1. Periodic solution of periodic linear dynamical system. Consider the linear dynamical system \[ \dot{x} = A(t)x \] where 
\[ A(t) = \begin{cases} A_1 & 2k \leq t < 2k + 1, \ k = 0, 1, 2, \ldots \\ A_2 & 2k + 1 \leq t < 2k + 2, \ k = 0, 1, 2, \ldots \end{cases} \]
In other words, \( A(t) \) switches between the two values \( A_1 \in \mathbb{R}^{n \times n} \) and \( A_2 \in \mathbb{R}^{n \times n} \) every second. The matrix \( A(t) \) is periodic with period 2, i.e., \( A(t + 2) = A(t) \) for all \( t \geq 0 \).

a) Existence of a periodic trajectory. What are the conditions on \( A_1 \) and \( A_2 \) under which the system has a nonzero periodic trajectory, with period 2? By this we mean: there exists \( x : \mathbb{R}_+ \to \mathbb{R}^n \), \( x \) not identically zero, with \( x(t + 2) = x(t) \) and \( \dot{x} = A(t)x \).

b) All trajectories are asymptotically periodic. What are the conditions on \( A_1 \) and \( A_2 \) under which all trajectories of the system are asymptotically 2-periodic? By this we mean: for every \( x : \mathbb{R}_+ \to \mathbb{R}^n \) with \( \dot{x} = A(t)x \), we have 
\[ \lim_{t \to \infty} \|x(t + 2) - x(t)\| = 0. \]
(Note that this holds when \( x \) converges to zero . . .)

Please note:

- Your conditions should be as explicit as possible. You can refer to the matrices \( A_1 \) and \( A_2 \), or any matrices derived from them using standard matrix operations, their eigenvalues and eigenvectors or Jordan forms, singular values and singular vectors, etc.

- We do not want you to give us a condition under which the property described holds. We want you to give us the most general conditions under which the property holds.

Solution. The important point in this problem is to express \( x(t + 2) \) in terms of \( x(t) \). To begin with, let’s consider what happens when \( t = 2k \) is an even integer. Then we have
\[ x(t + 1) = e^{A_1}x(t), \quad x(t + 2) = e^{A_2}x(t + 1) = e^{A_2}e^{A_1}x(t). \]
Please note that the expression \( x(t + 2) = e^{A_2 + A_1}x(t) \) is wrong, since \( e^{A_2 + A_1} \neq e^{A_2}e^{A_1} \) (in general). Proceeding, we see that if \( x \) is a periodic trajectory, then the matrix \( e^{A_2}e^{A_1} \) must satisfy 
\[ e^{A_2}e^{A_1}x(t) = x(t), \]
i.e., it has an eigenvalue of one. Conversely, suppose that \( e^{A_2}e^{A_1} \) has an eigenvalue of one, i.e., there is some nonzero \( v \) such that \( e^{A_2}e^{A_1}v = v \). Then, if we set \( x(0) = v \), the resulting trajectory is periodic with period 2. So the necessary and sufficient condition for the system to have a nonzero 2-periodic trajectory is that the matrix \( e^{A_2}e^{A_1} \) has an eigenvalue of one. One common error was to give the condition \( e^{A_2}e^{A_1} = I \), which is too strong. In fact, this is the condition under which every trajectory of the system is 2-periodic. It’s also interesting
to see what condition we get if we consider other starting times. For example, let 0 < h < 1. Then we have, for \( k = 1, 2, 3, \ldots \),
\[
x(2k + h + 2) = e^{hA_1}e^{A_2}e^{(1-h)A_1}x(2k + h).
\]
(These terms, from right to left, first propagate the state to time \( t = 2k + 1 \), then to \( t = 2k + 2 \), then finally, to \( t = 2k + 2 + h \).) Using the same argument as above, the condition is now that the matrix
\[
e^{hA_1}e^{A_2}e^{(1-h)A_1}x(2k + h)
\]
have an eigenvalue of one. The natural question is, how is this condition related to the one found above? It turns out (as it had to) that the two conditions are identical. The reason is that the eigenvalues of the matrices
\[
F = e^{A_2}e^{A_1}, \quad G = e^{hA_1}e^{A_2}e^{(1-h)A_1}x(2k + h)
\]
are exactly the same, since they are similar: \( G = T^{-1}FT \), where \( T = e^{-hA_1} \). All trajectories are asymptotically periodic. Again everything depends on the matrix \( F = e^{A_2}e^{A_1} \) which is called the monodromy matrix for the system, by the way. The requirement is that all solutions of the discrete-time system \( z(t+1) = Fz(t) \) are asymptotically constant. This means that the eigenvalues of \( F \) are each either smaller than one in magnitude, or exactly one. In the latter case, it is required that every eigenvalue equal to one must be associated with a Jordan block of size \( 1 \times 1 \). (Whew!) A common error here was to end up with the condition that all eigenvalues of \( F \) have magnitude less than one. This is too strong: it is the condition under which all solutions converge to zero.

