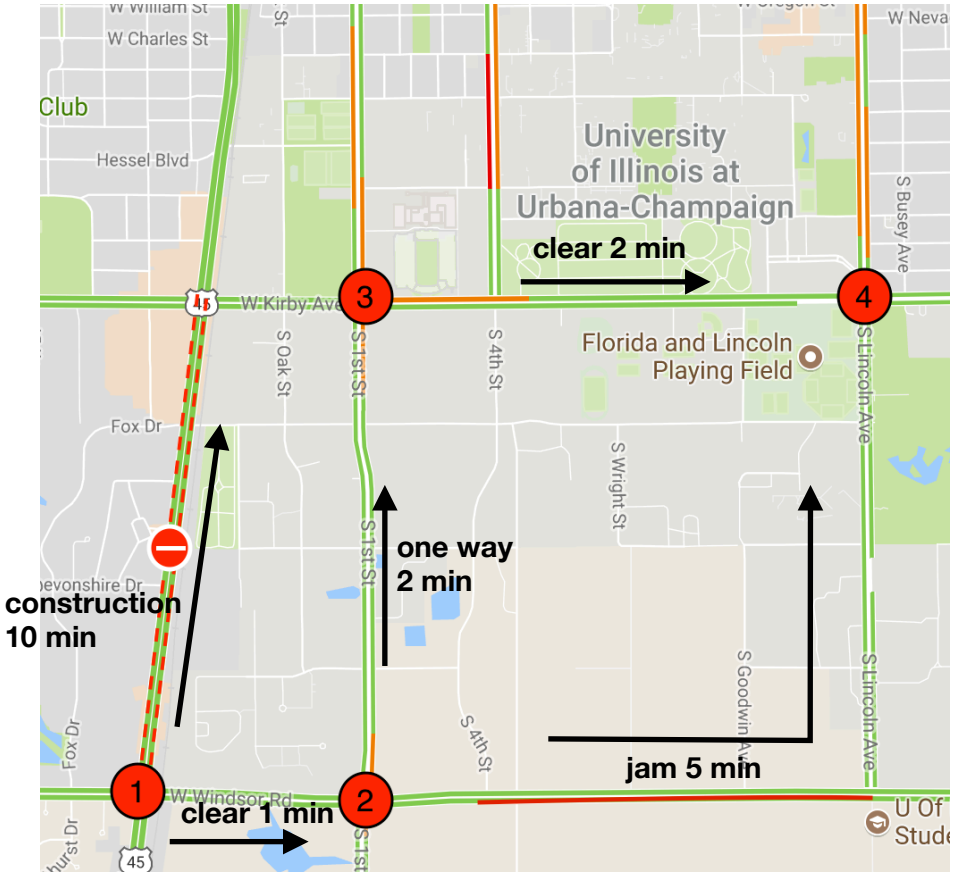
BIOE 198MI Biomedical Data Analysis. Spring Semester 2019.
Dynamic programming: finding the shortest path

Problem Statement: we're going to learn how to convert real life problem into a graphical diagram and solving the shortest path problem with 1) traditional ways , by counting the cost of all the possible paths and 2) dynamic programming.

Nowadays we use navigation applications such as google maps, to find the most efficient route. We can consider the streets map as a graph, made out of nodes (locations on the map), and edges (roads connecting different locations). For example, consider the following map. There are total of 4 nodes labeled as node 1, 2, 3, 4. The road between node 1 and 3 is under construction and may take up to 10 minutes to pass. The road from node 2 to node 4 via Windsor Rd has heavy traffic jam, the passing time is about 5 minutes. All other roads are clear. From node 1 to 2 takes about 1 minutes. Both cases from node 3 to 4 and 2 to 3 take about 2 minutes



To simply the map, we can draw graphical diagram as shown below.

