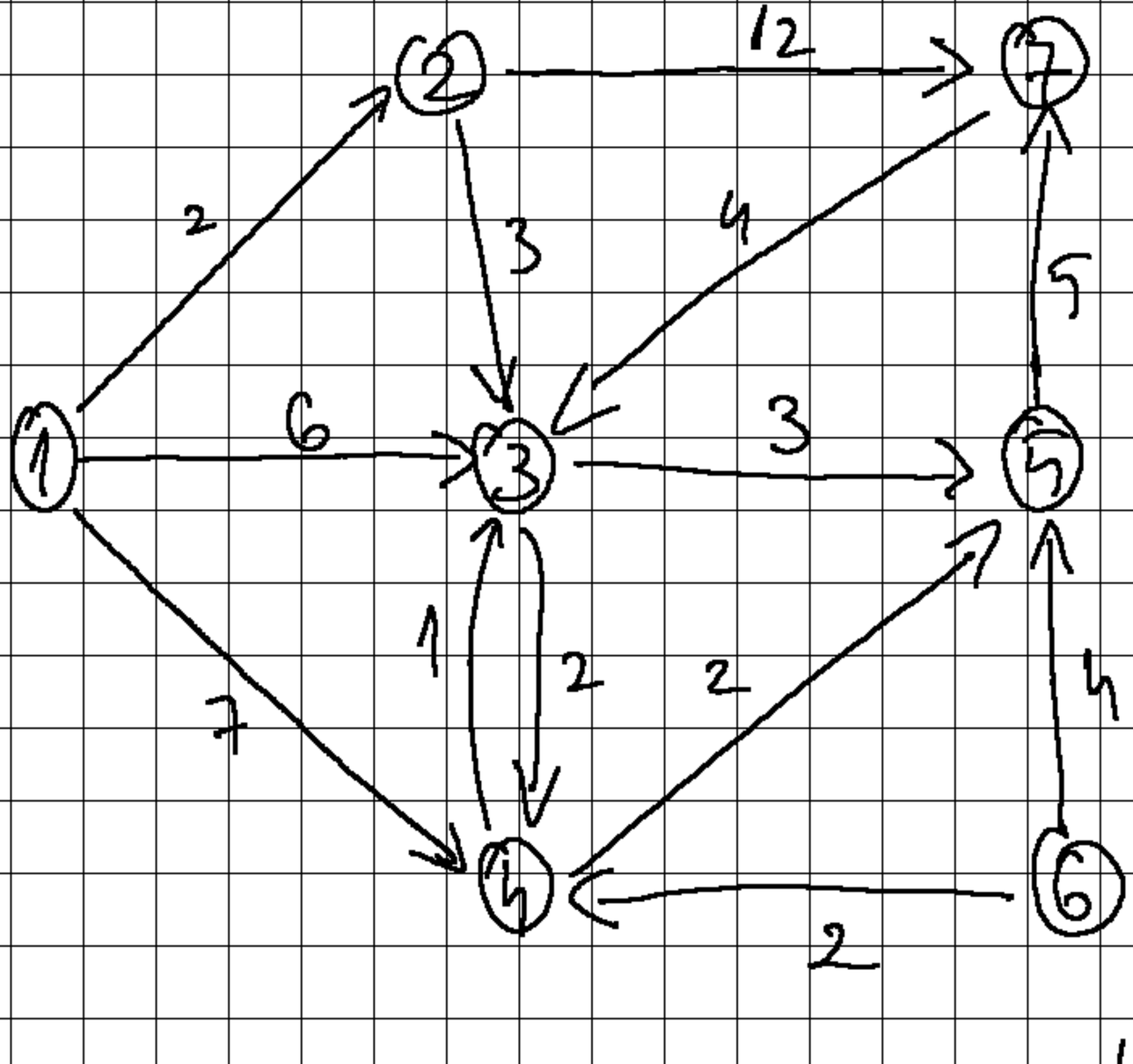
Exam Sample:

1. In the diagraph below, find the lowest cost path from all vertices to vertex 7, using the Dijkstra alg. in reverse.



1: 1→2→3→5→7 (cost 13)

2: 2→3→5→7 (cost 11)

3: 3→5→7 (cost 8)

4: 4→5→7 (cost 7)

5: 5→7 (cost 5)

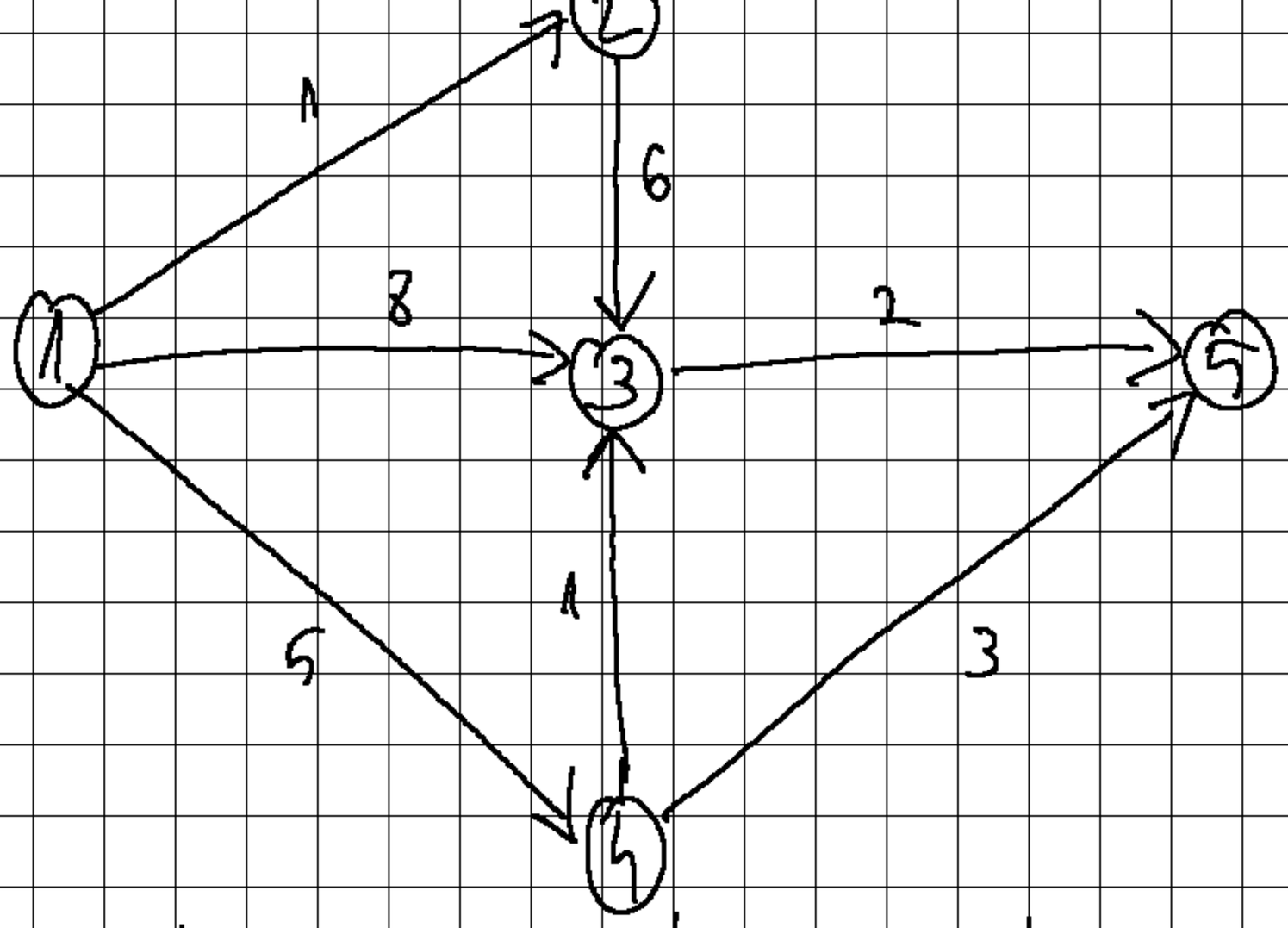
6: 6→5→7 (cost 9)

7: 7 (cost 0)

x	q	d							next						
		1	2	3	4	5	6	7	1	2	3	4	5	6	7
	7	∞	∞	∞	∞	∞	∞	0	-	-	-	-	-	-	-
7	2,5	∞	12	∞	∞	7	∞	0	-	7	-	-	7	-	-
5	2,3,4,6	∞	12	8	7	5	9	0	-	7	5	5	7	5	-
4	1,2,3,6	14	12	8	7	5	9	0	4	7	5	5	7	5	-
3	1,2,6	14	11	8	7	5	9	0	4	3	5	5	7	5	-
6	1,2	14	11	8	7	5	9	0	4	3	5	5	7	5	-
2	1	13	11	8	7	5	9	0	2	3	5	5	7	5	-
1		13	11	8	7	5	9	0	2	3	5	5	7	5	-

Exam 2018. Subj. no. 9

1. In the diagraph below, find the minimum cost path from all vertices to vertex 5, using the Dijkstra algorithm in reverse.



