distance_apart;
    sum1 = start1;
    sum2 = start2;
    num = steps;
    step_direction1 = rand(num, 1);
    step_direction2 = rand(num, 1);

    plott_1(1) = start1;
    plott_2(1) = start2;

    for i = 2:num
        if step_direction1(i) < no_step_prob1
            sum1 = sum1;
        elseif step_direction1(i) < no_step_prob1 + left_prob1
            sum1 = sum1 - 1;
        else
            sum1 = sum1 + 1;
        end
        plott_1(i) = sum1;

        if step_direction2(i) < no_step_prob2
            sum2 = sum2;
        elseif step_direction2(i) < no_step_prob2 + left_prob2
            sum2 = sum2 - 1;
        else
            sum2 = sum2 + 1;
        end
        plott_2(i) = sum2;
    end
```

```
32
33      figure (1);
34      c = plot(1:num, plott_1, '-rx');
35      hold on;
36      b = plot(1:num, plott_2, '-rx');
37      set(c, 'color', 'red');
38      set(b, 'color', 'blue');
39      grid on;
40      title('Two People doing random walk in 1 dimension')
41      xlabel('steps');
42      hold off;
43  end
```

```
function avgmeeting_step = task2Avg(start_1, distance, steps, left_prob, no_step_prob, simulations)
    avgplott_1(1) = start_1;
    avgplott_2(1) = start_1 + distance;
    all_simulations_1 = zeros(simulations, steps);
    all_simulations_2 = zeros(simulations, steps);
    for i = 1:simulations
        all_simulations_1(i,:) = task1(steps, start_1, left_prob, no_step_prob);
        all_simulations_2(i,:) = task1(steps, start_1 + distance, left_prob, no_step_prob);
    end

    for j = 2:steps
        sum_1 = 0;
        sum_2 = 0;
        for i = 1:simulations
            sum_1 = sum_1 + all_simulations_1(i, j);
            sum_2 = sum_2 + all_simulations_2(i, j);
        end
        avgplott_1(j) = sum_1/simulations;
        avgplott_2(j) = sum_2/simulations;
    end

    avgmeeting_plott = zeros(1,simulations);
    for i = 1:simulations
        avgmeeting_plott(1, i) = -1;
        for j = 1:steps
            if all_simulations_2(i, j) - all_simulations_1(i, j) <= 0
                avgmeeting_plott(1, i) = j;
                break
            end
        end
    end

%     sum_meeting_step = 0;
%     for i = 1:simulations
```

```
%          sum_meeting_step = sum_meeting_step + avgmeeting_plott(1, i);
%      end
%
%      mean_sum = mean(avgmeeting_plott,'all');
%      disp(mean_sum)

%      avgmeeting_step = sum_meeting_step/simulations;

    avgmeeting_step = mean(avgmeeting_plott,'all');

    figure (1);
    c = plot(1:steps, avgplott_1, '-rx');
    hold on;
    b = plot(1:steps, avgplott_2, '-rx');
    set(c, 'color', 'red');
    set(b, 'color', 'blue');
    grid on;
    title('two people doing random walk in 1 dimension')
    xlabel('steps');
    hold off;
end
```

3 Task 3

```
function task3(steps, startx, starty, prob_nostep, prob_halfstep, prob_left, prob_right, prob_down)
    num = steps;
    sumx = startx;
    sumy = starty;
    plottx(1) = startx;
    plotty(1) = starty;
    %step_direction = randi([1,4],1,num);
    step_direction = rand(num, 1);
    step_size = rand(num, 1);
    step = 0;

    for i = 2:num
        if step_size(i) < prob_nostep
            step = 0;
        elseif step_size(i) < prob_nostep + prob_halfstep
            step = 0.5;
        else
            step = 1;
        end
        if step_direction(i) < prob_left
            sumx = sumx - step;
            if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
                sumx = sumx + 2*step;
            end
        elseif step_direction(i) < prob_left + prob_right
            sumx = sumx + step;
            if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
                sumx = sumx - 2*step;
            end
        elseif step_direction(i) < prob_left + prob_right + prob_down
            sumy = sumy - step;
```