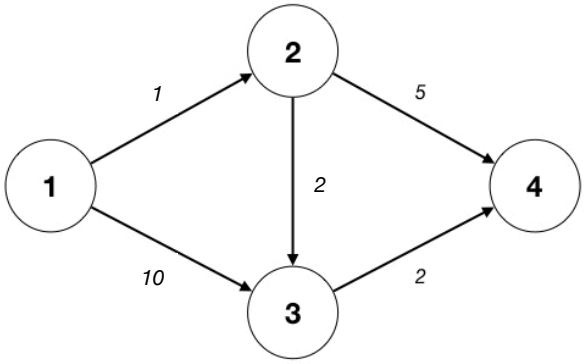


Fig.1 simplified graph of the map, edge cost has unit of minute

This graph is called topologically ordered directed graph.

- Node: states/conditions you want to reach. In this case, locations. Node 1 is the starting/origin node. Node 4 is the ending/destination node.
- Edge cost: sometimes also called edge weight, is the cost traveling from one source node to one target node.
- Directed: nodes are connected with arrows (also called edges). And the arrows are directional.
- Topologically ordered: arrows goes from smaller node to larger node.
- If there's no arrow between a pair of nodes, we assume the edge cost is infinity.

1. Traditional ways:

First for simple diagram, we can easily find **all the possible paths**

Path 1: Node(1) → Node(2) → Node(3) → Node(4)

Path 2: Node(1) → Node(2) → Node(4)

Path 3: Node(1) → Node(3) → Node(4)

Then we calculate the cost of each path

Path 1 cost: Edge(1⟶2) + Edge(2⟶3)+ Edge(3⟶4) = 1+2+2 = **5 min**

Path 2 cost: Edge(1⟶2) + Edge(2⟶4) = 1+5 = 6 min

Path 3 cost: Edge(1⟶3) + Edge(3⟶4) = 10+2 = 12 min

1. Solving the problem using **MATLAB**.

```
%% Section 1: Create MATLAB digraph
```

How do we transfer the graph into the MATLAB matrix that contains the node to node cost information?

From	1	1	2	2	3
To	2	3	3	4	4
Cost	1	10	2	5	2

We created a variable called 'Edge' to store all that information we generated.

Edge (i, j) — is the time cost along node i to node j, mathematically expressed as e_{ij} .

Note that $e_{ii} = \infty$, you can't travel to the node you're already at.

eg. Edge(1,2) = 1

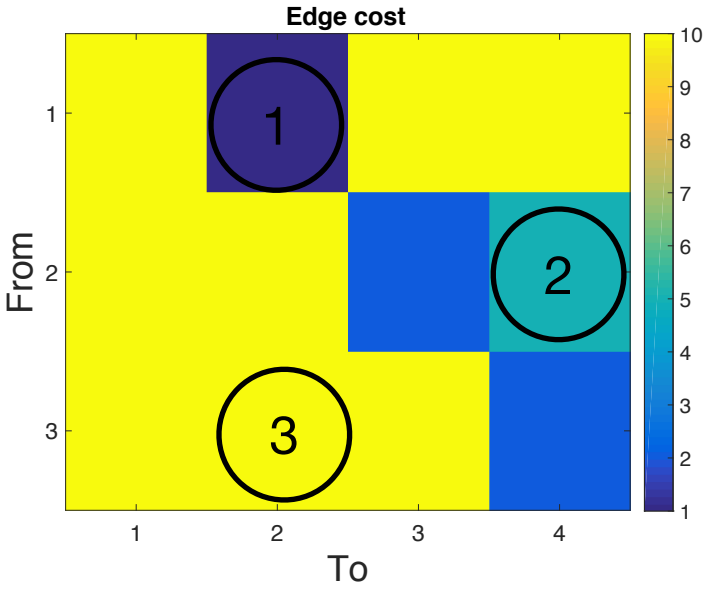
Edge(1,3)= 10

Question: What's the meaning of one column/ row of the 'Edge' matrix?