1: 1→4→5 (cost 8)

2: 2→3→5 (cost 8)

3: 3→5 (cost 2)

4: 4→5 (cost 3)

5: 5 (cost 0)

x	S	d					next				
		1	2	3	4	5	1	2	3	4	5
	5	∞	∞	∞	∞	0	–	–	–	–	–
5	3,4	∞	∞	2	3	∞	–	–	5	5	–
3	1,2,4	10	8	2	3	∞	3	3	5	5	–
4	1,2	8	8	2	3	∞	4	3	5	5	–
2	1	8	8	2	3	∞	4	3	5	5	–
1		8	8	2	3	∞	4	3	5	5	–

2. Considering the following activities, determine the earliest and the latest schedulings, and the critical activities (show the step by step computations for the topological sorting, then for the earliest and latest starting and ending times)

Act.	Duration	Prerequisites
A	1	—
B	4	A, D, E
C	1	B, F
D	3	—
E	2	—
F	5	E

⇒ sorting is

A, D, E, B, F, C

Step	Nodes remaining	Node processed	In-degrees A	B	C	D	E	F	queue
			0	3	2	0	0	1	A, D, E
1	A, B, C, D, E, F	—							
2	B, C, D, E, F	A	-	2	2	0	0	1	D, E
3	B, C, E, F	D	-	1	2	-	0	1	E
4	B, C, F	E	-	0	2	-	-	0	B, F
5	C, F	B	-	-	1	-	-	0	F
6	C	F	-	-	0	-	-	-	C
7	—	C	-	-	-	-	-	-	-

Earliest start times

A: 0

D: 0

E: 0

B: 3

F: 2

C: 7

Earliest finish times

A: 0 + 1 = 1

D: 0 + 3 = 3

E: 0 + 2 = 2

B: 3 + 4 = 7

F: 2 + 5 = 7

C: 7 + 1 = 8

Latest start times

C: 8 − 1 = 7

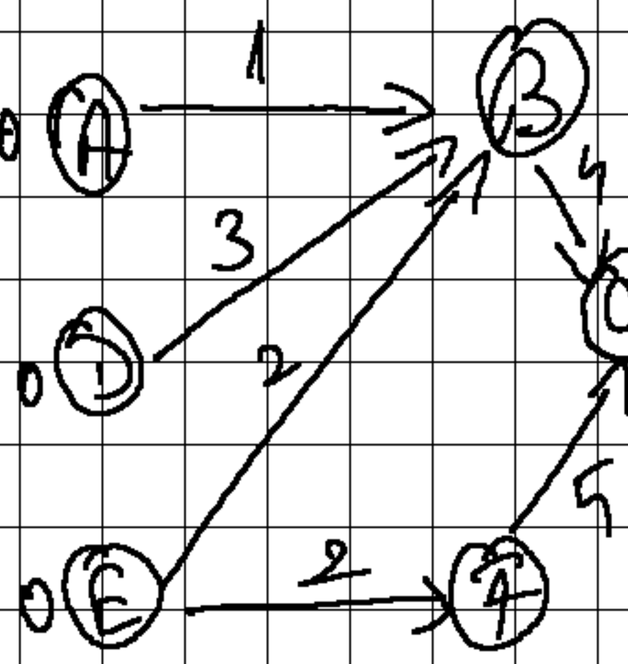
F: 7 − 5 = 2

B: 7 − 4 = 3

E: 2 − 2 = 0

D: 3 − 3 = 0

A: 3 − 1 = 2



Latest finish times

A: 3

D: 3

E: 2

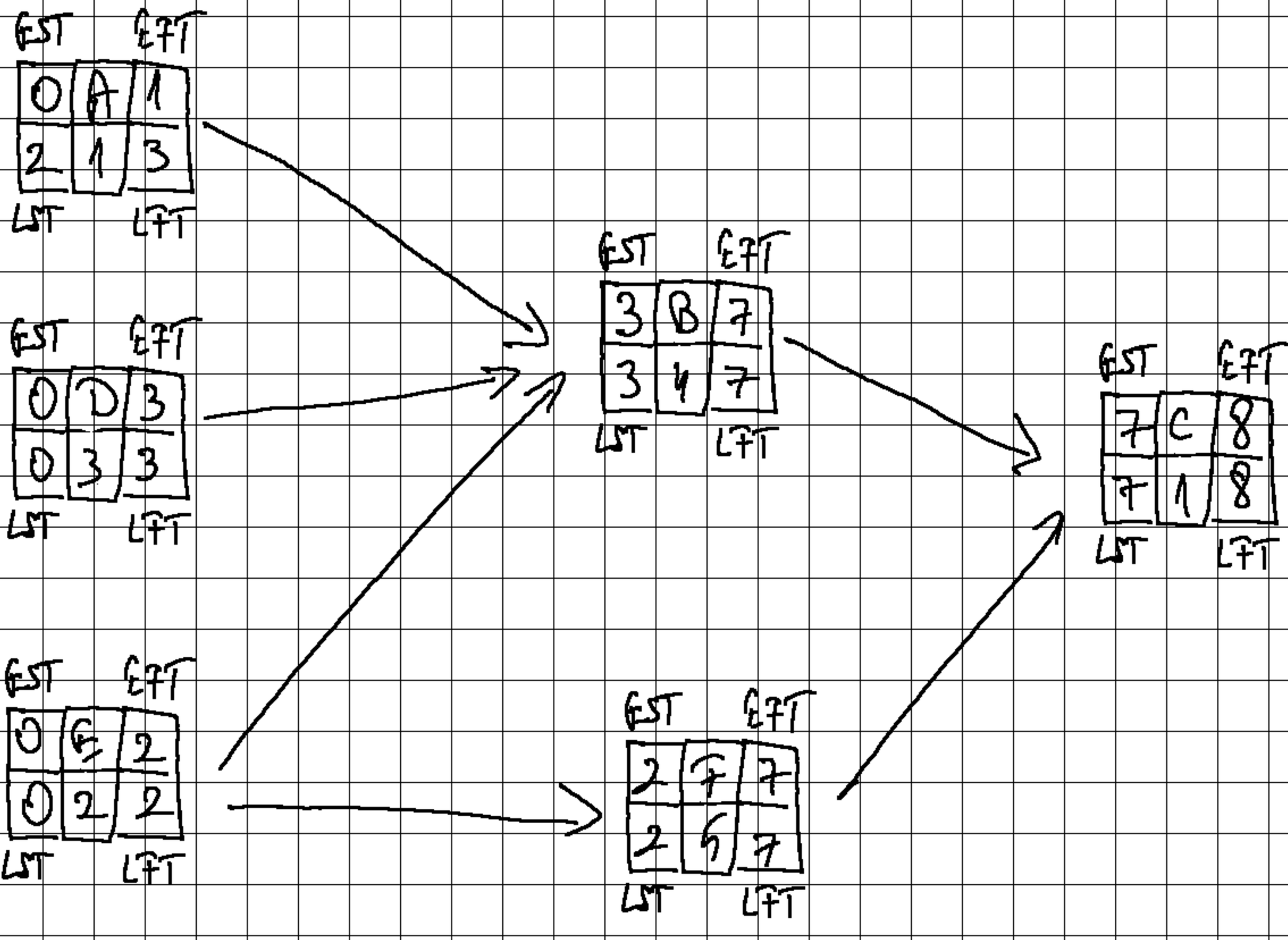
B: 7

F: 7

C: 8

Project completion time is 8

Critical activities: D, E, B, F, C (EST = LST)

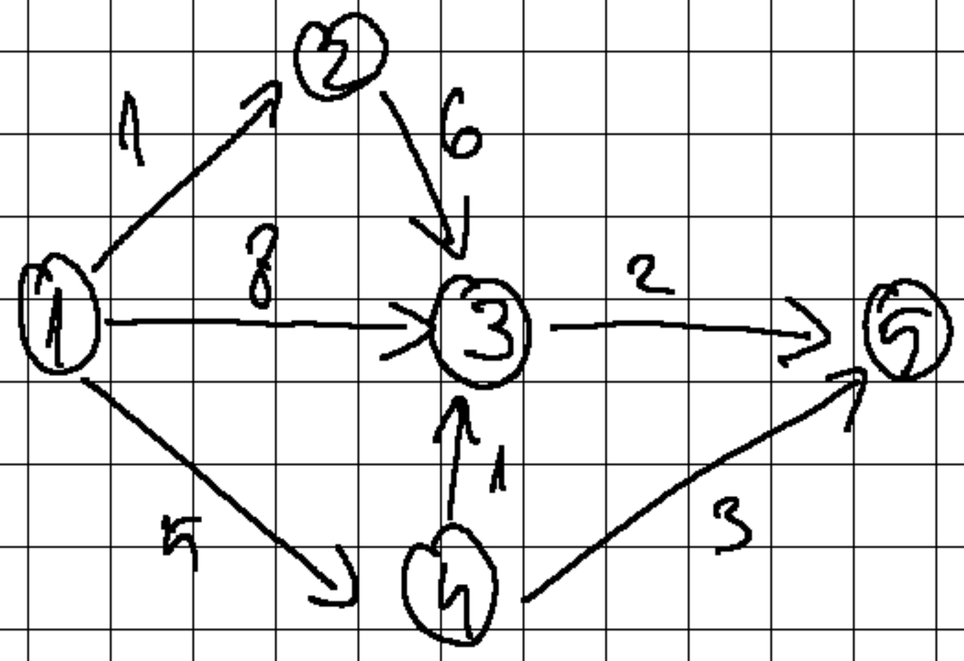

EST
LFT
0
A
1
2
1
3
LST
LFT
EST
LFT
0
D
3
0
3
3
LST
LFT
EST
LFT
3
B
7
3
4
7
LST
LFT
EST
LFT
7
C
8
7
1
8
LST
LFT
EST
LFT
0
E
2
0
2
2
LST
LFT
EST
LFT
2
F
7
2
5
7
LST
LFT

3. Write a polynomial time algorithm that, given a directed graph and a pair of vertices s and t, finds the number of distinct paths of minimum length from s to t.

```
def count_shortest_paths(graph, s, t)
    Q = deque([s])
    distance = defaultdict(lambda: float('inf'))
    distance[s] = 0
    count = defaultdict(int)
    count[s] = 1
    while Q:
        u = Q.popleft()
        for v in graph[u]:
            if distance[v] == float('inf'):
                distance[v] = distance[u] + 1
                Q.append(v)
            if distance[v] == distance[u] + 1:
                count[v] += count[u]
    return count[t]
```

Exam 2018. Subject no. 5

1. In the diagraph below, find the minimum cost path from all vertices to vertex 5, using the Dijkstra algorithm in reverse.



