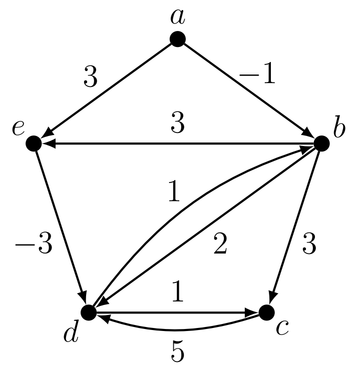
MTH6105 – Algorithmic Graph Theory **Spring 2024**

Problem Sheet 7 **F. Fischer**

You are expected to **attempt all exercises** before the seminar and to **actively participate** in the seminar itself.

---

1. Use the Bellman-Ford algorithm to find a shortest directed a–c-path in the following directed network.



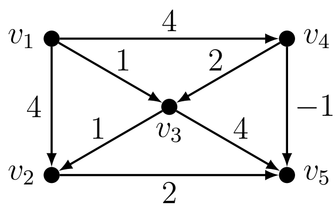
Solution: When started from vertex a, the Bellman-Ford algorithm computes the length $\delta_k(v)$ of a shortest directed a–v-path containing at most k arcs, for $k = 0, 1, \ldots, |V(D)|$ and all $v \in V(D)$. The lengths are shown in the following table, along with the predecessor of v along each shortest path.

	a	b	c	d	e
0	0	∞	∞	∞	∞
1	0	$-1(a)$	∞	∞	$3(a)$
2	0	$-1(a)$	$2(b)$	$0(e)$	$2(b)$
3	0	$-1(a)$	$1(d)$	$-1(e)$	$2(b)$
4	0	$-1(a)$	$0(d)$	$-1(e)$	$2(b)$
5	0	$-1(a)$	$0(d)$	$-1(e)$	$2(b)$

Since the rows for $k = 4$ and $k = 5$ have the same entries, the algorithm has not encountered any negative cycles and thus computed the lengths correctly. The arcs ba, cd, de, and eb connecting each vertex to its predecessor on a shortest directed path form a spanning tree of D, and this spanning tree contains a shortest directed a–v-path for every $v \in V(D)$. In particular, the directed path a, b, e, d, c is a shortest directed a–c-path, and its length is 0.

---

2. Consider the following directed acyclic network.

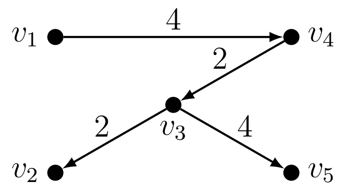


(a) Use Morávek's algorithm to find a longest directed v_1—v_5-path in the network.

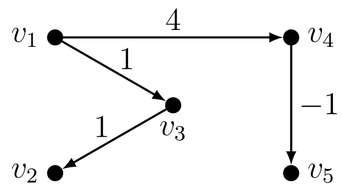
(b) Adapt Morávek's algorithm to find a shortest directed v_1—v_5-path in the network.

Solution:

(a) The algorithm adds vertices in the order v_1, v_4, v_3, v_2, v_5, and edges that maximise the length of the v_1—u-path when vertex u is added. It obtains the following tree. The unique v_1—v_5-path in this tree, the path v_1, v_4, v_3, v_5 is a longest directed v_1—v_5-path in the network.



(b) If edges are added to minimise rather than maximise the length of the v_1—u-path when vertex u is added, we obtain the following tree. The unique v_1—v_5-path in this tree, the path v_1, v_4, v_5 is a shortest directed v_1—v_5-path in the network.



3. Consider the following activities involved in the construction of a garage:

	activity	duration (days)	must follow
a	prepare foundations	7	
b	make door frame	2	
c	lay drains, floor base, and screed	15	
d	install services and fittings	8	e
e	erect walls	10	a, b
f	plaster ceiling	2	d, g
g	erect roof	5	e
h	install door and windows	8	g
i	fit gutters and pipes	2	c, f
j	paint outside	3	i

(a) Draw a directed network representing the activities, their interdependencies, and their durations.