2. Properties of the matrix exponential.

a) Show that \( e^{A+B} = e^Ae^B \) if \( A \) and \( B \) commute, i.e., \( AB = BA \).

b) Carefully show that \( \frac{d}{dt}e^{At} = Ae^{At} = e^{At}A \).

Solution.

a) We will show that if \( A \) and \( B \) commute then \( e^{A}e^{B} = e^{A+B} \). We begin by writing the expressions for \( e^A \) and \( e^B \)
\[
e^A = I + A + \frac{A^2}{2!} + \frac{A^3}{3!} + \cdots
\]
\[
e^B = I + B + \frac{B^2}{2!} + \frac{B^3}{3!} + \cdots
\]
Now we multiply both expressions and get
\[
e^{A}e^{B} = I + A + B + AB + \frac{A^2}{2!} + \frac{B^2}{2!} + \frac{A^3}{3!} + \frac{A^2B}{2!} + \frac{AB^2}{2!} + \frac{B^3}{3!} + \cdots
\]
\[
= I + A + B + \frac{A^2 + 2AB + B^2}{2!} + \frac{A^3 + 3A^2B + 3AB^2 + B^3}{3!} + \cdots
\]
Now we note that, if \( A \) and \( B \) commute, we are able to write things such as \( (A+B)^2 = A^2 + 2AB + B^2 \). So, if \( A \) and \( B \) commute we can finally write
\[
e^{A}e^{B} = I + (A + B) + \frac{(A+B)^2}{2!} + \frac{(A+B)^3}{3!} + \cdots = e^{A+B}
\]
b) It suffices to note that $A$ commute with itself. Then one can write

\[
\frac{de^{At}}{dt} = A + A^2t + \frac{A^3t^2}{2!} + \cdots
\]

\[
= A(I + At + \frac{(At)^2}{2!} + \cdots)
\]

\[
= (I + At + \frac{(At)^2}{2!} + \cdots)A
\]

\[
= Ae^{At} = e^{At}A
\]

3. Determinant of matrix exponential.

a) Suppose the eigenvalues of $A \in \mathbb{R}^{n \times n}$ are $\lambda_1, \ldots, \lambda_n$. Show that the eigenvalues of $e^A$ are $e^{\lambda_1}, \ldots, e^{\lambda_n}$. You can assume that $A$ is diagonalizable, although it is true in the general case.

b) Show that $\det e^A = e^{\text{trace } A}$. Hint: $\det X$ is the product of the eigenvalues of $X$, and $\text{trace } Y$ is the sum of the eigenvalues of $Y$.

Solution.

a) Suppose that $A$ is diagonalizable with eigenvalues $\lambda_1, \ldots, \lambda_n$. Therefore, the invertible matrix $T$ exists such that

\[
A = T \text{ diag}(\lambda_1, \ldots, \lambda_n)T^{-1}
\]

and we get

\[
e^A = Te^{\text{diag}(\lambda_1, \ldots, \lambda_n)}T^{-1} = T \text{ diag}(e^{\lambda_1}, \ldots, e^{\lambda_n})T^{-1}.
\]

As a result

\[
e^AT = T \text{ diag}(e^{\lambda_1}, \ldots, e^{\lambda_n})
\]

which shows that the eigenvalues of $e^A$ are $e^{\lambda_1}, \ldots, e^{\lambda_n}$. Note that this also shows that the eigenvectors of $A$ (the columns of $T$) and $e^A$ are the same.