1: 1→4→5 (cost 8)

2: 2→3→5 (cost 7)

3: 3→5 (cost 2)

4: 4→5 (cost 3)

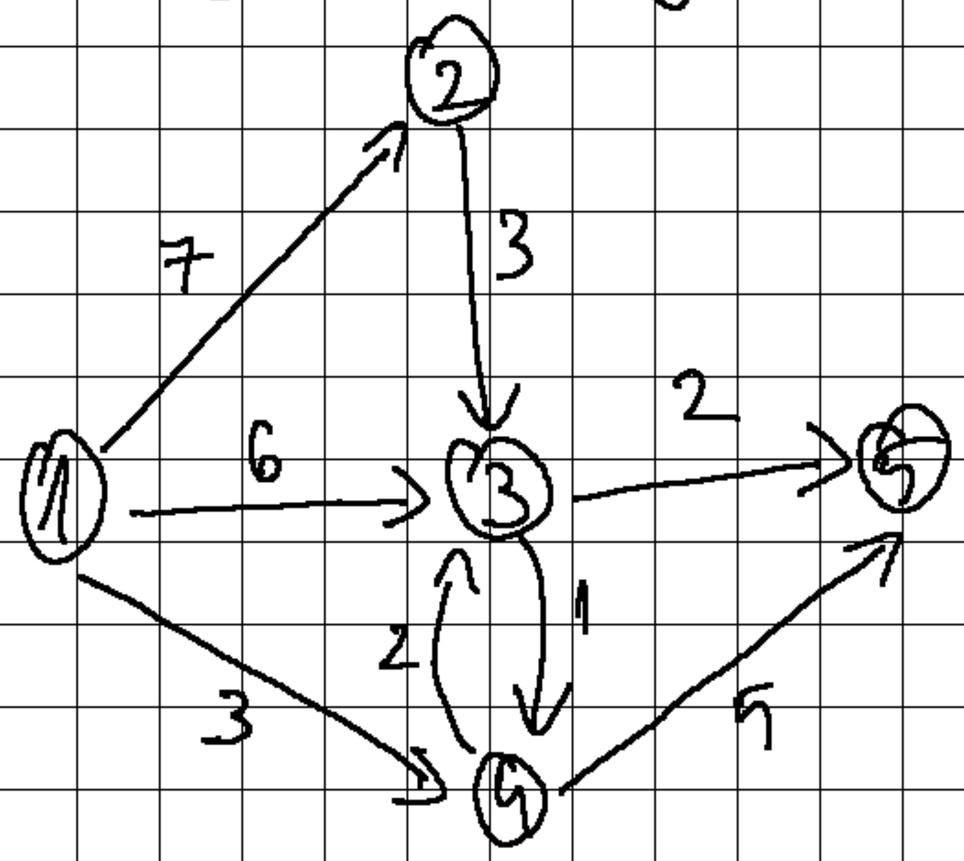
5: 5 (cost 0)

x	S	dist 1	dist 2	dist 3	dist 4	dist 5	next 1	next 2	next 3	next 4	next 5
	5	∞	∞	∞	∞	0	-	-	-	-	-
5	3,4	∞	∞	2	3	0	-	-	5	5	-
3	1,2,4	10	8	2	3	0	3	3	5	5	-
4	1,2	8	8	2	3	0	4	3	5	5	-
2	1	8	8	2	3	0	4	3	5	5	-
1	-	8	8	2	3	0	4	3	5	5	-

The rest of the ex. are the same as the prec. sample

Exam 2018. Subject no. 2.

1. In the diagraph below, find the minimum cost path from vertex 1 to all other vertices, using the Dijkstra algorithm.



1: 1 (cost 0)

2: 1 → 2 (cost 7)

3: 1 → 4 → 3 (cost 5)

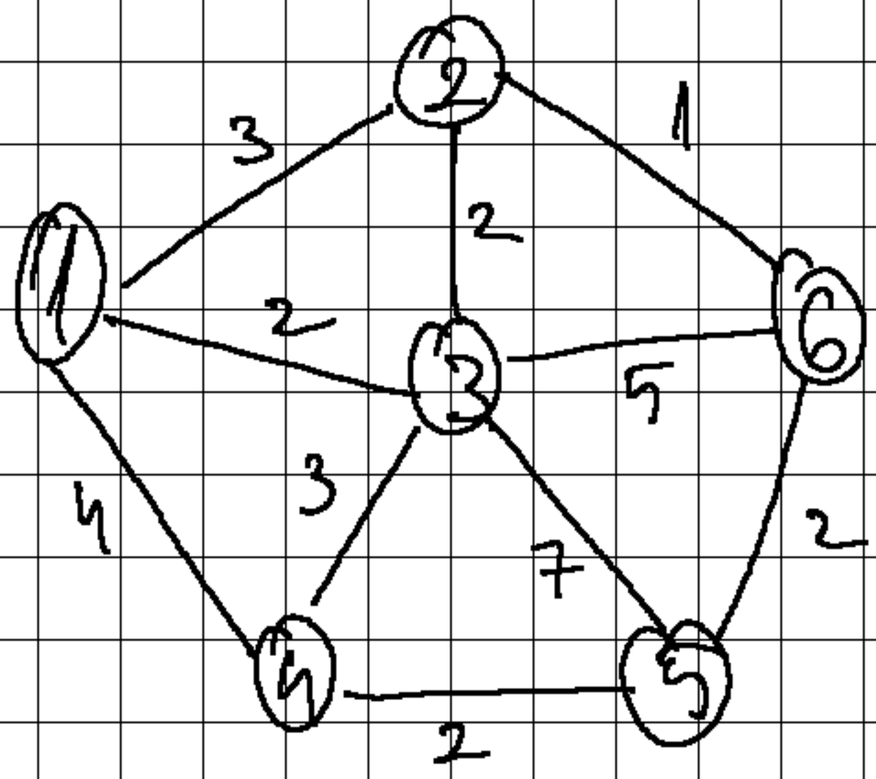
4: 1 → 4 (cost 3)

5: 1 → 4 → 3 → 5 (cost 7)

X	Q
	1
1	2, 3, 4
4	2, 3, 5
3	2, 5
5	2
2	–

dist					previous				
1	2	3	4	5	1	2	3	4	5
0	∞	∞	∞	∞	–	–	–	–	–
0	7	6	3	∞	–	1	1	1	–
0	7	5	3	8	–	1	4	1	4
0	7	5	3	7	–	1	4	1	3
0	7	5	3	7	–	1	4	1	3
0	7	5	3	7	–	1	4	1	3

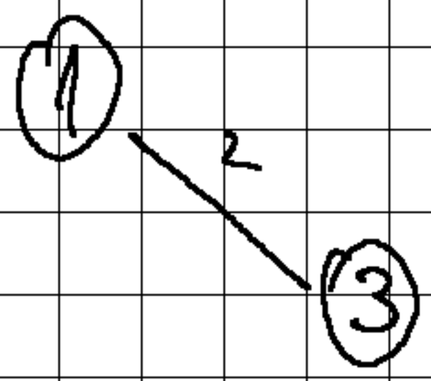
2. In the following graph, find the minimum Spanning tree, using Prim's algorithm