```
            if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
                sumy = sumy + 2*step;
            end
        else
            sumy = sumy + step;
            if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
                sumy = sumy - 2*step;
            end
        end
        plottx(i) = sumx;
        plotty(i) = sumy;
    end

    figure (1);
    c = plot3(plottx, plotty, 1:num, '-rx');
    set(c, 'color', 'red');
    grid on;
    numm = num2str(num);
    str1 = 'Discrete Random Walk in 2 dimension (n = ';
    str2 = ' steps)';
    result = [str1 numm str2];
    xlabel('x-axis');
    ylabel('y-axis');
    zlabel('time');
    title(result);
end
```

4 Task 4

```
function plott = task4(steps, start, left_prob, no_step_prob)
    num = steps; %number of steps
    sum = start; %starting position of the person
    plott(1) = start; %the steps-position plot
    step_direction = rand(num, 1); %random number which decides his movement
    step_size = rand(num, 1);

    for i = 2:num
        if step_direction(i) < no_step_prob
            sum = sum;
        elseif step_direction(i) < no_step_prob + left_prob
            sum = sum - step_size(i);
        else
            sum = sum + step_size(i);
        end
        plott(i) = sum;
    end

    figure (1);
    c = plot(plott, 1:num, '-rx');
    set(c, 'color', 'blue');
    grid on;
    title('random walk in 1 dimension')
    ylabel('steps');
end
```

```
function avgstep = task4Avg(start, steps, left_prob, no_step_prob, simulations, step_needed)
    avgplott(1) = start;
    all_simulations = zeros(simulations, steps);
    for i = 1:simulations
        all_simulations(i,:) = task4(steps, start, left_prob, no_step_prob);
    end

    for j = 2:steps
        sum = 0;
        for i = 1:simulations
            sum = sum + all_simulations(i, j);
        end
        avgplott(j) = sum/simulations;
    end
    avgstep = avgplott(step_needed);

    figure (1);
    c = plot(1:steps, avgplott, '-rx');
    set(c, 'color', 'blue');
    grid on;
    title('avg random walk in 1 dimension')
    ylabel('steps');
end
```

5 Task 5

```
function task5(steps, startx, starty)
    num = steps;
    sumx = startx;
    sumy = starty;
    plottx(1) = startx;
    plotty(1) = starty;
    theta = 2*pi*rand(num,1);
    r = rand(num, 1);

    for i = 2:num
        sumx = sumx + r(i)*cos(theta(i));
        if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
            sumx = sumx - 2*(r(i)*cos(theta(i)));
        end
        sumy = sumy + r(i)*sin(theta(i));
        if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
            sumy = sumy - 2*(r(i)*sin(theta(i)));
        end
        plottx(i) = sumx;
        plotty(i) = sumy;
    end

    figure (1);
    c = plot3(plottx, plotty, 1:num, '-rx');
    set(c, 'color', 'red');
    grid on;
    numm = num2str(num);
    str1 = 'person doing randomwalk in 2 dimension (n = ';
    str2 = ' steps)';
    result = [str1 numm str2];
    xlabel('x-axis');
```

```
    ylabel('y-axis');
    zlabel('time');
    title(result);
end
```

```
function plottxy = task5helper(steps, startx, starty)
    num = steps;
    sumx = startx;
    sumy = starty;
    plottx(1) = startx;
    plotty(1) = starty;
    theta = 2*pi*rand(num,1);
    r = rand(num, 1);

    for i = 2:num
        sumx = sumx + r(i)*cos(theta(i));
        if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
            sumx = sumx - 2*(r(i)*cos(theta(i)));
        end
        sumy = sumy + r(i)*sin(theta(i));
        if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
            sumy = sumy - 2*(r(i)*sin(theta(i)));
        end
        plottx(i) = sumx;
        plotty(i) = sumy;
    end

