- Interval Scheduling

\[ n = 5 \]
\[ \begin{array}{cccc}
(1, 3) & (2, 4) & (4, 5) & (8, 10) \\
\times & & & \times \\
(1, 3) & (4, 5) & (8, 10) \\
\end{array} \]

- Proof of correctness.

Assume towards contradiction that the alg is not correct. Then there is an instance with \( n \) meetings, and meeting time \( \{ (s_i, t_i) \}_{i=1,2,\ldots,n} \) where the alg is incorrect. Without loss of generality (wlog) assume \( t_1 \leq t_2 \leq t_3 \leq \ldots \leq t_n \).

For this instance, let ALG schedule meeting

\[ (i_1, i_2, i_3, \ldots, i_k) \]

Let OPT schedule meeting

\[ (j_1, j_2, j_3, \ldots, j_l) \]

If ALG is incorrect, then \( k < l \).

1. There is an index \( 1 \leq p \leq k \) s.t. \( i_p \neq j_p \)

Let \( p \) be the first index where \( i_p \neq j_p \).

By design of the algorithm, \( i_p \) has the earliest ending time among all meetings that can be scheduled together with \( (i_1, i_2, \ldots, i_{p-1}) \)

\[ \Rightarrow t_{i_p} \leq t_{j_p} \]

Consider an alternative solution

\[ (i_1, i_2, \ldots, i_p, j_p+1, j_{p+2}, \ldots, j_l) \]

This solution is also valid because
(i, \ldots, i_p) is valid (by design of ALG)

2. \((i_{p+1}, j_{p+2}, \ldots, j_c)\) is valid (by validity of OPT)

\[ t_{i_p} \leq t_{j_p} \leq s_{j_{p+1}} \]

Validity of OPT

---

- repeating this procedure \( \leq K \) times, can get an alternative optimal solution \((i^{'}, j_2, \ldots, j_c)\) s.t.
  for every \( 1 \leq p \leq K \) \( i_p = j_p \).

2. if for every \( 1 \leq p \leq K \), \( i_p = j_p \).

consider \( j_{K+1} \), by design of the algorithm

\[ t_{j_{K+1}} > t_{j_K} = t_{i_K} \]

alg will consider meeting \( j_{K+1} \) after scheduling \( i_K \).

Since \( j_{K+1} \) does not conflict with \( i_K(j_K) \), alg should have scheduled this meeting. This is a contradiction.

Therefore the algorithm is always correct.