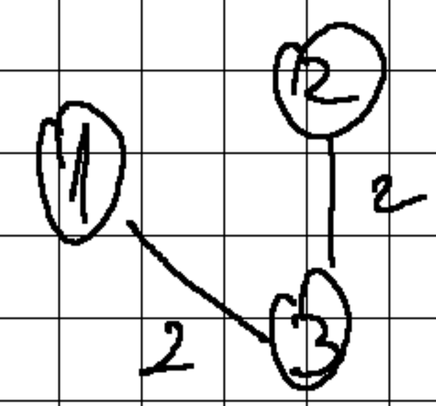
visited: { 1, 3, 2, 6, 5 }

I

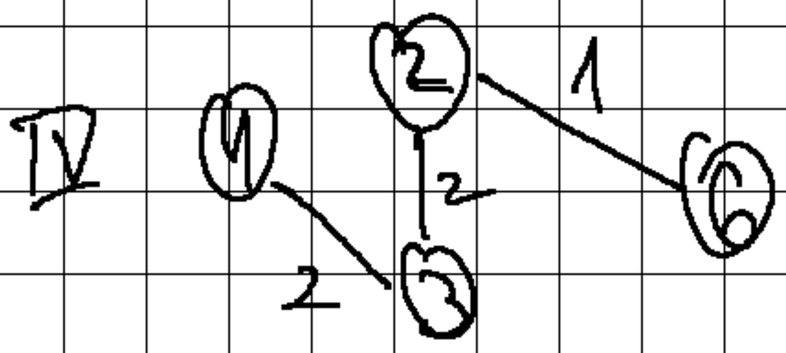
II



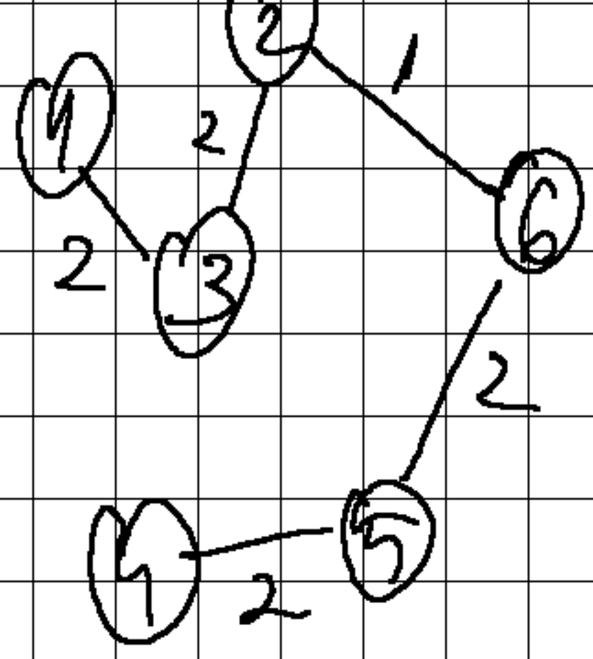
III



IV



IV



V

3. Design an algorithm that, given a directed graph, finds a cycle of minimum length, if one exists.

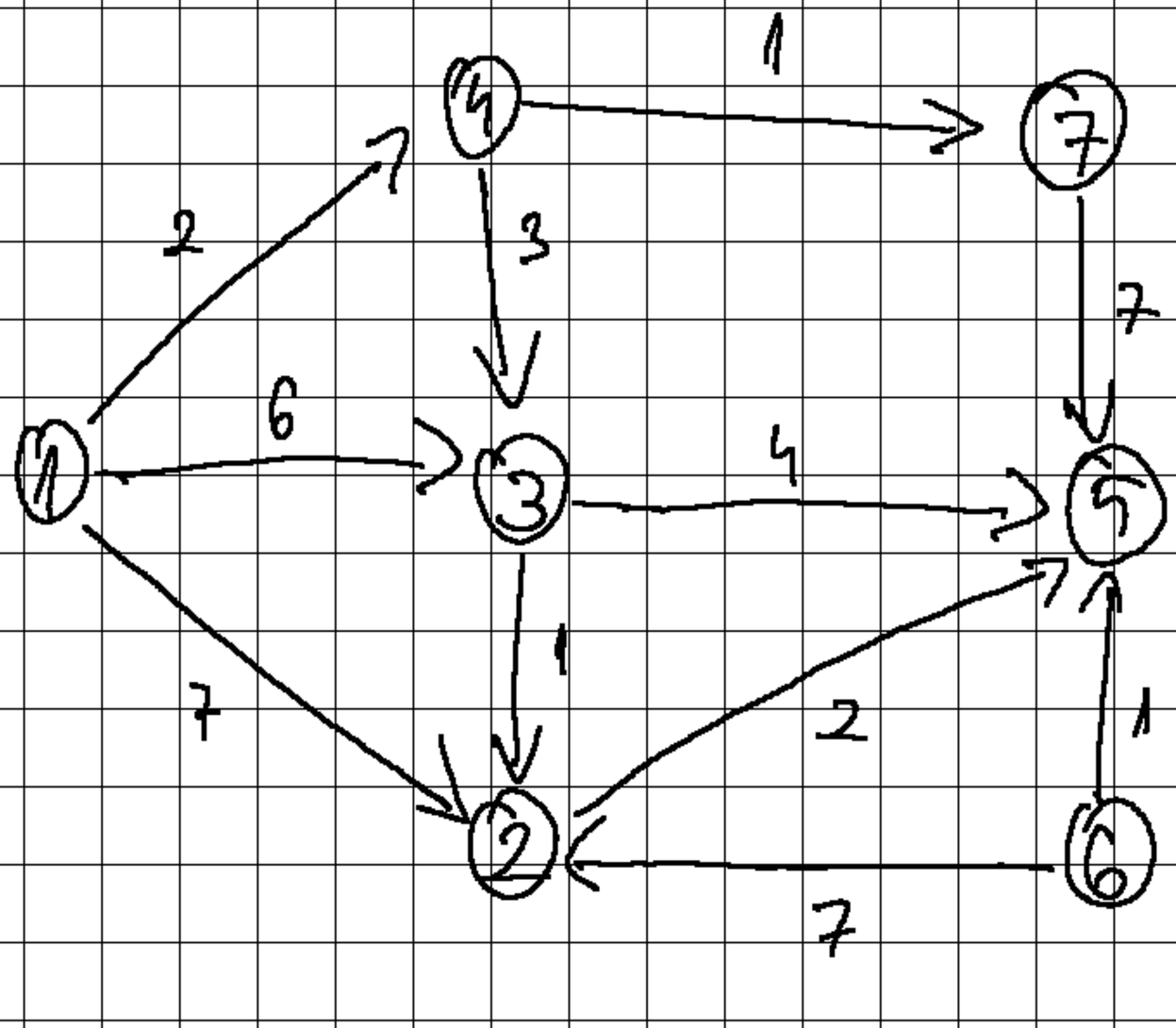
```
def floyd_warshall(graph):
    v = len(graph)
    dist = [[float(inf)] * v for _ in range(v)]
    for i in range(v):
        for j in range(v):
            if graph[i][j] != 0
                dist[i][j] = graph[i][j]
    for k in range(v):
        for i in range(v):
            for j in range(v):
                if dist[i][j] > dist[i][k] + dist[k][j]
                dist[i][j] = dist[i][k] + dist[k][j]
    if i == j and dist[i][k] + dist[k][j] <
                < dist[i][i]:
    dist[i][i] = dist[i][k] + dist[k][i]
```

min_cycle_length = min(dist[i][i] for i in
range(n))

return min_cycle_length if min_cycle_length !=
!= float('inf') else "No cycle exists".

Exam 2019 no. 1.

1. in the diagraph below, find the minimum cost path from vertex 1 to all other vertices, using the Dijkstra algorithm



1: 1 (cost 0)

2: 1 → 4 → 3 → 2 (cost 6)

3: 1 → 4 → 3 (cost 5)

4: 1 → 4 (cost 2)

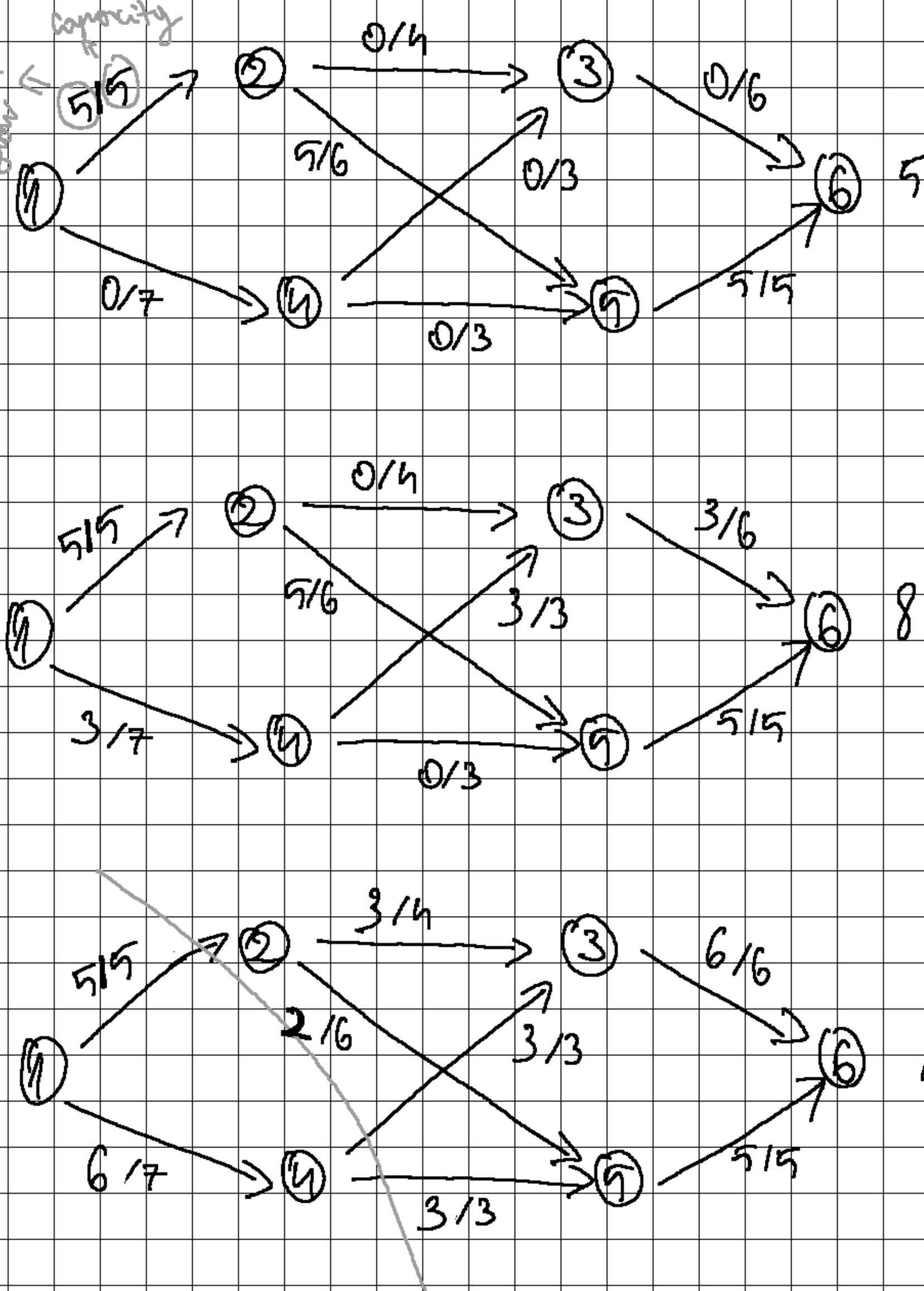
5: 1 → 4 → 3 → 2 → 5 (cost 8)