%     figure (1);
%     c = plot(plottx, plotty, '-rx', 'Color', [rand, rand, ran]);
%
%     grid on;
%     numm = num2str(num);
%     str1 = 'person doing randomwalk in 2 dimension (n = ';
%     str2 = ' steps)';
%     result = [str1 numm str2];
%     xlabel('x-axis');
%     ylabel('y-axis');
%     zlabel('time');
%     title(result);
```

```
%
    plotxy = zeros(2, steps);
    plottxy(1, :) = plottx;
    plottxy(2, :) = plotty;

end
```

6 Task 7

```
function task7(steps, startx, starty, prob_nostep, prob_halfstep)
    num = steps;
    sumx = startx;
    sumy = starty;
    plottx(1) = startx;
    plotty(1) = starty;
    theta = 2*pi*rand(num,1);
    step_size = rand(num, 1);

    for i = 2:num
        if step_size(i) < prob_nostep
            r = 0;
        elseif step_size(i) < prob_nostep + prob_halfstep
            r = 0.5;
        else
            r = 1;
        end
        sumx = sumx + r*cos(theta(i));
        if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
            sumx = sumx - 2*(r*cos(theta(i)));
        end
        sumy = sumy + r*sin(theta(i));
        if ((sumx-startx)^2 + (sumy-starty)^2)^(1/2) > 100
            sumy = sumy - 2*(r*sin(theta(i)));
        end
        plottx(i) = sumx;
        plotty(i) = sumy;
    end

    figure (1);
    c = plot3(plottx, plotty, 1:num, '-rx');
```

```
    set(c, 'color', 'red');
    grid on;
    numm = num2str(num);
    str1 = 'person doing randomwalk in 2 dimension (n = ';
    str2 = ' steps)';
    result = [str1 numm str2];
    xlabel('x-axis');
    ylabel('y-axis');
    zlabel('time');
    title(result);
end
```

7 Task 8

```

function avg_step = task8(sim, steps, a_x, a_y, b_x, b_y)

    nums = sim;
    steps = steps;
    particles1 = zeros(2, steps, nums);
    particles2 = zeros(2, steps, nums);

    for i = 1:nums
        particle1(:,:,i) = task5helper(steps, a_x, a_y);
        particle2(:,:,i) = task5helper(steps, b_x, b_y);
    end

    step_arr = zeros(1, steps);
    % mean_step_arr = zeros(1, nums);


    for i = 1:nums
```

```
        for j = 1:steps
            if ((particle2(1,j,i)-particle1(1,j,i))^2 +(particle2(2,j,i)-particle1(2,j,i)^2))^(1/2) <= 1
                step_arr(i) = j;
            end
        end

    end


avg_step = mean(step_arr, 'all');

end
```

```
function task8sim(sim,sim_internal, steps,a_x, a_y, b_x, b_y)

    num = sim;
    mean_step_arr = zeros(1, num);
    for i = 1:num
        mean_step_arr(i) = task8(sim_internal, steps, a_x, a_y, b_x, b_y);
    end

    mean_mean = mean(mean_step_arr, 'all');
    figure(1);
    c = plot(1:num, mean_step_arr, 'r');

    euc_dist = ((a_x-b_x)^2+(a_y-b_y)^2)^(1/2);
    euc_dist = num2str(euc_dist);

    xlabel('Simulation Number');
    ylabel('Mean of Step at which 2 Particles are within 1 unit of distance');
    mean_mean = num2str(mean_mean);
    str1 = 'Expected Steps of Proximity When Starting from a Distance of ';
    str2 = ' with a deviation of ';
    result = [str1 euc_dist str2 mean_mean];
    title(result);

end
```

8 Task 9

```
function avg_step = task93(sim, steps, a_x, a_y, b_x, b_y, c_x, c_y)

    nums = sim;
    steps = steps;
    particles1 = zeros(2, steps, nums);
    particles2 = zeros(2, steps, nums);
    particles3 = zeros(2, steps, nums);

