Table of Contents

Script Information 1
Problem 1 1
Problem 2 2
Problem 3 2
Problem 4 3
Problem 5 3
Problem 6 6
Problem 7 7

Script Information

```
% ME112 HW 2
% Author: Chunhui XU
% Date: 2024/03/06
```

Problem 1

```
close all; clear; clc;

% Define the array A
A = [0 -1 4; 9 -14 25; -34 49 64];
B = zeros(size(A));

% Iterate over each element of A
for i = 1:numel(A)
    if A(i) >= 0
        B(i) = sqrt(A(i));
    else
        B(i) = A(i) + 50;
    end
end

A
B
```

```
A =

     0    -1     4
     9   -14    25
   -34    49    64


B =

     0    49     2
     3    36     5
```

```
    16     7     8
```

Problem 2

```
clear; close all; clc;

P = 500;
r = 1.05;
UPPER_LIMIT = 10000;

years = 0;
A = 500;

while A < UPPER_LIMIT
    years = years + 1;
    A = (A .* r) + P;
end

disp(['The answer is: ', num2str(years)]);
```

```
The answer is: 14
```

Problem 3

```
clear;
close all;

PRICES = [19, 18, 22, 21, 25, 19, 17, 21, 27, 29];

shares_owned = 1000;

spent_sum = 0;
sell_sum = 0;

for i = 1:numel(PRICES)
    if PRICES(i) < 20
        spent_sum = spent_sum + 100 * PRICES(i);
        shares_owned = shares_owned + 100;
    end
    if PRICES(i) > 25
        sell_sum = sell_sum + 100 * PRICES(i);
        shares_owned = shares_owned - 100;
    end
end

disp(['Spent in buying: ', num2str(spent_sum)]);
disp(['Received from sale: ', num2str(sell_sum)]);
disp(['Shares own after the 10th day: ', num2str(shares_owned)]);
disp(['Net increase: ', num2str(sell_sum - spent_sum)]);
```

```
Spent in buying: 7300
Received from sale: 5600
```

Shares own after the 10th day: 1200
Net increase: -1700

Problem 4

```
clear; close all; clc;

cmat = [1 28 3;
        7 18 7;
        8 16 4;
        17 2 5;
        22 10 2;
        27 8 6;];

min_cost = Inf;
res = [0, 0];
for x = 0:30
    for y = 0:30
        pos = [x, y];
        total_cost = 0;
        for i = 1:6
            distance = sqrt((cmat(i, 1) - pos(1)).^2 + (cmat(i, 2) -
pos(2)).^2);
            total_cost = total_cost + 0.5 .* distance .* cmat(i, 3);
        end
        if (total_cost < min_cost)
            min_cost = total_cost;
            res = [x, y];
        end
    end
end

fprintf("The result position is [%d, %d]", res(1), res(2));
```

The result position is [9, 16]

