

Laboratory 4 – Zsok Alina-Valentina

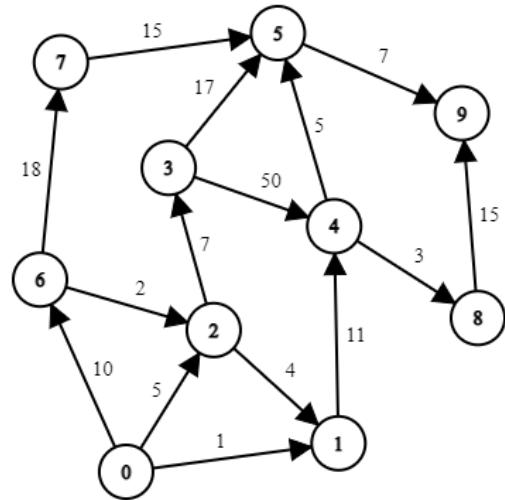
Problem 4 - Write a program that, given a graph with costs, does the following:

- verify if the corresponding graph is a DAG and performs a topological sorting of the activities using the algorithm based on predecessor counters;
- if it is a DAG, finds a highest cost path between two given vertices, in $O(m+n)$.

Topological Sorting Orders:

- 0 6 2 7 1 3 4 5 8 9
- 0 6 7 2 1 3 4 8 5 9
- 0 6 2 7 3 1 4 8 5 9
- 0 6 7 2 3 1 4 5 8 9

Dout – Dictionary Value	
Key	Value(s)
0	[1, 2, 7]
1	[4]
2	[1, 3]
3	[4, 5]
4	[5, 8]
5	[9]
6	[2, 7]
7	[5]
8	[9]
9	[]



It.	current Vertex	predecessorCounting										queue	list									
-	-	0	1	2	3	4	5	6	7	8	9	0										
		0	2	2	1	2	3	1	1	1	2											
1	0	0	1	2	3	4	5	6	7	8	9	6			0							
		0	1	1	1	2	3	0	1	1	2											
2	6	0	1	2	3	4	5	6	7	8	9	2	7		0	6						
		0	1	0	1	2	3	0	0	1	2											
3	2	0	1	2	3	4	5	6	7	8	9	7	1	3	0	6	2					
		0	0	0	0	2	3	0	0	1	2											
4	7	0	1	2	3	4	5	6	7	8	9	1	3		0	6	2	7				
		0	0	0	0	2	2	0	0	1	2											
5	1	0	1	2	3	4	5	6	7	8	9	3			0	6	2	7	1			
		0	0	0	0	1	2	0	0	1	2											
6	3	0	1	2	3	4	5	6	7	8	9	4			0	6	2	7	1	3		
		0	0	0	0	0	1	0	0	1	2											
7	4	0	1	2	3	4	5	6	7	8	9	5	8		0	6	2	7	1	3	4	
		0	0	0	0	0	0	0	0	0	2											
8	5	0	1	2	3	4	5	6	7	8	9	8			0	6	2	7	1	3	4	5
		0	0	0	0	0	0	0	0	0	1											

9	8	0	1	2	3	4	5	6	7	8	9	9		0	6	2	7	1	3	4	5	8	
		0	0	0	0	0	0	0	0	0	0	0											
10	9	0	1	2	3	4	5	6	7	8	9			0	6	2	7	1	3	4	5	8	9
		0	0	0	0	0	0	0	0	0	0												
TOPOLOGICAL SORTING RESULT :													0	6	2	7	1	3	4	5	8	9	

- Highest Cost Path for the Directed Acyclic Graph for source Vertex : 0 and target Vertex : 4.

It.	current Vertex	adjacent Vertex List	current adjacent Vertex	distances									
-	-		-	0	6	2	7	1	3	4	5	8	9
				0	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
1	0	1 2 6	-	0	6	2	7	1	3	4	5	8	9
				0	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
1.1	0	1 2 6	1	0	6	2	7	1	3	4	5	8	9
				0	-∞	-∞	-∞	1	-∞	-∞	-∞	-∞	-∞
1.2	0	1 2 6	2	0	6	2	7	1	3	4	5	8	9
				0	-∞	5	-∞	1	-∞	-∞	-∞	-∞	-∞
1.3	0	1 2 6	6	0	6	2	7	1	3	4	5	8	9
				0	10	5	-∞	1	-∞	-∞	-∞	-∞	-∞
2	6	2 7	-	0	6	2	7	1	3	4	5	8	9
				0	10	5	-∞	1	-∞	-∞	-∞	-∞	-∞
2.1	6	2 7	2	0	6	2	7	1	3	4	5	8	9
				0	10	12	-∞	1	-∞	-∞	-∞	-∞	-∞
2.1	6	2 7	7	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	1	-∞	-∞	-∞	-∞	-∞
3	2	1 3	-	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	1	-∞	-∞	-∞	-∞	-∞
3.1	2	1 3	1	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	-∞	-∞	-∞	-∞	-∞
3.2	2	1 3	3	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	-∞	-∞	-∞	-∞
4	7	5	-	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	-∞	-∞	-∞	-∞
4.1	7	5	5	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	-∞	43	-∞	-∞
5	1	4	-	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	-∞	43	-∞	-∞
5.1	1	4	4	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	27	43	-∞	-∞
6	3	4 5	-	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	27	43	-∞	-∞