b) The determinant of a matrix is equal to the product of its eigenvalues and therefore

\[
\det e^A = e^{\lambda_1}e^{\lambda_2} \cdots e^{\lambda_n} = e^{\lambda_1 + \lambda_2 + \cdots + \lambda_n}.
\]

But $\lambda_1 + \lambda_2 + \cdots + \lambda_n$ is the sum of the eigenvalues of $A$ which is equal to $\text{trace } A$. Thus

\[
\det e^A = e^{\text{trace } A}.
\]

4. Optimal initial conditions for a bioreactor. The dynamics of a bioreactor are given by $\dot{x}(t) = Ax(t)$, where $x(t) \in \mathbb{R}^n$ is the state, with $x_i(t)$ representing the total mass of species or component $i$ at time $t$. Component $i$ has (positive) value (or cost) $c_i$, so the total value (or cost) of the components at time $t$ is $c^T x(t)$. (We ignore any extra cost that would be incurred in separating the components.) Your job is to choose the initial state, under a budget constraint, that maximizes the total value at time $T$. More specifically, you are to choose $x(0)$, with all entries nonnegative, that satisfies $c^T x(0) \leq B$, where $B$ is a given positive budget. The problem data (i.e., things you know) are $A$, $c$, $T$, and $B$. 

3
You can assume that $A$ is such that, for any $x(0)$ with nonnegative components, $x(t)$ will also have all components nonnegative, for any $t \geq 0$. (This occurs, by the way, if and only if the off-diagonal entries of $A$ are nonnegative.)

a) Explain how to solve this problem.

b) Carry out your method on the specific instance with data

$$A = \begin{bmatrix} 0.1 & 0.1 & 0.3 & 0 \\ 0 & 0.2 & 0.4 & 0.3 \\ 0.1 & 0.3 & 0.1 & 0 \\ 0 & 0 & 0.2 & 0.1 \end{bmatrix}, \quad c = \begin{bmatrix} 3.5 \\ 0.6 \\ 1.1 \\ 2.0 \end{bmatrix}, \quad T = 10, \quad B = 1.$$

Give the optimal $x(0)$, and the associated (optimal) terminal value $c^T x(T)$.

Give us the terminal value obtained when the initial state has equal mass in each component, i.e., $x(0) = \alpha \mathbf{1}$, with $\alpha$ adjusted so that the total initial cost is $B$. Compare this with the optimal terminal value.

Also give us the terminal value obtained when the same amount, $B/n$, is spent on each initial state component (i.e., $x(0)_i = B/(nc_i)$). Compare this with the optimal terminal value.

Solution.

a) We have $c^T x(T) = c^T e^{tA} x(0) = b^T x(0)$, where we define $b = (e^{T A}) c$, so our problem is to maximize $b^T x(0)$, subject to $x(0) \geq 0$ (this means all its entries are nonnegative), and $c^T x(0) \leq B$. You can think of $c_i$ as the cost of investing in a unit of component $i$, and $b_i$ as the payoff received. Thus, the gain is $b_i/c_i$. The solution to this problem is to invest everything (i.e., the whole budget $B$) in any component that has maximum gain. More formally, we choose any $k$ for which $b_k/c_k = \max \{b_1/c_1, \ldots, b_n/c_n\}$, and then set $x(0) = B e_k/c_k$. (Recall that we assume $b_i \geq 0$ and $c_i > 0$ here.)