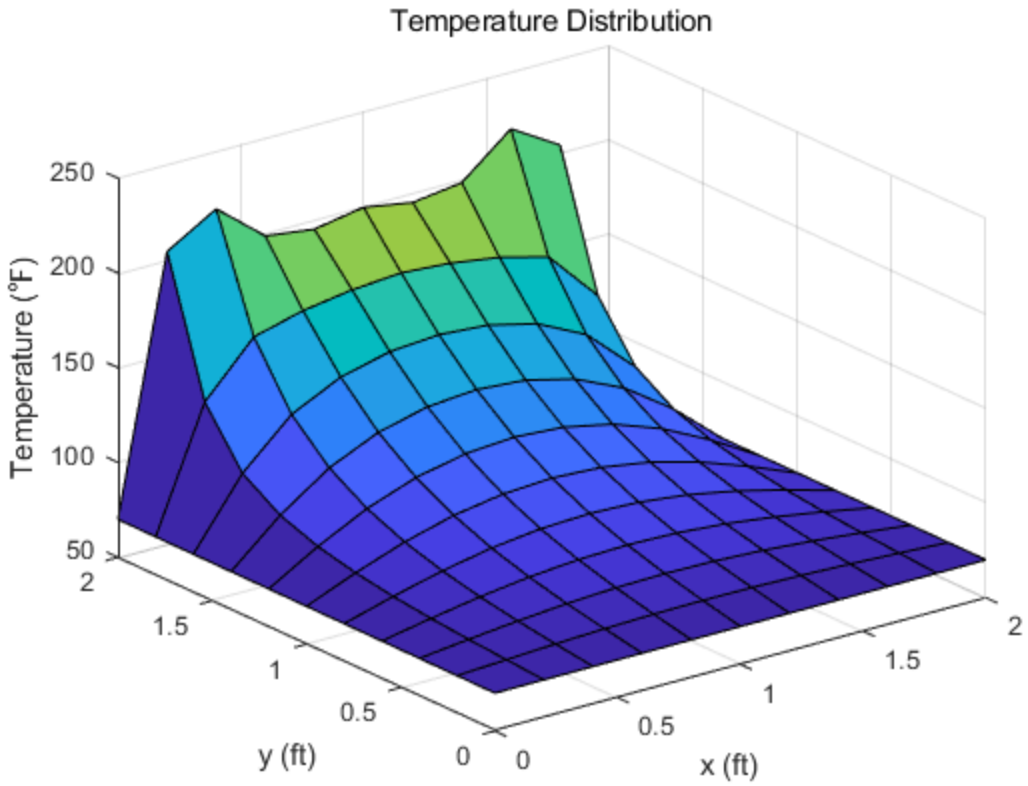
Problem 5

```
clear; close all; clc;

% Constants
L = 2;
W = 2;
T1 = 70;
T2 = 200;
x = 1;
y = 1;

disp('For problem a.:')

first_term = (2/pi) * 2 * sin(pi*x/L) * sinh(1*pi*y/L) / sinh(1*pi*W/L);
w_xy_res = 0;
diff_max = first_term;
```

```
is_convergent = true;

% Compute and display terms for n=1 to 19
for n = 3:2:19 % Loop only through odd numbers
    diff = (2/pi) * (2/n) * sin(pi*x/L) * sinh(1*pi*y/L) / sinh(1*pi*W/L);
    if (diff > diff_max)
        is_convergent = false;
        fprintf('Something wrong at n = %d\n', n);
        break;
    end
    diff_max = diff;
end

if(is_convergent)
    disp('Great, it is convergent');
else
    disp('Oh no, it is not convergent');
end

disp('For problem b.:')

function res = estimate_res(x, y, esp, disp_n)
    w_xy = (2/pi) * 2 * sin(pi*x/2) * sinh(1*pi*y/2) / sinh(1*pi);
    T_init = get_T(w_xy);

    diff_percent = Inf;
    n = 1;
    while diff_percent > esp
        n = n + 2;
        w_xy = w_xy + (2/pi)* (2/n) * sin(n*pi*x/2) * sinh(n*pi*y/2) /
sinh(n*pi);
        T_res = get_T(w_xy);
        diff_percent = abs(T_res - T_init)/ T_init;
        T_init = T_res;
    end
    res = T_init;
    if(disp_n)
        fprintf('Needed n is %d, needed term is %d\n', n, (n+1)/2);
    end
end

function res = get_T(w_xy)
    res = (200 - 70) * w_xy + 70;
end

estimate_res(x, y, 0.01, true);

disp('For problem c.:')

[X, Y] = meshgrid(0:0.2:L, 0:0.2:W);

% Compute temperature distribution
T = zeros(size(X));
for i = 1:size(X,1)
```

```
    for j = 1:size(Y,1)
        % ATTENTION:
        % Too high precision here seems to cause MATLAB to abandon the loop
directly
        % So my figure didn't fit well
        T(i,j) = estimate_res(X(i,j), Y(i,j), 0.005, false);
    end
end

% Plot the temperature distribution
figure;
surf(X, Y, T);
xlabel('x (ft)');
ylabel('y (ft)');
zlabel('Temperature (°F)');
title('Temperature Distribution');
```

For problem a.:
Great, it is convergent
For problem b.:
Needed n is 3, needed term is 2
For problem c.:



Problem 6

```
clear; close all; clc;

% Initialization
balance = 10000;
totalEarned = 0;

% Display header
fprintf('%-7s %-15s %-17s %-13s %-14s\n', 'Month', 'Interest Rate','Interest
Earned', 'New Balance', 'Total Interest');
fprintf('-------------------------------------------------------------------
--------\n');

% Loop 12 months
for month = 1:12
    rate = 0;
    if balance <= 15000
        rate = 1;
    elseif balance <= 20000
        rate = 1.5;
    else
        rate = 2;
    end

    % Calculate monthly interest
    interestEarned = balance * (rate / 100);
    % Update the total interest earned
    totalEarned = totalEarned + interestEarned;
    % Update the balance with the interest earned
    balance = balance + interestEarned;
    % Add the monthly deposit to the balance
    balance = balance + 1000;
    % Display information for the current month
    fprintf('%-7d %%%-15.1f $%-17.2f $%-13.2f $%-14.2f\n', month, rate,
interestEarned, balance, totalEarned);
end
```