6.1	3	4 5	4	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	69	43	-∞	-∞
6.2	3	4 5	5	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	69	43	-∞	-∞
7	4	-	-	0	6	2	7	1	3	4	5	8	9
				0	10	12	28	16	19	69	43	-∞	-∞
HIGHEST COST PATH BETWEEN 0 AND 4:				69									

- Highest Cost Path for the Directed Acyclic Graph for source Vertex : 0 and target Vertex : 2.

It.	current Vertex	adjacent Vertex List	current adjacent Vertex	distances																												
-	-	<table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>									-	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
0	6	2	7	1	3	4	5	8	9																							
0	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞																							
1	0	<table><tr><td>1</td><td>2</td><td>6</td><td></td><td></td><td></td><td></td><td></td></tr></table>	1	2	6						-	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
1	2	6																														
0	6	2	7	1	3	4	5	8	9																							
0	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞																							
1.1	0	<table><tr><td>1</td><td>2</td><td>6</td><td></td><td></td><td></td><td></td><td></td></tr></table>	1	2	6						1	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>-∞</td><td>-∞</td><td>-∞</td><td>1</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	-∞	-∞	-∞	1	-∞	-∞	-∞	-∞	-∞
1	2	6																														
0	6	2	7	1	3	4	5	8	9																							
0	-∞	-∞	-∞	1	-∞	-∞	-∞	-∞	-∞																							
1.2	0	<table><tr><td>1</td><td>2</td><td>6</td><td></td><td></td><td></td><td></td><td></td></tr></table>	1	2	6						2	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>-∞</td><td>5</td><td>-∞</td><td>1</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	-∞	5	-∞	1	-∞	-∞	-∞	-∞	-∞
1	2	6																														
0	6	2	7	1	3	4	5	8	9																							
0	-∞	5	-∞	1	-∞	-∞	-∞	-∞	-∞																							
1.3	0	<table><tr><td>1</td><td>2</td><td>6</td><td></td><td></td><td></td><td></td><td></td></tr></table>	1	2	6						6	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>10</td><td>5</td><td>-∞</td><td>1</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	10	5	-∞	1	-∞	-∞	-∞	-∞	-∞
1	2	6																														
0	6	2	7	1	3	4	5	8	9																							
0	10	5	-∞	1	-∞	-∞	-∞	-∞	-∞																							
2	6	<table><tr><td>2</td><td>7</td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>	2	7							-	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>10</td><td>5</td><td>-∞</td><td>1</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	10	5	-∞	1	-∞	-∞	-∞	-∞	-∞
2	7																															
0	6	2	7	1	3	4	5	8	9																							
0	10	5	-∞	1	-∞	-∞	-∞	-∞	-∞																							
2.1	6	<table><tr><td>2</td><td>7</td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>	2	7							2	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>10</td><td>12</td><td>-∞</td><td>1</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	10	12	-∞	1	-∞	-∞	-∞	-∞	-∞
2	7																															
0	6	2	7	1	3	4	5	8	9																							
0	10	12	-∞	1	-∞	-∞	-∞	-∞	-∞																							
2.2	6	<table><tr><td>2</td><td>7</td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>	2	7							7	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>10</td><td>12</td><td>28</td><td>1</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	10	12	28	1	-∞	-∞	-∞	-∞	-∞
2	7																															
0	6	2	7	1	3	4	5	8	9																							
0	10	12	28	1	-∞	-∞	-∞	-∞	-∞																							
3	2	-	-	<table><tr><td>0</td><td>6</td><td>2</td><td>7</td><td>1</td><td>3</td><td>4</td><td>5</td><td>8</td><td>9</td></tr><tr><td>0</td><td>10</td><td>12</td><td>28</td><td>1</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td><td>-∞</td></tr></table>	0	6	2	7	1	3	4	5	8	9	0	10	12	28	1	-∞	-∞	-∞	-∞	-∞								
0	6	2	7	1	3	4	5	8	9																							
0	10	12	28	1	-∞	-∞	-∞	-∞	-∞																							
HIGHEST COST PATH BETWEEN 0 AND 2 :				12																												