Solutions

1. **(a)** The optimal solution is S_1, S_2, S_4, S_5 with value $11+12+14+15 = 52$.

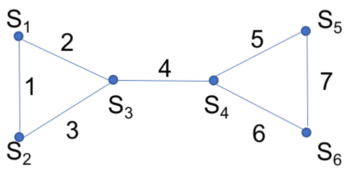
(b)

$$
\begin{aligned}
\min \quad & 11x_1 + 12x_2 + 13x_3 + 14x_4 + 15x_5 + 16x_6 \\
s.t. \quad & x_1 + x_2 \geq 1 \\
& x_1 + x_3 \geq 1 \\
& x_2 + x_3 \geq 1 \\
& x_3 + x_4 \geq 1 \\
& x_3 + x_5 \geq 1 \\
& x_4 + x_5 \geq 1 \\
& x_5 + x_6 \geq 1 \\
& x_1, x_2, x_3, x_4, x_5, x_6 \in \{0, 1\}.
\end{aligned}
$$

(c)

$$
\begin{aligned}
\max \quad & y_1 + y_2 + y_3 + y_4 + y_5 + y_6 + y_7 \\
s.t. \quad & y_1 + x_2 \leq 11 \\
& y_1 + y_3 \leq 12 \\
& y_2 + y_3 + y_4 \leq 13 \\
& y_4 + y_5 + y_6 \leq 14 \\
& y_5 + y_7 \leq 15 \\
& y_6 + y_7 \leq 16 \\
& y_1, y_2, y_3, y_4, y_5, y_6, y_7 \geq 0.
\end{aligned}
$$

(d)



(labeling is not required)

(e) Assume the solution is not feasible. Then, some element i is not covered. But then we can increase the dual value y_i by some small amount and get a better dual solution. However, that is not possible since the dual solution is optimal.

2. We reduce from the Hamiltonian Path (HP) problem. Assume there exists such an α-approximation algorithm with $\alpha < 3/2$. Let $G = (V, E)$ be a graph.

If G has a HP then this is a spanning tree with 2 leaves. The algorithm will return a solution with at most $\alpha \cdot 2 < 3$ leaves. (Hence, the solution has exactly 2 leaves.)

If G has no HP then every spanning tree has at least 3 leaves. The algorithm will return a solution with at least 3 leaves.

Conclusion: Such an algorithm could be used to solve the HP problem in polynomial time. This is not possible assuming P $\neq$ NP.

3. The main idea is to partition in large and small numbers. Then to enumerate over all possible solutions for large numbers and to add the small numbers in a greedy way. For example:

Let $\varepsilon > 0$ be some constant. Say that a number a_i is large if $a_i \geq \varepsilon B$ and say that it is small otherwise.

Algorithm: For all subsets S of at most $1/\varepsilon$ large numbers do the following:
Add small numbers to S until the the total sum is at least B. Store the solution. (If the total sum is less than B then do not store this solution.)
From all solutions found, return the best solution.

Running time:
The number of solutions computed is $n^{O(1/\varepsilon)}$ which is polynomial.

Ratio:
There are at most $1/\varepsilon$ large numbers in an optimal solution. So for one of the stored solutions the set S of large numbers is the same as in the optimal solution. If the optimal solution consists of only large numbers then we are optimal (since the algorithm will not add small numbers). In the other case, the total sum is at most $B + \varepsilon B \leq (1+\varepsilon)\text{OPT}$.

4. (a)

$$\begin{array}{llll} \text{(ILP)} & \min & Z = \sum\limits_{i\in F} f_i y_i + \sum\limits_{i\in F, j\in D} c_{ij} x_{ij} & \\ & s.t. & \sum\limits_{i\in F} x_{ij} = 1 & \text{for all } j \in D, \\ & & x_{ij} \leqslant y_i & \text{for all } i \in F, j \in D, \\ & & x_{ij} \in \{0,1\} & \text{for all } i \in F, j \in D, \\ & & y_i \in \{0,1\} & \text{for all } i \in F. \end{array}$$

(b) $\sum\limits_{i\in F} x_{ij} = 1$ is replaced by $\sum\limits_{i\in S_j} x_{ij} = 1$.