6: —

7: 1 → 4 → 7 (cost 3)

X	2	dist							previous						
		1	2	3	4	5	6	7	1	2	3	4	5	6	7
	1	0	∞	∞	∞	∞	∞	∞	-	-	-	-	-	-	-
1	2,3,4	0	7	6	2	∞	∞	∞	-	1	1	1	-	-	-
4	2,3,7	0	7	5	2	∞	∞	3	-	1	4	1	-	-	4
7	2,3,5	0	7	5	2	10	∞	3	-	1	4	1	7	-	4
3	2,5	0	6	5	2	9	∞	3	-	3	4	1	3	-	4
2	5	0	6	5	2	8	∞	3	-	3	4	1	2	-	4
5	—	0	6	5	2	8	∞	3	-	3	4	1	2	-	4

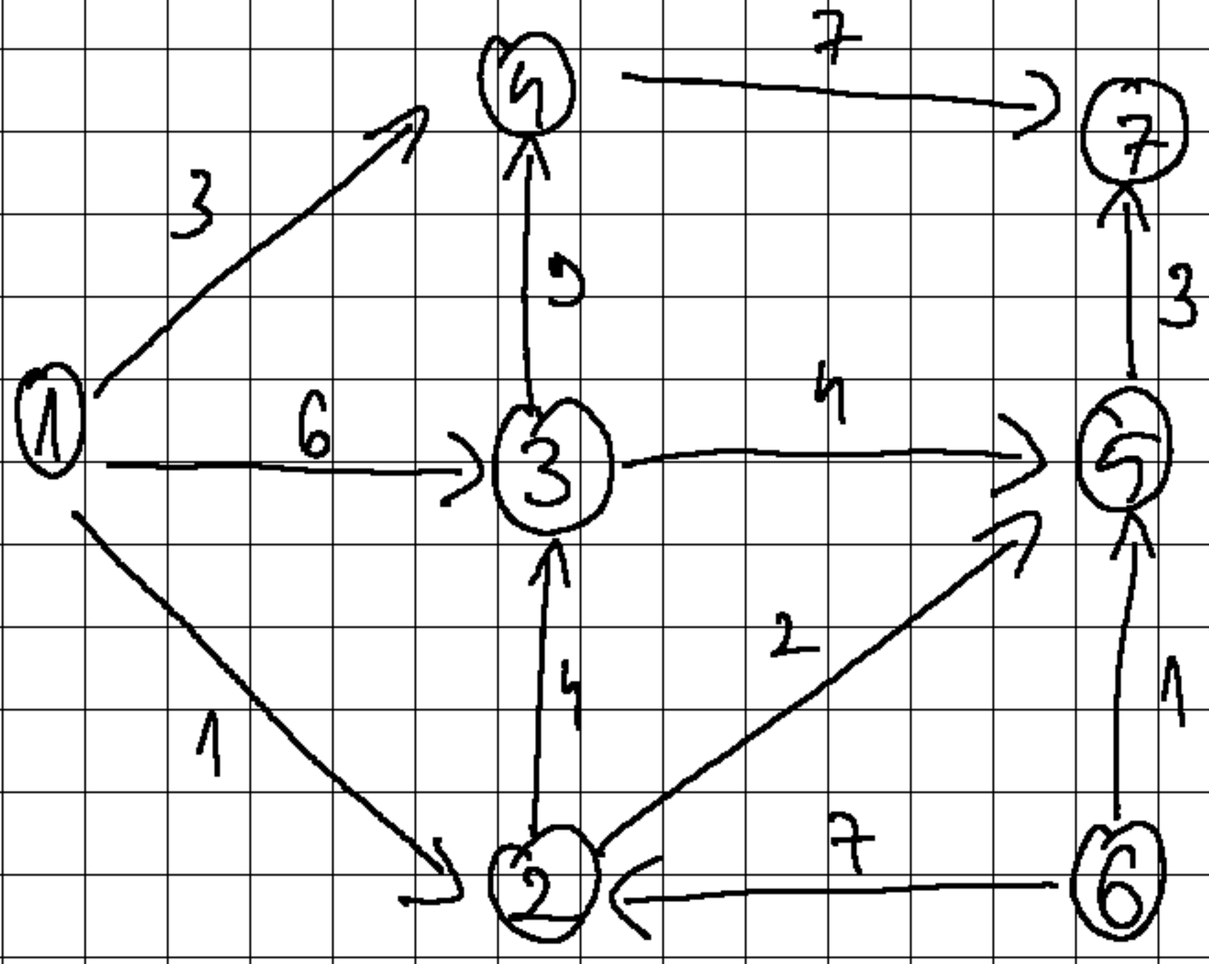
2. in the following graph, find the maximum flow from vertex 1 to 6. Start from the given flow and maximize it using Ford-Fulkerson algorithm. Also show the minimum cut.



The cut $\{1,4\}$ to $\{2,3,5,6\}$ is saturated (capacity = 11, flow = 11)

Exam 2023 no.2

1. in the diagraph below, find the minimum cost paths from vertex 1 to each of the vertices, using the Dijkstra algorithm.



1: 1 (cost 0)

2: 1→2 (cost 1)

3: 1⇒2⇒3 (cost 5)

4: 1→4 (cost 3)

5: 1⇒2⇒5 (cost 3)

6: -

7: 1⇒2⇒5⇒7 (cost 6)

X	2	distance							previous						
		1	2	3	4	5	6	7	1	2	3	4	5	6	7
	1	0	∞	∞	∞	∞	∞	∞	-	-	-	-	-	-	-
1	2,3,4	0	1	6	3	∞	∞	∞	-	1	1	1	-	-	-
2	3,4,5	0	1	5	3	3	∞	∞	-	1	2	1	2	-	-
4	3,5	0	1	5	3	3	∞	10	-	1	2	1	2	-	4
5	3	0	1	5	3	3	∞	6	-	1	2	1	2	-	5
3	-	0	1	5	3	3	∞	6	-	1	2	1	2	-	5

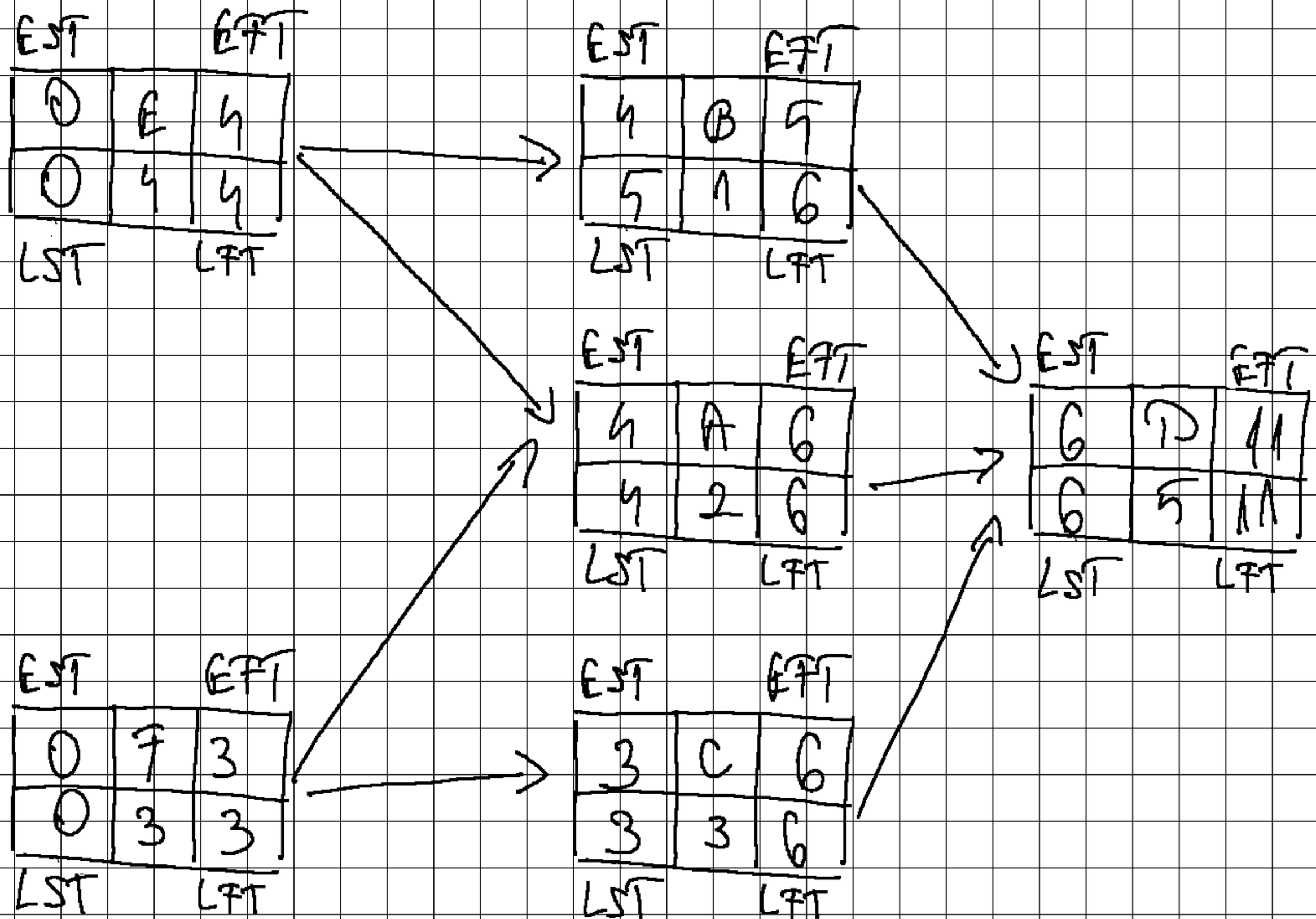
2. Considering the following activities, determine the earliest and the latest schedulings, and the critical activities (show the steps by step computations for the topological sorting, then for the startings and ending times)