Note that, now all the edge costs for non-existing arrows are zeros. This doesn't make sense. We should set them to infinity. Can you write something in line 18 to make every element in the variable `Edge` that has value of zero to infinity? (hint: use logical operators (==) you learn from lab 8, infinity is `inf` in MATLAB, eg: `a = inf`)

```
% Visualize The Edge matrix
```

Figure 1 uses the MATLAB function ‘imagesc’ to visualize the Edge cost. Do you know how to interpret this figure? and what do values in location 1,2 and 3 mean?




```
% Visualize the diagram
```

Using the MATLAB function ‘digraph’ with input of source nodes, target nodes and edge weights , we create a directed graph object ‘G’, and simply use ‘plot’ function to draw the diagram.

```
p = plot(G);
p.EdgeLabel = Cost; % show the edge cost
p.NodeColor = 'r'; % set the nodes color to red
p.LineWidth = 2; % set line width
```

`p` is a handle of the directed graph. In workspace, there’s a variable called ‘`p`’. Double click, it pops up a list of properties of the graph. Here we choose to modify the ‘EdgeLabel' so the graph shows certain text as edge cost, ‘NodeColor’ is changed to red, ‘LineWidth’ of the edge to 2. Can you change the marker size 8 and arrow size to 20?

```
%% Section 2: Find the shortest path and minimum time cost by
calculate the cost of every possible path.
```

function '`pathbetweennodes`' helps us find all possible paths.

Question: What class is the variable `pth`? What's it size? - type `'whos pth'`

What information does it contain? - type `'open pth'`

What's pth{2} and pth{2}(2)?

In line 41, a variable '`Cost_path`' is with certain size created, but now it's filled with zeros.

In line 44~46, fill in the code that will add up the time cost of each path recursively.

Hint: `i` is the iteration index for different paths, figure out what index `j` represents?

Pseudocode: Cost = Cost + Edge(j,j+1)

line 49: `[MinCost,ind]=min(Cost_path);` To find the shortest path, simply find the one with minimum cost. The output of this function provides both the minimum value and location of that value.

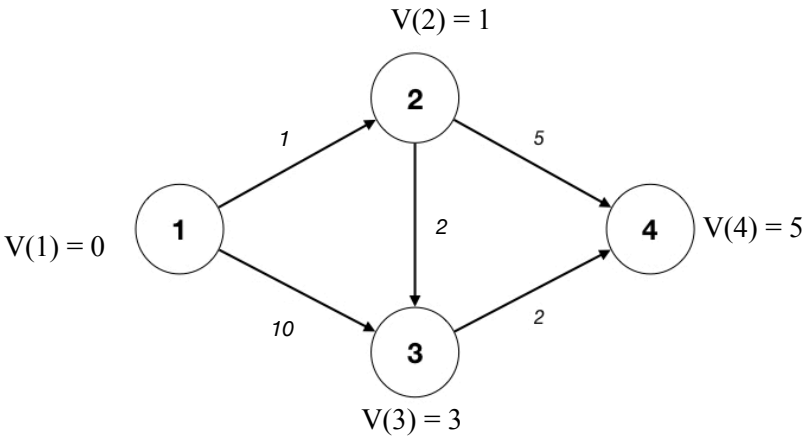
line 51~52: outputs a line in command window. `%d` is used for integer number. `%s` is used for string. Therefore we need to use '`num2str`' function to convert the variable from double to char.

line 54~64: draws multiple figures with all possible paths. The chosen path is highlighted in red.

In line 63, can you write something there so the title of the figure will display the cost of that highlighted path? Hint: title of the figure should be a string. The title should be something like 'path #: cost is ###'

```
Section 3: Find the shortest path and minimum time cost using
dynamic programming
%% Section3 part 1: Calculate the forward value function (V)
```

1) Define the **value function V(i)** as the minimum cost from the starting node (node 1) to node i. The **goal** is to find the path that minimize V(4).
2) At the beginning, we are at node 1, so there's no cost of traveling, V(1) = 0. Initially since we don't know the value of V yet, let's set V(i) = ∞ for i = 2,3,4
3) Update the next value function by adding previous value function and edge cost of traveling from current node to next node. The graph is **topologically ordered**, means only traveling from smaller-value node to larger-value node is allowed. If current state is at node 2, we only need to update V for nodes beyond node 2.
4) Go to next node, update the current node, repeat step 3) until we reach the destination node. Since the value function is calculated from the starting node to ending node, the algorithm is called **forward shortest path algorithm**.