We didn’t require a completely formal proof that this is the optimal strategy. But here is one, just so you know what one looks like. Suppose that $x(0)$ satisfies $x(0) \geq 0$, $c^T x(0) \leq B$. Then we have

$$b^T x(0) = \sum_{i=1}^{n} (b_i/c_i)(c_i x(0)_i)$$

$$\leq \left( \max_{i=1,\ldots,n} (b_i/c_i) \right) \left( \sum_{i=1}^{n} c_i x(0)_i \right)$$

$$\leq B \max_{i=1,\ldots,n} (b_i/c_i).$$

This shows that no feasible choice of $x(0)$ can yield terminal value $c^T x(T) = b^T x(0)$ more than $B \max_{i=1,\ldots,n} (b_i/c_i)$. But the choice described above yields this value of $b^T x(0)$, and so must be optimal.
b) The code below solves the problem.

```matlab
% problem data
A=[ 0.1 0.1 0.3 0; 0 0.2 0.4 0.3; 0.1 0.3 0.1 0; 0 0 0.2 0.1];
c=[3.5; 0.6; 1.1; 2.0];
n=length(c);
T=10;
B=1;
b=(expm(T*A))'*c;
[g,k]=max(b./c); % get max value and index k
opt_x0=zeros(n,1);
opt_x0(k)=B/c(k);

% terminal value with equal mass in each initial component
x0mass=(B/sum(c))*ones(n,1);
term_value=b'*x0mass

% terminal value with equal value in each initial component
x0val=(B/n)./(c);
term_value=b'*x0val
```

The optimal initial condition is $x(0) = (5/3)e_2$, which yields terminal value $1168$.

With equal initial mass in each component, the terminal value is $300$; with equal initial investment in each component, the terminal value is $552$.

5. **Optimal espresso cup pre-heating.** At time $t = 0$ boiling water, at $100^\circ$C, is poured into an espresso cup; after $P$ seconds (the 'pre-heating time'), the water is poured out, and espresso, with initial temperature $95^\circ$C, is poured in. (You can assume this operation occurs instantaneously.) The espresso is then consumed exactly $15$ seconds later (yes, instantaneously). The problem is to choose the pre-heating time $P$ so as to maximize the temperature of the espresso when it is consumed.

We now give the thermal model used. We take the temperature of the liquid in the cup (water or espresso) as one state; for the cup we use an $n$-state finite element model. The vector $x(t) \in \mathbb{R}^{n+1}$ gives the temperature distribution at time $t$: $x_1(t)$ is the liquid (water or espresso) temperature at time $t$, and $x_2(t), \ldots, x_{n+1}(t)$ are the temperatures of the elements in the cup. All of these are in degrees C, with $t$ in seconds. The dynamics are

$$ \frac{d}{dt}(x(t) - 20 \cdot 1) = A(x(t) - 20 \cdot 1), $$

where $A \in \mathbb{R}^{(n+1) \times (n+1)}$. (The vector $20 \cdot 1$, with all components $20$, represents the ambient
temperature.) The initial temperature distribution is 

\[
x(0) = \begin{bmatrix}
100 \\
20 \\
\vdots \\
20
\end{bmatrix}.
\]

At \( t = P \), the liquid temperature changes instantly from whatever value it has, to 95; the other states do not change. Note that the dynamics of the system are the same before and after pre-heating (because we assume that water and espresso behave in the same way, thermally speaking).

We have very generously derived the matrix \( A \) for you. You will find it in \texttt{espressodata.json}. In addition to \( A \), the file also defines \( n \), and, respectively, the ambient, espresso and preheat water temperatures \( T_a \) (which is 20), \( T_e \) (95), and \( T_l \) (100).

Explain your method, submit your code, and give final answers, which must include the optimal value of \( P \) and the resulting optimal espresso temperature when it is consumed. Give both to an accuracy of one decimal place, as in

\[ P = 23.5 \text{ s}, \] which gives an espresso temperature at consumption of 62.3°C.’

(This is not the correct answer, of course.)

\section*{Solution.} After \( P \) seconds of pre-heating, we will have

\[ x(P) - 20 \cdot 1 = e^{PA}(x(0) - 20 \cdot 1). \]

Define a new vector \( \tilde{x}(P) \) with \( \tilde{x}_i(P) = x_i(P) \) for \( i = 2, \ldots, n + 1 \), and \( \tilde{x}_1(P) = 95 \). (Thus, \( \tilde{x}(P) \) is the state immediately after the water is replaced with espresso.) The temperature distribution at time \( P + 15 \) will be

\[ x(P + 15) - 20 \cdot 1 = e^{15A}(\tilde{x}(P) - 20 \cdot 1). \]

We now have a method for calculating the temperature of the espresso at the instant of consumption for a given \( P \):

\[ T(P) - 20 = e_1^T x(P + 15) = e_1^T e^{15A}(\tilde{x}(P) - 20 \cdot 1), \]

where \( e_1 \) is the first unit vector. Thus, we have

\[ T(P) = e_1^T e^{15A}(\tilde{x}(P) - 20 \cdot 1) + 20. \]

To find the optimal value of \( P \) we use a simple search method, by calculating \( T(P) \) over a finely-sampled range of values of \( P \), and selecting the maximum value.

