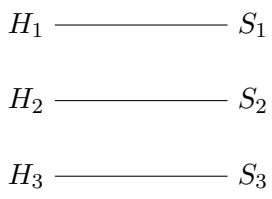
Midterm

1

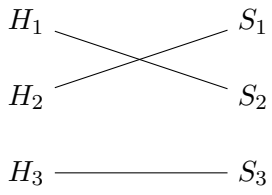
1.a

Below are two stable matchings for the given input:

(H_1, S_1), (H_2, S_2), (H_3, S_3):

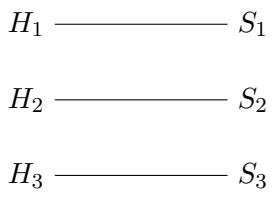


(H_1, S_1), (H_2, S_2), (H_3, S_3):



1.b

The Gale-Shapely algorithm – in which hospitals propose to the students – the output is (H_1, S_1), (H_2, S_2), (H_3, S_3):



2

2.a

Consider the following set of intervals: $(1, 2)$, $(2, 5)$, $(1, 3)$, $(3, 4)$.

The "earliest finish time" algorithm would allocate a first machine for $(1, 2)$, and a second machine for $(1, 3)$. It would send $(3, 4)$ to the first machine, since $(1, 2)$ is now done, and finally it would allocate a third machine for $(2, 5)$, since the first

and second machines are already in use. This interval is partitioned using only two machines using the "earliest start time" algorithm, and so the "earliest finish time" algorithm is not optimal.

2.b

We first observe that we need at least as many machines as we have jobs overlapping at any one point in time so the maximum number of mutually overlapping jobs is a lower bound for the number of machines we must use.

In the greedy algorithm, we check to see if any of the already allocated machines is compatible with each job before we give that job to the machine. The only time we allocate a new machine is when this is not the case – that is, every machine allocated so far is not compatible with the job. This can only happen if there are already jobs running on every allocated machine Since we sorted the jobs by start time, each of the running jobs started before the job we're currently trying to allocate, so we could not have allocated them any differently to use fewer machines.

The number of machines we allocate is therefore equal to the maximum number of mutually overlapping jobs – our lower bound – and thus the algorithm is optimal.

3

Suppose we came up with a way to multiply two 3×3 matrices using exactly 21 multiplications. Our recurrence would look like

$$T(n) = 21T\left(\frac{n}{3}\right) + \Theta\left(n^2\right).$$

By the master theorem, since $2 < \log_3 21$, we have

$$T(n) \in \Theta\left(n^{\log_3 21}\right),$$

and $\log_3 21 < 2.807$, so this is indeed an improvement over Strassen's algorithm.

4

Below is our table after we've filled it out:

		P	A	L	A	T	E
	0	2	4	6	8	10	12
P	2	0	2	4	6	8	10
A	4	2	0	2	4	6	8
L	6	4	2	0	2	4	6
E	8	6	4	2	1	3	5
T	10	8	6	4	3	1	3
T	12	10	8	6	5	3	2
E	14	12	10	8	7	5	3

Our final answer is 3, and it is obtained with the following alignment:
P A L A T – E
P A L E T T E