P3

Claim:

The optimal order in which to execute jobs to minimize the average completion time is when the jobs are sorted in ascending order.

Proof:

Let $K = \{k_0, k_1, ..., k_{n-1}, k_n\}$ be a list of jobs sorted in ascending order of processing time. Therefore, the processing time of any job k_i is $k_0 \leq k_i \leq k_n$ at any ith index in K. Let C be a list where each entry $c_i = \sum_i \{k_0, k_1, ...k_i\}$ such that $C = \{k_0, k_0 + k_1, k_0 + k_1 + k_2, ..., \sum K\}$. Note, that the average completion time will be $\frac{\sum C}{|K|}$ and, since the number of entries in K doesn't change, minimizing $\sum C$ is equivalent to minimizing the average completion time. For a contradiction assume that ascending sort order doesn't minimize the average completion time of the jobs. In that case, there must exist a pair of jobs $\{k_m, k_n\}$ in K with processing times $k_m < k_n$ such that swapping $\{k_m, k_n\}$ will result in a lower average completion time. Swap $\{k_m, k_n\}$ in K. Then $C_{swapped} = \{..., k_n + \sum K_{m-1}, ..., k_m + \sum K_{n-1}, ...\}$ where $\sum K_{m-1}$ and $\sum K_{n-1}$ are the sum of $K's$ entries up to m and n, respectively, before swapping. Notice that the value of entry $k_m + \sum K_{n-1}$ in $C_{swapped}$ is equivalent to the entry in $C_{unswapped}$ at that same index, i.e. $k_n + \sum K_{n-1}$, because in $C_{swapped}$ $\sum K_{n-1}$ includes k_n and in $C_{unswapped}$ $\sum K_{n-1}$ includes k_m. Now consider $k_n + \sum K_{m-1}$ in $C_{swapped}$ and $k_m + \sum K_{m-1}$ in $C_{unswapped}$. In $C_{swapped}$ and $C_{unswapped}$ the term $\sum K_{m-1}$ is equivalent, neither includes any terms that were swapped. But, $k_n > k_m$ so $k_n + \sum K_{m-1}$ in $C_{swapped}$ is greater than $k_m + \sum K_{m-1}$ in $C_{unswapped}$. Thus, $\sum C_{swapped}$ is larger than $\sum C_{unswapped}$ which is a contradiction. $\square$