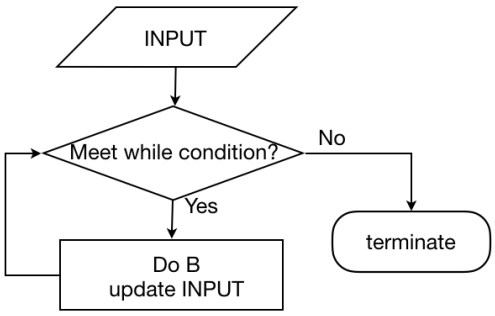
i (current state)	j	$V(i)$	e_{ij}	$V(i)+e_{ij}$	$V(j)$ [before]	is $V(i)+e_{ij} < V(j)$?	$V(j)$ [updated]
1	2	0	1	1	∞	YES	1
	3	0	10	10	∞	YES	10
	4	0	∞	∞	∞	NO	∞
2	3	1	2	3	10	YES	3
	4	1	5	6	∞	YES	6
3	4	3	2	5	6	YES	5

Table 1. update value function step by step from node 1 to 3

```
%% Section3 part 2: Find the path that minimize the cost using
while-loops
```

We have calculate V(4), which is the minimum cost traveling from node 1 to node 4. But we have no idea what the path is for that cost. Note that it always takes a roundtrip to find 1) the cost of shortest path and 2) the actual path. Now let's do that.

Instead of nest of for-loop, here we introduced another recursive loop that goes on and on called **While loop.** It is more time efficient because it will repeat as often as necessary to achieve the goal. While loop is commonly used when the number of loops needs to be excused is unknown.




```
while condition true
   do B;
   update INPUT
end
```

```
example:
i = 1;
text = 'hello world'
while text(i)~=' '
   i = i+1;
end
fprintf('%dth element in
text is space\n', i);
```

We use a nest of while-loops to find the path we want.

Outer while

inner while

a) We start from the destination node, node 4 (initialization: j=4).

b) Starting from node j-1

c) check if i is the nodes V(i)+[cost from node i to j] > V(j)

d) Repeat step (c) until we find the node i that provide V(j).

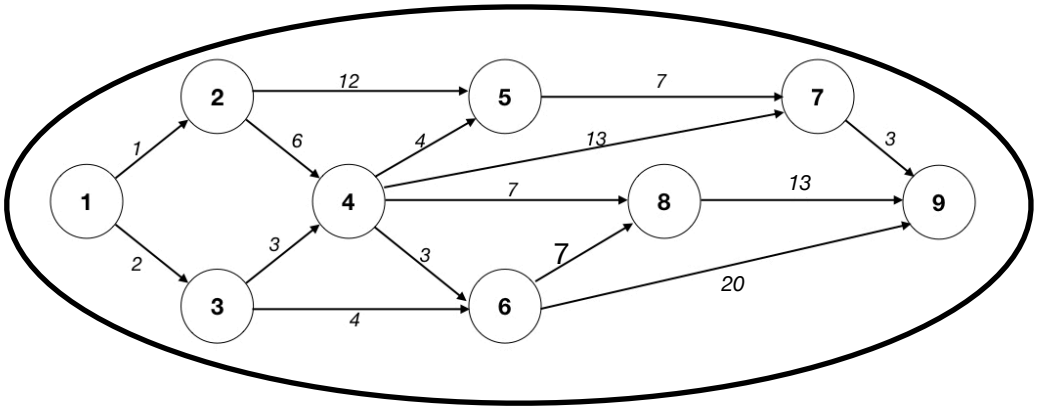
e) travel from node j to node i, update the path by adding node i to the path.

f) Repeat step (b)~(e) until we reach the starting node

Assignment - cell metabolism and drug delivery

Metabolic and biochemical networks are powerful tools for studying and modeling metabolism in various organisms. As metabolic pathways, we consider a series of chemical reactions occurring within a cell at different time points. The main role within a metabolic network is played by the enzymes. Often, enzymes are dependent on other cofactors such as vitamins for proper functioning. You're given a metabolic network of a virus as shown in the figure. Each node indicates one chemical product. Some chemical products rely on the existence of another material. The edge costs denotes the energy cost of the chemical reaction. The initial material is chemical product # 1.