(c) Algorithm:

(1) Solve the LP-relaxation.

(2) For each facility i, open it if $y_i \geq 1/3$.
For each client j, connect j to some $i \in S_j$ for which $x_{ij} \geq 1/3$)
(Alternatively, connect each client j to its nearest open facility in S_j.)

By (b), $x_{ij} \geq 1/3$ for at least one $i \in S_j$. Then, $y_i \geq x_{ij} \geq 1/3$ and i will be opened. So the solution is feasible

The solution is a 3-approximation since all x- and y-values are multiplied by at most a factor 3.

5. (a) It has exponentially many constraints.

(b) A separation oracle detects if all constraints are satisfied and if not, it will return a constraint which is not satisfied.

This LP-has separation oracle. Given a solution x^* we take these x-values as lengths of the edges. Now we compute a shortest path from s to t. If the length of this path is at least 1 then all constraints are satisfied. If it is less than 1 then we found a path for which the constraint is violated.

(c) The probability that an edge (u,v) is in the cut is equal to $|L(u)-L(v)| \leq x^*_{uv}$ (by the triangle inequality). Thus, the expected number of edges in the cut is at most $\sum_{(u,v)\in E} x^*_{uv} = Z^*$.

(d) There are at least two answers possible here. (Answer 2 gives a stronger result than answer 1 but both are fine.)

Answer 1:
Try different values of γ and take the best solution found. Note that there are only $O(n)$ different values of γ to try. To see this note that when γ increases from 0 to 1 then the corresponding solution changes at most n times. We take the best of those solutions.

Answer 2:
Form (c) we know that $\mathbb{E}(|W|) \leqslant Z^*$. Let OPT be the value of a minimum cut. Then, $Z^* \geq OPT$ since it is an LP-relaxation of the real min-cut problem. Hence, $|W| \geq \text{OPT} \geq Z^*$ for every choice of γ. Combining both inequalities we conclude that $|W| = Z^*$ for any choice of γ. So for the derandomized algorithm we can take any $\gamma \in [0,1]$.

6. (a)

$$
\begin{aligned}
\text{(QP)} \quad \max \quad & \tfrac{1}{2} \sum_{(i,j)\in E} (1 - y_i y_j) \\
s.t. \quad & y_i \in \{-1, 1\} \qquad i = 1, \ldots, n.
\end{aligned}
$$

(b)

$$
\begin{aligned}
\text{(VP)} \quad \max \quad & \tfrac{1}{2} \sum_{(i,j)\in E} (1 - v_i \cdot v_j) \\
s.t. \quad & v_i \cdot v_i = 1,\ v_i \in \mathbb{R}^n \quad i = 1, \ldots, n.
\end{aligned}
$$

(c) For example $S = \{1,2,6\}$. The number of edges in the cut is 7. This is maximum since for each of the two triangles we can have at most 2 edges in the cut. So at most $2+2+3 = 7$.
(Other optimal solutions are $S = \{1,3,5\}$, $S = \{2,3,4\}$, $S = \{1,5,6\}$, $S = \{3,4,5\}$, and $S = \{2,4,6\}$.)

(d) $v_1 \cdot v_2 = v_2 \cdot v_3 = v_1 \cdot v_3 = -0.5$. Also $v_4 \cdot v_5 = v_5 \cdot v_6 = v_4 \cdot v_6 = -0.5$.
Further, $v_1 \cdot v_4 = v_2 \cdot v_5 = v_3 \cdot v_6 = -1$.
In total $\frac{1}{2} \sum_{(i,j)\in E} (1 - v_i \cdot v_j) = \frac{1}{2}(6 \cdot 3/2 + 3 \cdot 2) = 7.5$.

(e) The probability that edge $(1,2)$ is in the cut is $2/3$. The same holds for the edges $(1,3), (2,3), (4,5), (5,6), (4,6)$.
The probability that edge $(1,4)$ is in the cut is 1 and the same holds for the edges $(2,5)$ and $(3,6)$.
Thus, the expected number of edges in the cut is $6 \cdot 2/3 + 3 = 7$.