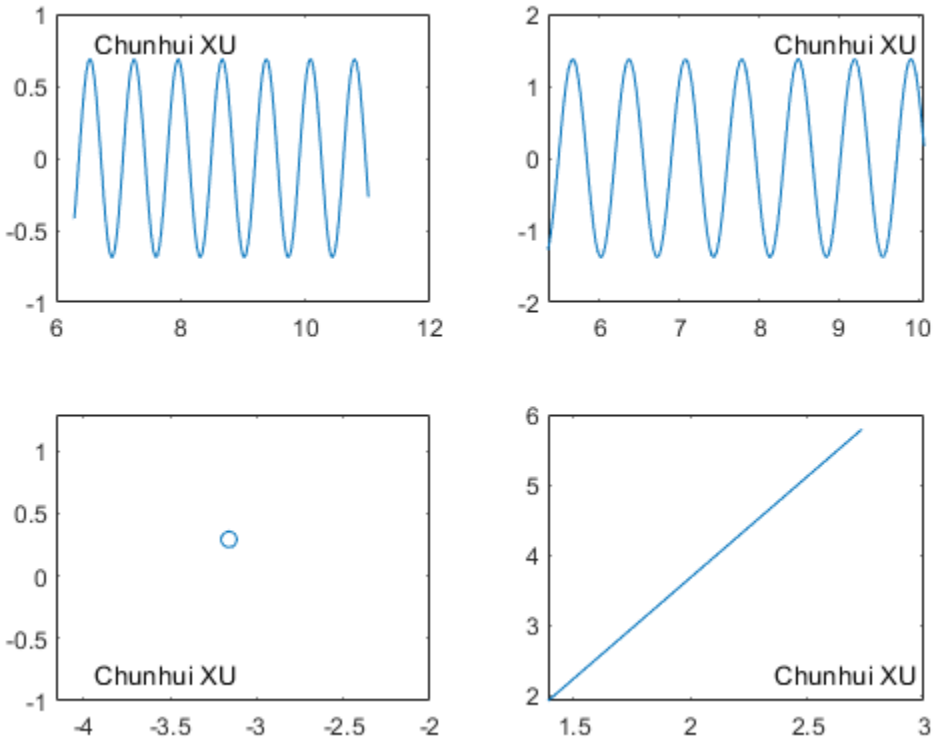
```
Month   Interest Rate   Interest Earned   New Balance   Total Interest
-------------------------------------------------------------------------
1       %1.0            $100.00           $11100.00     $100.00
2       %1.0            $111.00           $12211.00     $211.00
3       %1.0            $122.11           $13333.11     $333.11
4       %1.0            $133.33           $14466.44     $466.44
5       %1.0            $144.66           $15611.11     $611.11
6       %1.5            $234.17           $16845.27     $845.27
7       %1.5            $252.68           $18097.95     $1097.95
8       %1.5            $271.47           $19369.42     $1369.42
9       %1.5            $290.54           $20659.96     $1659.96
10      %2.0            $413.20           $22073.16     $2073.16
11      %2.0            $441.46           $23514.62     $2514.62
12      %2.0            $470.29           $24984.92     $2984.92
```

Problem 7

This MATLAB script tests a user-defined function that places a name on randomly sized plots.

```
% Generate randomized data to plot
clear; close all; clc;

figure;
% x data
xmin = (-10) + (10-(-10)).*rand; % Generate random number between -10 and 10
xrange = 2 + (5-2).*rand; % Generate random number between 2 and 5
xmax = xmin + xrange;
numPts = 150; % Number of data points
x = linspace(xmin,xmax,numPts);
x2 = x-0.2*xrange;
% y data
Amp = 0.5 + (2-0.5).*rand; % Generate random amplitude between 0.5 and 2
Freq = 0.5 + (1.5-0.5) .*rand; % Generate random freg between 0.5 and 1.5
y = Amp*sin(2*pi*Freq*x);
y2 = 2*Amp*cos(2*pi*Freq*x2);

% Plot data and test your function
r = 2; % number of subplot rows
c = 2; % number of subplot columns

plot1 = subplot(r,c,1);
plot(x,y);
putNameOnPlot('UL');

plot2 = subplot(r,c, 2);
plot(x2,y2);
putNameOnPlot('UR');

plot3 = subplot(r,c,3);
plot(-5*rand,3*rand,'o');
putNameOnPlot('LL');

plot4 = subplot(r,c,4);
plot([5*rand 5*rand] ,[2*rand,6*rand]);
putNameOnPlot('LR');

function putNameOnPlot(type)
    position = [0, 0];
    switch type(1)
        case 'L'
            position(2) = 0.1;
        case 'U'
            position(2) = 0.9;
    end
    switch type(2)
        case 'L'
            position(1) = 0.1;
        case 'R'
```

```
            position(1) = 0.6;
    end

    % Place the name on the plot
    text('String', 'Chunhui XU', 'Units', 'Normalized', 'Position',
position);

    hold on;
end
```



Published with MATLAB® R2024a