Act.	Duration	Prerequisites
A	2	E, F
B	1	E
C	3	F
D	5	A, B, C
E	4	—
F	3	—

Sorting is

E, F, B, A, C, D

Step	Nodes remaining	Node processed	in-degrees A	B	C	D	E	F	queue
1	A, B, C, D, E, F	—	2	1	1	3	0	0	E, F
2	A, B, C, D, F	E	1	0	1	3	—	0	F, B
3	A, B, C, D	F	0	0	0	3	—	—	B, A, C
4	A, C, D	B	0	—	0	2	—	—	A, C
5	C, D	A	—	—	0	1	—	—	C
6	D	C	—	—	—	0	—	—	D
7	—	D	—	—	—	—	—	—	—

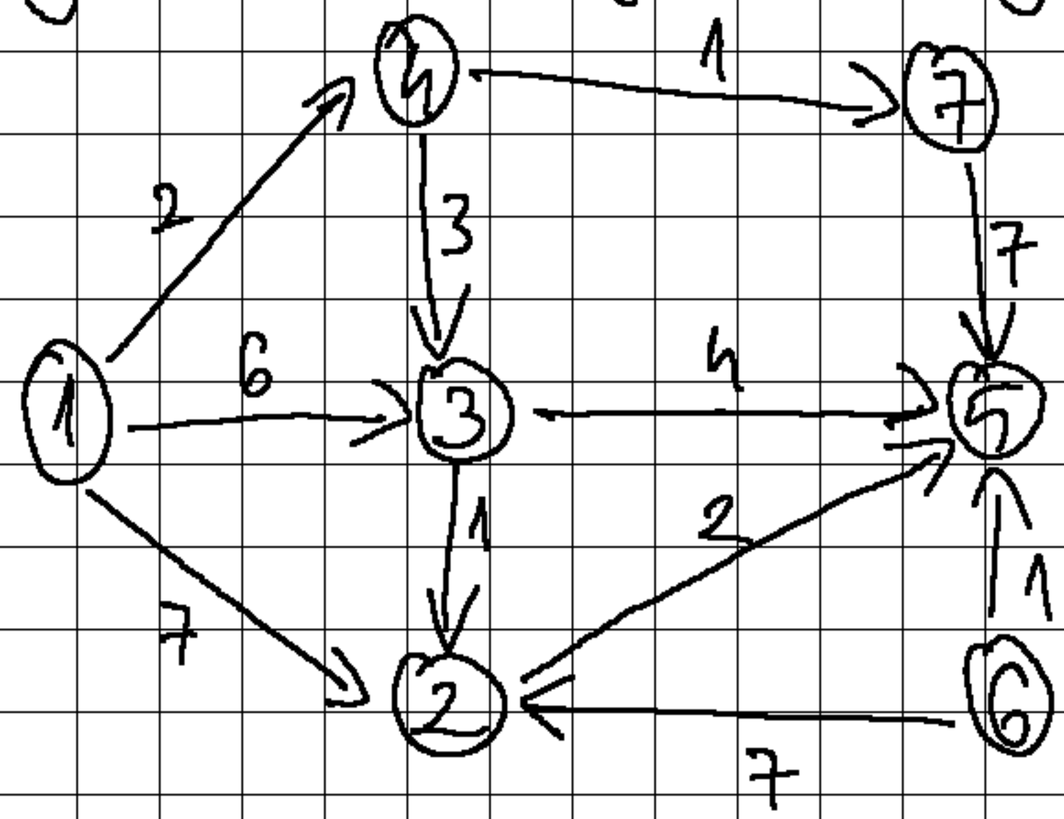


Critical activities: (EST = LST)

D, C, A, E, F

Exam 2023. no. 1:

1. in the diagraph below, find the minimum cost paths from vertex 1 to each of the vertices, using the Dijkstra algorithm.



1: 1 (cost 0)
2: 1→4→3→2 (cost 6)
3: 1→4→3 (cost 5)
4: 1→4 (cost 2)
5: 1→4→3→2→5 (cost 8)
6: —
7: 1→4→7 (cost 3)

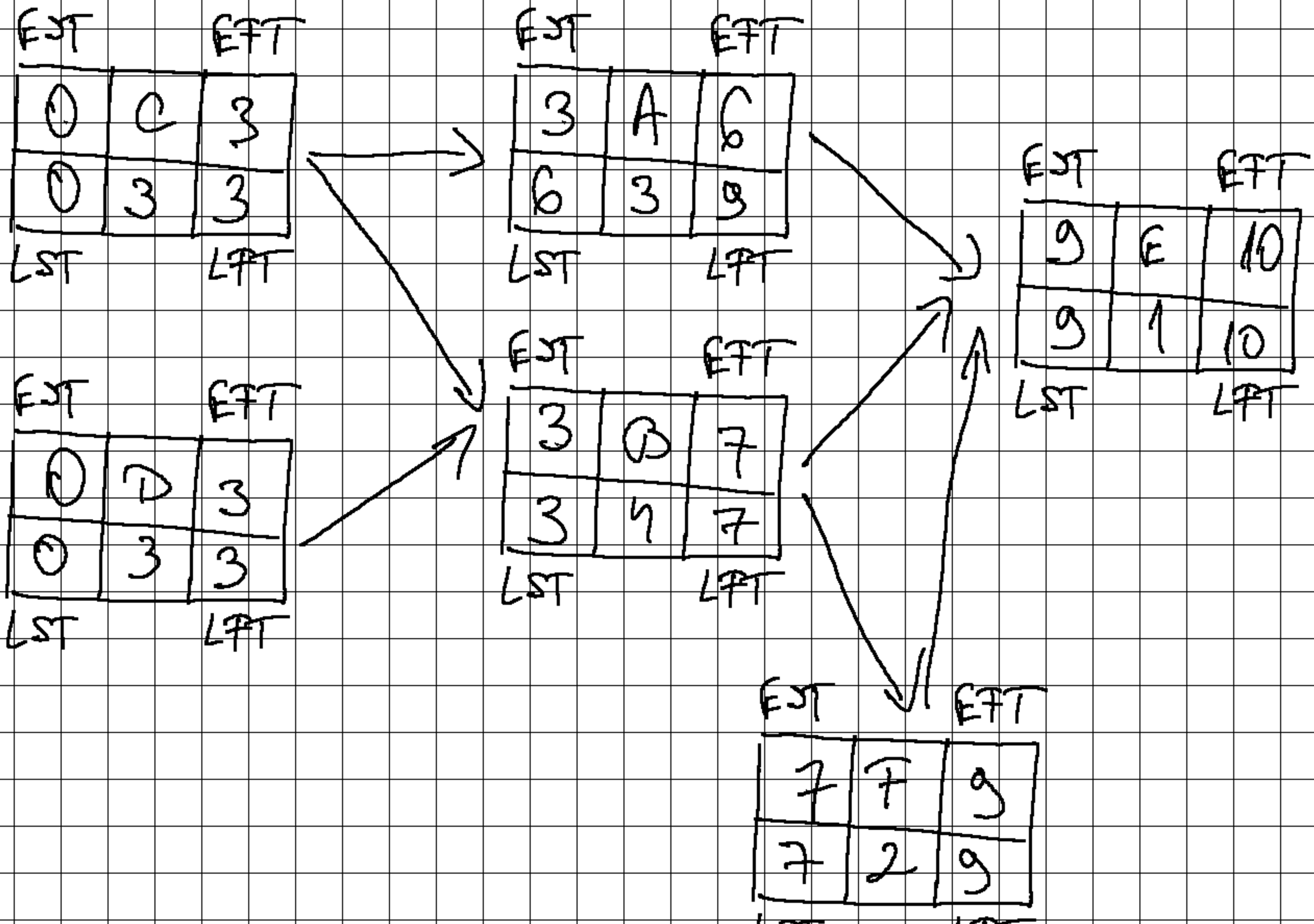
x	S	distance							previous						
		1	2	3	4	5	6	7	1	2	3	4	5	6	7
	1	0	∞	∞	∞	∞	∞	∞	–	–	–	–	–	–	–
1	2,3,4	0	7	6	2	∞	∞	∞	–	1	1	1	–	–	–
4	2,3,7	0	7	5	2	∞	∞	3	–	1	4	1	–	–	4
7	2,3,5	0	7	5	2	10	∞	3	–	1	4	1	7	–	4
3	2,5	0	6	5	2	9	∞	3	–	3	4	1	3	–	4
2	5	0	6	5	2	8	∞	3	–	3	4	1	2	–	4
5	–	0	6	5	2	8	∞	3	–	3	4	1	2	–	4

2. Considering the following activities, determine the earliest and the latest schedulings, and the critical activities (show the step-by-step computations for the topological sorting, then for starting and ending times).