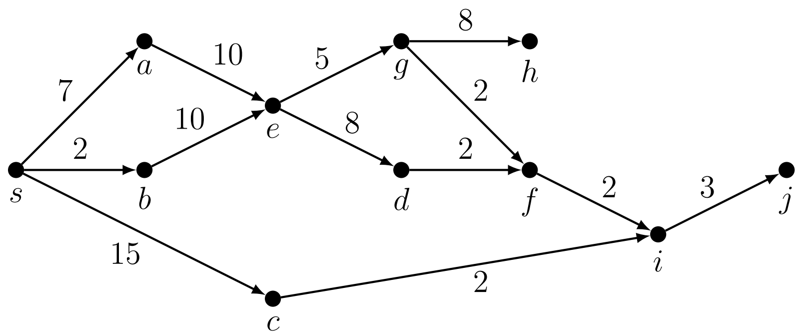
(b) Determine a sequence in which the activities can be executed.

(c) Assume that activities can be executed simultaneously as long as their interdependencies are met. Determine the minimum amount of time required to construct the garage, along with the set of activities that must be completed within their stated duration in order for the minimum time for the construction to be achieved.

Note: It is natural to represent activities by arcs and durations by weights. As some activities have more than one other activity they must follow or be followed by, you may have to introduce additional arcs or vertices to accurately model the interdependencies among activities. There is more than one way in which this can be done.

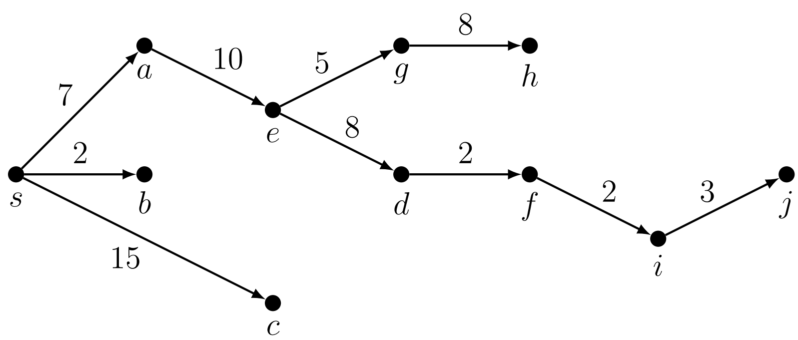
Solution:

(a) For each activity x, we can represent the state of the project where x has been completed by a vertex in the network. It makes sense in this case to also introduce an additional vertex s, corresponding to the start of the project where none of the activities have been completed. Interdependencies and durations can then be represented by an arc xy with weight equal to the duration of y for any pair of activities x and y such that y must follow x. For activities $y \in \{a, b, c\}$, which do not have to follow any other activities, we introduce an arc sy with weight equal to the duration of y. We obtain the following network.



(b) Any topological ordering of the network gives us an order in which the activities can be executed. One such order, which can be found by repeatedly finding a vertex with indegree zero and deleting it from the network, is $s, a, b, c, e, d, g, f, i, j, h$. Note that the network is a directed acyclic network, which is necessary and sufficient for the existence of a topological ordering.

(c) If activities can be executed simultaneously, the minimum overall duration of the project is equal to the length of a longest directed path in the network. As every vertex in the network can be reached along a directed path from vertex s, we can find a longest path by starting Morávek's algorithm from s. The algorithm needs a topological ordering, and we can use the topological ordering we found in the previous part of the question. For this ordering the algorithm constructs the following spanning tree of the network.



A longest path is the unique s–v-path in the spanning tree for a vertex v for which $\delta(v)$ is maximized, which is the s–j-path s, a, e, d, f, i, j. This path has length 32, so the construction of the garage takes at least 32 days.