The final products, chemical product #9 will be released and causes disease. When the total cost of producing chemical product #9 from chemical product #1 is no greater or equal to 25 unit, the reaction stops.



Metabolic network inside a virus.

- In assignment, write a script that uses the **forward shorted path algorithm** to calculate the minimum energy cost of producing chemical product #9 from chemical product #1.

- You're also given 3 different medications, each medicine causes increment of energy cost of one edge (shown in table). We can only use one medicine at a time otherwise there will be drug antagonism. Which medicine should we use? Describe how does the value function change with each medicine. Draw the diagram with MATLAB, highlight the updated new shortest path and indicate minimum cost of that path.

	Effective edge	Original energy cost	New cost
medicine 1	7—> 9	3	10
medicine 2	1—> 3	2	10
medicine 3	4—> 5	4	10

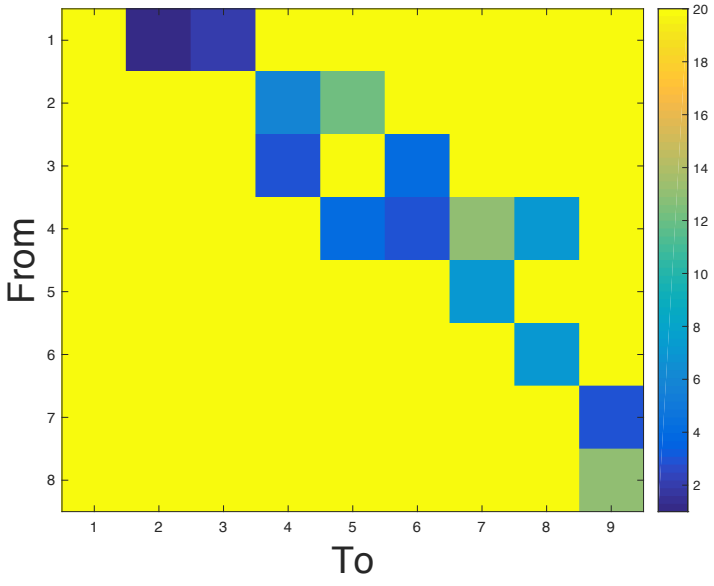
Exercise 1. (5 point)

Not a real report. No need for Introduction, method, discussion or conclusion.

Import the directed graph of the virus into MATLAB. Plot the Edge matrix and answer the following questions.

- What code should we use to get the edge cost from node 4 to 8? Mark that point on the Edge plot as point 1.
- What does Edge(7,9) mean, and what is the value? Mark the point on the Edge plot as point 2.
- What does Edge(5,6) mean, and what is the value? Mark the point on the Edge plot as point 3.
- What does Edge(4,4) mean, and what is the value? Mark the point on the Edge plot as point 4.

Please attache your MATLAB code to the end of the report.



Exercise 2. (5 point)

Again, no need for official report format.

For the virus directed graph, without any coding, can you manually compute the Value function for node 2 through 9? Use similar strategy as table 1 shows. Here's something to begin with…

i (current state)	j	$V(i)$	e_{ij}	$V(i)+e_{ij}$	$V(j)$ [before]	is $V(i)+e_{ij} < V(j)$?	$V(j)$ [updated]
1	2						
	3						
	4						
	5						
	6						
	7						
	8						
	9						
2	3						
	…						
	9						
…							
8	9						

- Please complete and report the table.
- When current node is node 2 and tried targeting node of node 6, what are values of V(1), V(2) … and V(9), before and after? Can you explain why or why not it's changing?
- When current node is node 6 and tried targeting node of node 8, what are values of V(1), V(2) … and V(9), before and after? Can you explain why or why not it's changing?
- What are the final values of V(1), V(2) … and V(9)?