Act	Duration	Prerequisites
A	3	C
B	4	C, D
C	3	—
D	3	—
E	1	A, B, F
F	2	B

Sorting is:
C, D, A, B, F, E

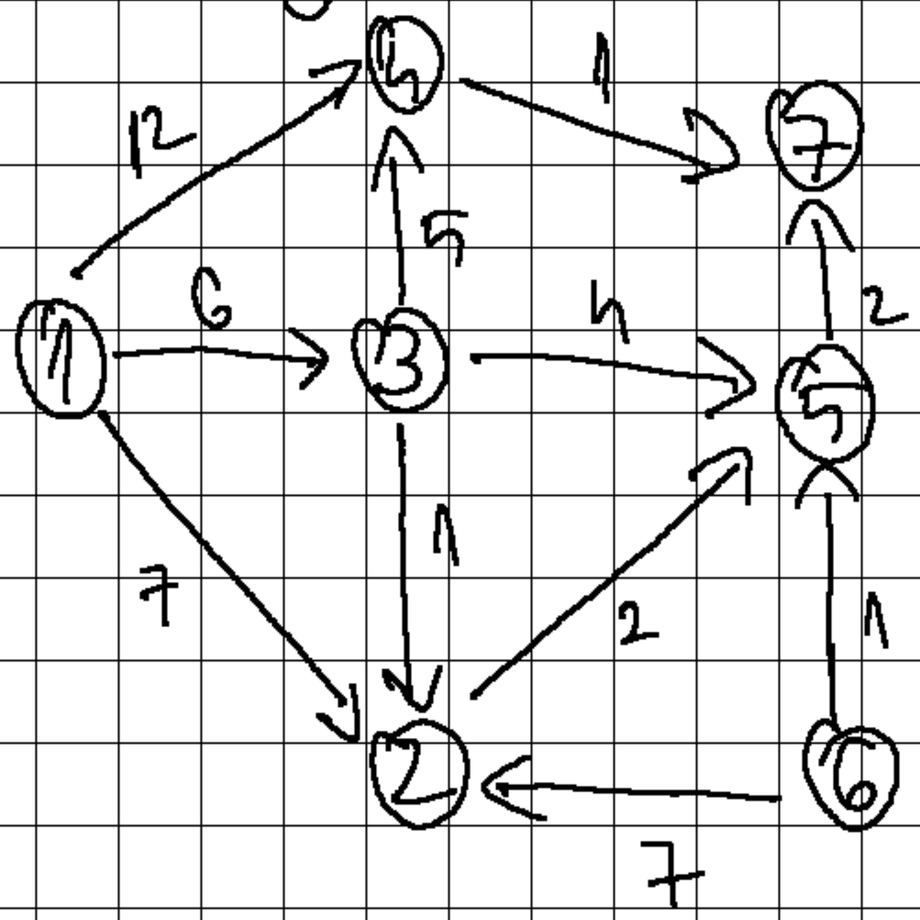
Step	Nodes remaining	Node processed	in degree A	B	C	D	E	F	Queue
1	A, B, C, D, E, F	—	1	2	0	0	3	1	C, D
2	A, B, D, E, F	C	0	1	–	0	3	1	D, A
3	A, B, E, F	D	0	0	–	–	3	1	A, B
4	B, E, F	A	–	0	–	–	2	1	B
5	E, F	B	–	–	–	–	1	0	F
6	E	F	–	–	–	–	0	–	E
7	–	E	–	–	–	–	–	–	–



Critical activities: (EST = LST): C, D, B, F, E

Exam 2023. no 3.

1. in the diagraph below, find the minimum cost paths from all vertices to vertex 7, using the Dijkstra algorithm in reverse.



1: 1→2→5→7 (cost 11)

2: 2→5→7 (cost 4)

3: 3→2→5→7 (cost 5)

4: 4→7 (cost 1)

5: 5→7 (cost 2)

6: 6→5→7 (cost 3)

7: 7 (cost 0)

X	S	distance							next						
		1	2	3	4	5	6	7	1	2	3	4	5	6	7
	7	∞	∞	∞	∞	∞	∞	0	-	-	-	-	-	-	-
7	4,5	∞	∞	∞	1	2	∞	0	-	-	-	7	7	-	-
4	1,3,5	13	∞	6	1	2	∞	0	4	-	4	7	7	-	-
5	1,3,6,2	13	4	6	1	2	3	0	4	5	4	7	7	5	-
6	1,3,2	13	4	6	1	2	3	0	4	5	4	7	7	5	-
2	1,3	11	4	5	1	2	3	0	2	5	2	7	7	5	-
3	1	11	4	5	1	2	3	0	2	5	2	7	7	5	-
1	-	11	4	5	1	2	3	0	2	5	2	7	7	5	-

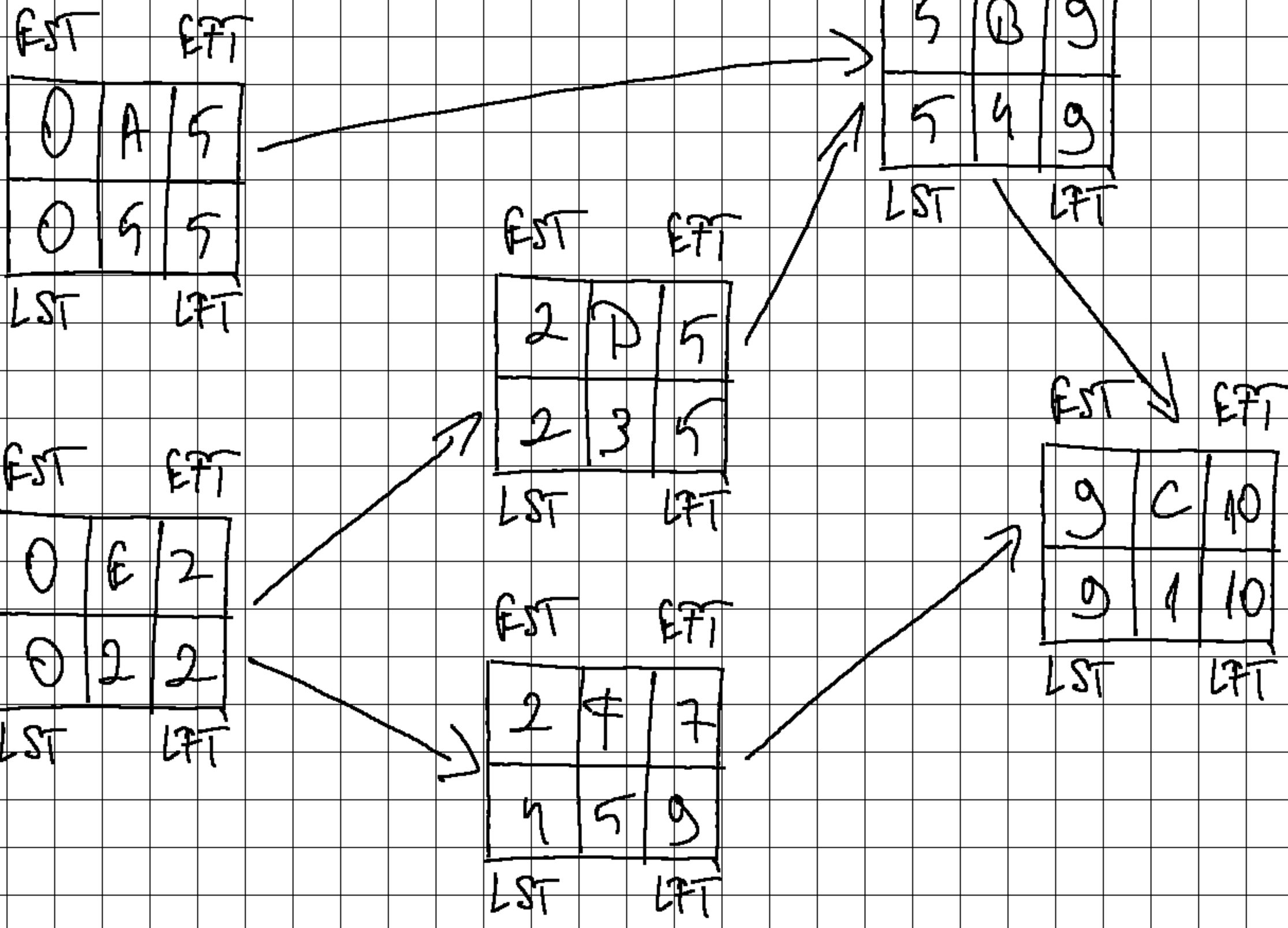
2. Consider the following activities, determine the earliest and the latest scheduling, and the critical activities (show the step by step computations for the topological sorting, then for the starting and ending times)

