## Question 1, What is the dual of the linear program

The variables are $Y_S$, $S \in \mathbb{S}$. Objective

$$\min \sum_{S \in \mathbf{S}} Y_s$$

the constraints are :

$$\forall e \in E : \sum_{S \in \mathbf{S} : e \in \delta(S)} Y_s \leq w_e$$

and

$$\forall S \in \mathbb{S} : Y_S \geq 0$$

## Question 2, How many iterations can $Y_S$ be increased?

Since the $e'$ in step 3 belongs $\delta(C)$, one vertice of $e'$ belongs to $C$ and the other vertice $v$ belongs to $V \setminus C$. Thus by adding $e'$ to $F$ the new $C$ contains $v$ as vertice. Thus in each step $C$ has one more vertice and thus all $C$ are different.

Thus a given dual variable can be increased only once.

## Question 3, Algorithm terminates

The algorithm stops because $\mathbb{S}$ is finite and each $Y_S$ is handled only once.

The algorithm terminates because the size of $C$ is increased by one in each step.

The output $P$ is a solution since the algorithm stops if there is a path that conencts $s$ and $t$ which is the case once $t \in C$.

## Question ?, Proof of lemma

The case $i = 1$ holds since a graph with one edge is a tree.

Case $i > 1$, assume the $C$ with $i - 1$ edges is a tree. Proof by contradiction: If $C_{i-1}$ is a tree and $C_i$ is not a tree, it has a cycle the added edge $e'$ in thi steps must be part of the cycle. But this would imply this would contradict the fact the newly added vertice is not in $C_{i-1}$ (Question 2).

## Question 4, Tight lower bound

The shortest Path $P*$ defines the best integral solution to the primal LP, the value $\mathsf{val}(\mathsf{P}*)$ is greater or equal to the value of the optimal fraction solution $\mathsf{val}(\mathsf{Y}*)$ of the dual LP.

## Question 5, Why is y feasible?

The first $y$ is equal to 0 and therefore feasible.

If in the third step of the algorith:

$$e' \text{ such that } \sum_{S \in \mathbf{S}: e' \in \delta(S)} y_S = w(e')$$

then $e'$ will be added $F$ and both its endpoints belong to new $C$ and therefore all further $C$ in the algorithm. Thus $e'$ will not be in some $\delta(C)$ in the rest of the algorithm and therefore in the $y_S$ of its constrained will be increased and so its constrained remains valid.

## Question 6, $\mathsf{val(y)}$ and shortest path

$$\mathsf{val(y)} \leq \mathsf{val(P*)}$$

## Question 7, Relation $w(e)$ and that $\sum_{s \in \mathbf{S}: e \in \delta(S)} y_S$

If $e \in P$ then $e \in F$ (final $F$ of the algorithm). Then $e$ was added because $\sum_{s \in \mathbf{S}: e \in \delta(S)} y_S = w(e)$ which is the sought relation.

## Question 8, $\sum_{e \in P} w(e)$

Summing up both sides of the relation of question 7 gives:

$$\sum_{e \in P} w(e) = \sum_{e \in P} \sum_{s \in \mathbf{S}: e \in \delta(S)} y_S$$

## Question 9, simplify question 8

$$\begin{aligned} \sum_{e \in P} \sum_{s \in \mathbf{S}: e \in \delta(S)} y_S &= \sum_{s \in \mathbf{S}} y_S |\{e \in P : e \in \delta(S)\}| \\ &= \sum_{s \in \mathbf{S}} y_S |P \cap \delta(S)| \end{aligned}$$

## Question 10, Explain Contradiction

If $y_S > 0$ then $S$ is connected. Let $(p_1, q_1)$ and $(p_2, q_2)$ be the first two edges in $P \cap \delta(S)$, the position of the vertice $p_1$ nearer to the start point $s$ than $p_2$. with $p_1, q_2 \in S$ and $q_1, p_2 \notin S$. The last $C$ of the algorithm contains $P$ and is a tree by lemma 1. Since $S$ is conencted there is a path from $p_1$ to $q_2$ with all vertices in $S$. But there is another path from $p_1$ to $q_2$ which contains the vertices $q_1$, $, p_2$ and $(p_2, q_2)$ in $S$. This a contradiction to $C$ being a tree.

## Question 11, Conclusion

$$\sum_{e \in P} w(e) = \sum_{s \in \mathbf{S}} y_S |P \cap \delta(S)| = \sum_{s \in \mathbf{S}} y_S \leq \mathsf{val(y*)} \leq \mathsf{val(P*)}$$

Since also $\sum_{e \in P} w(e) \geq \mathsf{val(P*)}$ it holds that

$$\sum_{e \in P} w(e) = \mathsf{val(P*)}$$

Conclusion: the algorithm computes a path with optimum length.

## Question 12, why is pruning necessary

It is necessary for proving $|P \cap \delta(S)| = 1$ which is used in question 10.

## Question 13, first edge ir vertice added

For the primal dual algorithm:

The first edge $e = (i, j)$ added to $F$ is the first edge such that

$$\sum_{S \in \mathbf{S}, e \in \delta(S)} y_S = w(e)$$

when $y_{\{s\}}$ is increased the first $e$ incident on $s$ such that

$$y_{\{s\}} = w(e)$$

which is the edge $e$ incident on $s$ with minimum weight and the first vertice added is the vertice which is connected to $s$ with a edge of minimum weight.
Dikstra Algorithm:

In the first step $d(j) = \infty$ for $j \neq s$ and thus the $d$ is updated as

$$d(j) := \min(d(s) + w(s, j), d(j)) = \min w(s, j)$$

The edge $e = (s, j)$ which minimizes the $d(j)$ is the edge with minimum weight incident on $s$. Thus the second vertcie added to $D$ is vertice connected to $s$ with the edge of minimal weight.

## Question 14, next vertice added to $C_0$

By the definition of the $a(\cdot)$, the next edge added is $e$ which minimum $a(e)$. Thus the next vertice added is the vertice where in $V \setminus C_0$ which is the other endpoint of $e$. (the first is in $C_0$). This is the vertice with minimum $l(\cdot)$.

## Question 15, edges in $\delta(S)$ and $\delta(S')$

$$\delta(S') \setminus \delta(S) = \{(j, i) \in E : i \notin S\}$$

where $j$ is the last node added.

## Question 16,modify $l(k)$

The node $j$ is added at time $l(j)$ to the set of vertices. For $e \in \delta(S') \setminus \delta(S)$ we have

$$a(e) = l(j) + w(e)$$

the term $l(j)$ comes form the time concept and the term $w(e)$ from satisfying the constraints.

Thus the new $l(k)$ have to be updated to

$$l'(k) := \min(l(k), l(j) + w((j, k))$$

## Question 17, $d(j)$ and $l(k)$

The definition of $d$ amd $l$ is equivalent and thus the edges are added to $G'$ and $D$ in the same order.

## Question 18, complexity

As the primal dual algorithm adds the vertices in the same order as the Dijkstra algorithm the they have the same time complexity. For each added node the look at each edges once. Thus the promal dual algorithm has the worts case complexity of

$$O(|E| + |V| \log |V|$$