Philippe WS CSC-349

Assignment 4

1.) Triathlon Competitor Scheduling

a. The greedy algorithm will order the contestants by descending of their combined running and biking times.

i. Note: The swimming times of the contestants are not considered by the algorithm, because regardless of their ordering, it will take the same amount of time to complete the swimming portion.

b. Proof By Exchange Argument

a. Assumptions

- Times all begin at 0.
- Assume that the (bike + run) times by the algorithm have been sorted in ascending order of length in the set Σ, in accordance to the greedy algorithm.
- Assume that there is an ordering Σ* that is the optimal solution but is different from the ordering Σ, the ordering given by the algorithm.

b. Notation

- Σ: The ordering given by the greedy algorithm. Ordering by ascending (bike + run) times.
- Σ*: The ordering given by a supposed optimal solution. Different ordering than that of Σ.
- Athletes i and j, the two athletes to be considered
  - s, r, b are their swim, run, and bike times respectively.
- C the total swim time of all previous athletes before the two considered athletes.

c. If Σ is not an optimal solution, then Σ* must contain at least two athletes (i & j), which are not ordered by ascending deadlines, and are out of order compared to Σ. Let these be the two athletes we consider.

d. Swap

Athletes Before i	Athlete j	Athlete i	Athletes after j

Since athlete i's (bike + run) time is greater than or equal to that of athlete j's,

$(b_i + r_i) - (b_j + r_j) \geq 0$ must be true.

If we compare the total triathlon times between athletes i and j, before and after the swap.

Triathlon Time (i & j)	Athlete i	Athlete j
Before	$(C + s_j + s_i + r_i + b_i)$	$(C + s_j + r_j + b_j)$
After	$(C + s_i + r_i + b_i)$	$(C + s_i + s_j + r_j + b_j)$

Compare Before and After: We should only compare the greatest triathlon completion times before and after the swap, because any of the lesser times values will not contribute to the maximum.

The athletes with the greatest times before and after the swap have completion times of $(C + s_i + s_j + r_i + b_j)^{Before}$ and $(C + s_j + s_i + r_i + b_i)^{After}$. If we compare local maximum triathlon completion times before and after the swap by subtracting them: $(C + s_j + s_i + r_i + b_i) – (C + s_i + s_j + r_j + b_j)$, we end up with

$(r_i + b_i) – (r_j + b_j)$, which we know previously to be greater than or equal to 0. This means that for this swap, we have either reduced the local greatest triathlon time or not changed it.

e. Thus, by continuously swapping every pair that is relatively mis ordered in Σ*, we will continue to reduce or not change the local greatest triathlon time between those two swapped athletes.

   After enough swaps, we will have reduced or not changed the local maxima between every mis ordered pair, and therefore either reduced the overall completion time of the triathlon, or not changed it all, since every swap cannot make the time worse.

   The result is that Σ* will become Σ. This means that the greedy solutions triathlon completion time is either the same or better than the supposed optimal solution.

f. Hence, the optimal solution is the greedy solution.

c. The algorithm's complexity is θ($n * \log(n)$) since the list must be sorted, and any efficient sorting algorithms θ($n * \log(n)$).

2.) Shipping Efficiency

Proof By Induction

Assumptions: Assume $\Sigma^* = \{ T_1^*, \ldots , T_n^* \}$ is an optimal solution to the shipping efficiency problem, where T_k is the total number of packages shipped by all trucks k. Let $\Sigma = \{ T_1, \ldots , T_m \}$ be the total number of packages shipped by the proposed greedy solution algorithm for the same problem set.

Proposition: **Let P(k) be the proposition $T_k^* \leq T_k$ to ship all to T packages, and every previous package up to each respective interval has been shipped on a truck, with no trucks weight exceeding W**: Want to show this is true for all $1 \leq k \leq n$. This will also be used to prove Σ is optimal, having the same number of utilized trucks.

Proof By Induction: Let P(k) be the proposition $T_k^* \leq T_k$ and that all packages thus for have shipped or loaded. Want to show this is true $1 \leq k \leq m$.

Base Case: Show P(1) is true, i.e. $T_1^* \leq T_1$.

Proof of Base Case: $T_1^* \leq T_1$ is true because for the first set of packages, by definition of the algorithm, the first truck will have maximally loaded packages up to its limit W. The optimal solution must also do so, lest it leave packages unshipped and require additional trucks to ship said packages.

Inductive Case: Show P(k) is true $\rightarrow$ P(k+1) is true for any $1 \leq k \leq m$. Thus, we need to show that $T_k^* \leq T_k \rightarrow T_{k+1}^* \leq T_{k+1}$:

Proof of Inductive case: T_{k+1} is the total number of packages shipped by all previous trucks, including the number of packages it will ship.

Claim: The total number of packages T_{k+1}^* must be less than or equal to that T_{k+1}.

a. Suppose the claim were not true.
b. Thus, $T_{k+1} < T_{k+1}^*$
c. This cannot be true, because by definition of the algorithm, T_{k+1} is loaded every truck with the maximum number of packages who weights do not exceed W, and if T_{k+1}^* were to attempt to exceed this, it would require an additional truck, and thus not be an optimal solution.
d. Thus $T_{k+1}^* \leq T_{k+1}$ holds true.

Thus by Principle of Mathematical Induction: P(k) is true for k = 1 … m.

Conclusion: Show m = n. Since the proposition above was proven true, the greedy algorithm is shown to ship as many packages or more than the optimal solution. If the optimal solution were to place too few packages in any truck, then it is possible it would require an additional truck to ship the missed packages, thus not be an optimal solution. Thus m = n must be true.

Time Complexity

The complexity is $\theta(n)$ since the every package must be compared total package weight of the truck W, to ensure it does not exceed the capacity.