Act	Duration	Prerequisites
A	5	
B	4	A, D
C	1	B, F
D	3	E
E	2	—
F	5	E

Sorting is

A E D F B C

Step	Remaining nodes	Node processed	in-degree A	B	C	D	E	F	queue
1	A, B, C, D, E, F	—	0	2	2	1	0	1	A, E
2	B, C, D, F	A	—	1	2	1	0	1	E
3	B, C, D, F	E	—	1	2	0	—	0	D, F
4	B, C, F	D	—	0	2	—	—	0	F, B
5	B, C	F	—	0	1	—	—	—	B
6	C	B	—	—	0	—	—	—	C
7	—	C	—	—	—	—	—	—	—

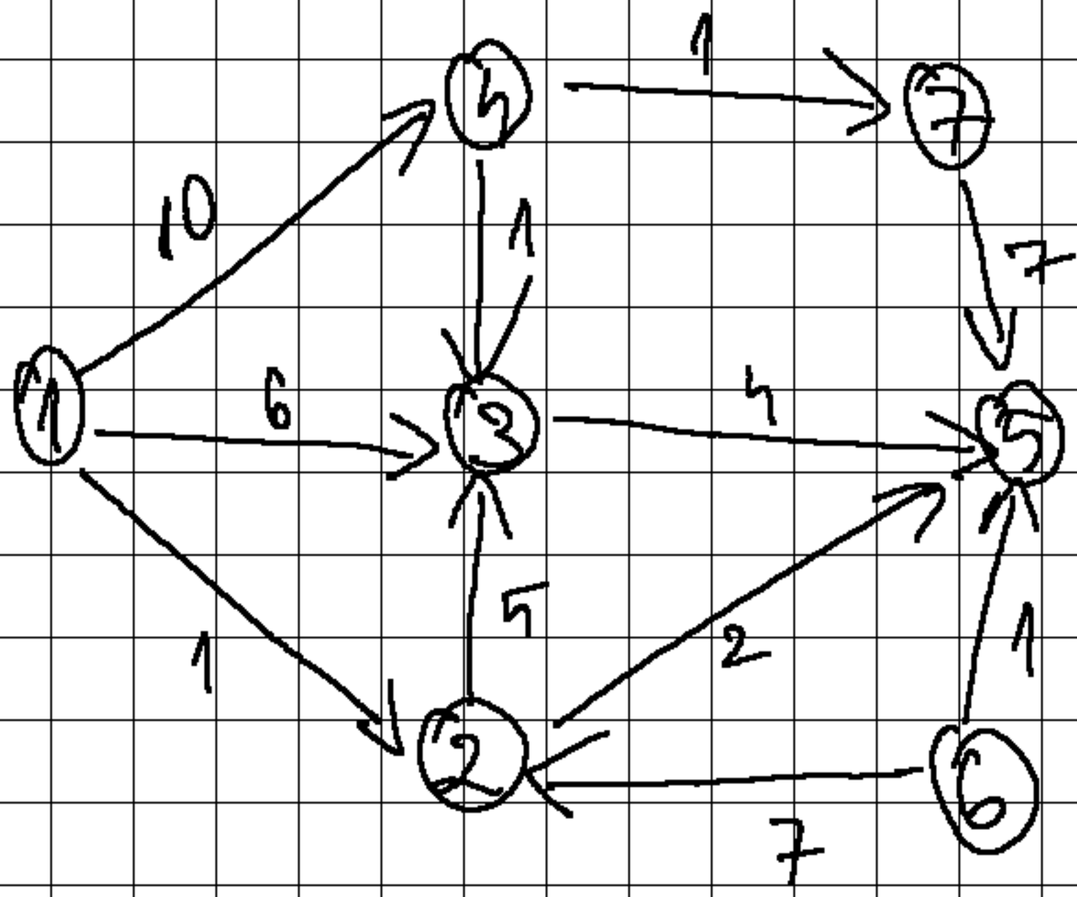


Critical activities (EST = LST):

A, F, D, B, C

Exam 2023 no. 4.

1. in the diagraph below, find the minimum cost path from all vertices to vertex 7, using the Dijkstra algorithm in reverse.



1: 1->4->7 (cost 11)

2: —

3: —

4: 4->7 (cost 1)

5: —

6: —

7: —

x	2	distance							next						
		1	2	3	4	5	6	7	1	2	3	4	5	6	7
	7	∞	∞	∞	∞	∞	∞	0	-	-	-	-	-	-	-
7	4	∞	∞	∞	1	∞	∞	0	-	-	-	7	-	-	-
4	1	11	∞	∞	1	∞	∞	0	4	-	-	7	-	-	-
1		11	∞	∞	1	∞	∞	0	4	-	-	7	-	-	-

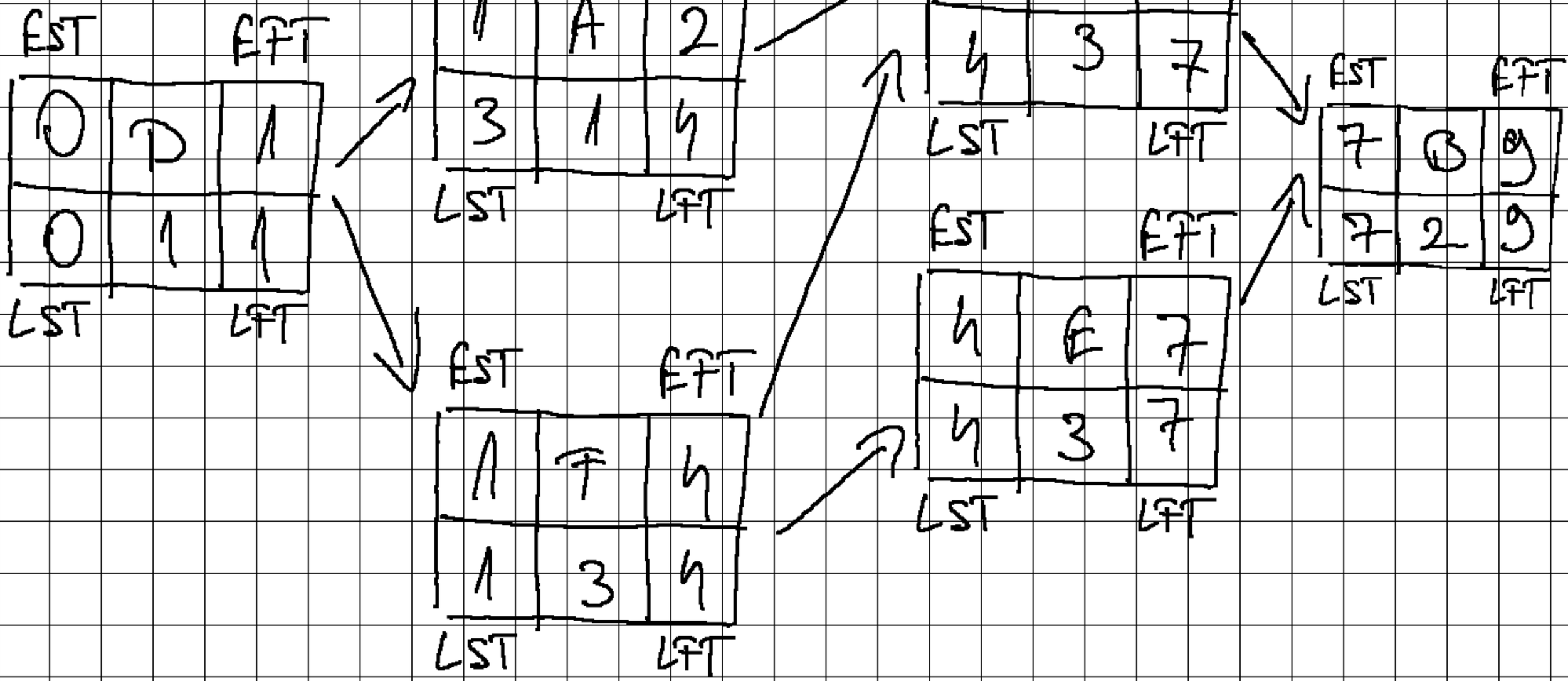
2. Consider the following activities, determine the earliest and the latest scheduling, and the critical activities (show the step by step computations for the topological sorting, then for the starting and ending times)

Act.	Duration	Prerequisites
A	1	D
B	2	C, E
C	3	A, F
D	1	—
E	3	F
F	3	D

sorting is

D, A, F, C, E, B

Step	Remaining nodes	Node processed	in degree A B C D E F	Queue
1	A, B, C, D, E, F	—	1 2 2 0 1 1	D
2	A B, C, E, F	D	0 2 2 – 1 0	A, F
3	B, C, E, F	A	– 2 1 – 1 0	F
4	B, C, E	F	– 2 0 – 0 –	C, E
5	B, E	C	– 1 – – 0 –	E
6	B	E	– 0 – – – –	B
7	1	B	– – – – – –	–



Critical activities (EST = LST):

D , F , C , E , B