    for i = 1:nums
        particle1(:,:,i) = task5helper(steps, a_x, a_y);
        particle2(:,:,i) = task5helper(steps, b_x, b_y);
        particle3(:,:,i) = task5helper(steps, c_x, c_y);
    end

    step_arr = zeros(1, steps);
    % mean_step_arr = zeros(1, nums);


    for i = 1:nums
        for j = 1:steps
            a = [particle1(1,j,i), particle1(2,j,i)];
            b = [particle2(1,j,i), particle2(2,j,i)];
            c = [particle3(1,j,i), particle3(1,j,i)];
            if norm(a-b) <= 6 && norm(a-c) <=6 && norm(c-b) <=6
                step_arr(i) = j;
            end
        end

    end


```

```
avg_step = mean(step_arr, 'all');

end
```

```
function task93sim(sim,sim_internal, steps,a_x, a_y, b_x, b_y, c_x, c_y)

    num = sim;
    mean_step_arr = zeros(1, num);
    for i = 1:num
        mean_step_arr(i) = task93(sim_internal, steps, a_x, a_y, b_x, b_y, c_x, c_y);
    end

    mean_mean = mean(mean_step_arr, 'all');
    figure(1);
    c = plot(1:num, mean_step_arr, 'r');

    euc_dist = ((a_x-b_x)^2+(a_y-b_y)^2)^(1/2);
    euc_dist = num2str(euc_dist);

    xlabel('Simulation Number');
    ylabel('Mean of Step at which 3 Particles are within 1 unit of distance');
    mean_mean = num2str(mean_mean);
    str1 = 'Expected Steps of Proximity When Starting from a Distance of ';
    str2 = ' with a deviation of ';
    result = [str1 euc_dist str2 mean_mean];
    title(result);

end
```

```
function avg_step = task94(sim, steps, a_x, a_y, b_x, b_y, c_x, c_y, d_x, d_y)

    nums = sim;
    steps = steps;
    particles1 = zeros(2, steps, nums);
    particles2 = zeros(2, steps, nums);
    particles3 = zeros(2, steps, nums);
    particles4 = zeros(2, steps, nums);

    for i = 1:nums
        particle1(:,:,i) = task58(steps, a_x, a_y);
        particle2(:,:,i) = task58(steps, b_x, b_y);
        particle3(:,:,i) = task58(steps, c_x, c_y);
        particle4(:,:,i) = task58(steps, d_x, d_y);
    end

    step_arr = zeros(1, steps);
    % mean_step_arr = zeros(1, nums);


    for i = 1:nums
        for j = 1:steps
            a = [particles1(1,j,i), particles1(2,j,i)];
            b = [particles2(1,j,i), particles2(2,j,i)];
            c = [particles3(1,j,i), particles3(1,j,i)];
            d = [particles4(1,j,i), particles4(1,j,i)];

            if norm(a-b) <= 6 && norm(a-c) <=6 && norm(a-d) <=6 && norm(b-c) <=6 && norm(b-d) <= 6 && norm(c-d) <=6

                step_arr(i) = j;

            end
        end
    end
```

```
35
36
37 avg_step = mean(step_arr, 'all');
38
39 end
```

```
function task94sim(sim,sim_internal, steps,a_x, a_y, b_x, b_y, c_x, c_y, d_x, d_y)

    num = sim;
    mean_step_arr = zeros(1, num);
    for i = 1:num
        mean_step_arr(i) = task94(sim_internal, steps, a_x, a_y, b_x, b_y, c_x, c_y, d_x, d_y);
    end

    mean_mean = mean(mean_step_arr, 'all');
    figure(1);
    c = plot(1:num, mean_step_arr, 'r');



    xlabel('Simulation Number');
    ylabel('Mean of Step at which 3 Particles are within 1 unit of distance');
    mean_mean = num2str(mean_mean);
    str1 = 'Expected Steps of Proximity When Starting from a Distance of ';
    str2 = ' with a deviation of ';
    result = [str1 euc_dist str2 mean_mean];
    title(result);

end
```