Homework 0

Zi Yan

September 9, 2011

Problem 1 Solution:

Fact 1. *A ship should be the last one that schedules to visit a port that it will remain in.*

Proof. By contradiction, we suppose ship S, which is not the last ship that will visit port P, will remain in port P, beginning on day d, and another ship S' has a schedule to visit port P on day $d + x$, where $0 \leq x \leq (m - d)$. But this will conflict with the requirement (†). Therefore S cannot remain in the port P for the rest of the month, which contradicts with the assumption. □

In order to proof that a set of truncations can always be found, we can use contradiction.

Proof. Suppose on some day d, a ship S cannot remain in any port, while all the other $n - 1$ ships all remain in their ports respectively.

There are three cases here,

1. all ports are occupied, so S has no place to remain in.
2. S remains in port P, but there is another ship S' that will schedule to visit port P on day $d + x$ $(0 \leq x \leq (m - d))$, in order not to violate the requirement (†), S cannot remain in P.
3. a port P is the only empty port, but S stayed in P before, so S cannot visit and remain in P.

For first case, because there are n ships and n ports, and the rest $n - 1$ ships already remain in the ports, there must be exactly one empty port.

For second case, from the assumption, we know all the other ships are in ports, so there cannot be another ship that will visit port P.

For last case, it means another ship S', instead of S, is the last ship that visit the port P. From Fact 1, we know S' will remain in P, so P is not empty. There is a contradiction.

In sum, the ship S will find a port to remain in, while all the other ships are in their ports already. In other words, there is a set of truncations can always be found. □

Algorithm 1 An algorithm for finding a set of truncation

Require: Initially all n ships $s \in S$ and all n ports $p \in P$ are all free.
Ensure: At last, all ships remain in different ports respectively after m days.

```
for $i = 1$ to $m$ do
    for $j = 1$ to $n$ do
        if $s_j$ has a schedule to visit port $p$ on day $i$ and $s_j \in S$ then
            let $s_j$ remain in $p$
            remove $s_j$ from $S$
            if there is already another ship $s'$ in the port then
                move $s'$ out
                put $s'$ back into $S$
            end if
        end if
    end for
end for
```