The optimal preheating time for this example is 11.1 seconds. This will give an espresso temperature of 87.6°C.

Matlab code to calculate the answers appears below.

```matlab
% load data.
espressodata;
```
% Test a range of preheating times up to a minute.
Tphs = linspace(0, 60, 1000);
% Condition at instant when preheating liquid is added.
% Note change of coordinates by subtracting Ta (and elsewhere).
p0 = [Tl; Ta*ones(n,1)] - Ta;

y = zeros(size(Tphs));
for i = 1:length(Tphs)
    Tph = Tphs(i);
    % Find state after preheating by propagating forward.
xph = expm(Tph*A)*p0;
    % Instantaneously add espresso, changing only the liquid portion of the
    % state.
xph(1) = Te - Ta;
    % Record temperature at time 15.
z = expm(15*A)*xph;
y(i) = z(1);
end

[Tmax, i] = max(y+Ta);

The graph below shows how preheat time affects the drinking temperature.

The next graph shows the temperature of the espresso over a 5 minute period, with and
6. Linear dynamical system with constant input. We consider the system \( \dot{x} = Ax + b \), with \( x(t) \in \mathbb{R}^n \). A vector \( x_e \) is an equilibrium point if \( 0 = Ax_e + b \). (This means that the constant trajectory \( x(t) = x_e \) is a solution of \( \dot{x} = Ax + b \).)

a) When is there an equilibrium point?

b) When are there multiple equilibrium points?

c) When is there a unique equilibrium point?

d) Now suppose that \( x_e \) is an equilibrium point. Define \( z(t) = x(t) - x_e \). Show that \( \dot{z} = Az \). From this, give a general formula for \( x(t) \) (involving \( x_e \), \( \exp(tA) \), \( x(0) \)).

e) Show that if all eigenvalues of \( A \) have negative real part, then there is exactly one equilibrium point \( x_e \), and for any trajectory \( x(t) \), we have \( x(t) \to x_e \) as \( t \to \infty \).

Solution.

a) An equilibrium point \( x_e \) exists if and only if \( 0 = Ax_e + b \), i.e., \( -b = Ax_e \). This happens exactly when \( -b \in \text{range}(A) \).

b) If \( x_e \) is any equilibrium point, and \( z \in \text{null}(A) \), then \( x_e + z \) is also an equilibrium point. It follows that in order to have multiple equilibrium points, we need \( \text{null}(A) \neq \{0\} \), as well as \( b \in \text{range}(A) \).

c) For uniqueness, we need that \( \text{null}(A) = \{0\} \), in addition to \( b \in \text{range}(A) \). The nullspace condition implies that \( A \) is nonsingular. But this means that \( \text{range}(A) = \mathbb{R}^n \), so the condition \( b \in \text{range}(A) \) holds automatically. In this case, the unique equilibrium point is \( x_e = -A^{-1}b \). In summary: there is a unique equilibrium point if and only if \( A \) is nonsingular; in this case, we have \( x_e = -A^{-1}b \).
d) \( z(t) = \exp(tA)z(0) \), so

\[
x(t) = x_e + \exp(tA)(x(0) - x_e).
\]

e) Assume that all eigenvalues of \( A \) have negative real part. In particular, no eigenvalue can be zero, which means \( A \) is nonsingular. Therefore the unique equilibrium point is \( x_e = -A^{-1}b \). Since all eigenvalues of \( A \) have negative real part, the matrix \( \exp(tA) \) goes to zero as \( t \to \infty \). From the formula for \( x(t) \) above, we see that \( x(t) \) converges to \( x_e \).