CMPS 102 — Fall 2018 — Homework 3

*"I have read and agree to the collaboration policy." - **Kevin Wang***

Solution to Problem 2: Space-Efficient MST

Let connected graph $G = (V, E)$, where $|V| = n$ and $|E| = m$, have a weight function $w : E \to \mathbb{R}$. Vertex s is the source.

Algorithm 1 Finds the MWST using eager implementation of Prim's

```
EAGER-PRIMS $(G, s)$:
Initialize array edgeTo[ ] // Stores the shortest edge from a MST vertex
Initialize array distTo[ ] // Stores weight of edgeTo[ ]
Initialize array onTree[ ] // Stores 'true' if vertex is on MST
Initialize indexable priority queue Q
for $i = 1$ to $n$ do
    distTo$[i] = \infty$
end for
distTo$[0] = 0.0$
$Q = (s$, 0.0) // Initialize indexable queue with key: source weight: 0.0
while $|Q| \neq 0$ do
    $x$ =EXTRACT-MIN($Q$) // Get closest vertex
    onTree$[x]$ = true
    for each $y \in$ adj$[x]$ do
        if onTree$[y]$ then
            Continue
        else
            if distTo$\{y\}$ >weight($x$—$y$) then
                edgeTo$[y]$ = $x$—$y$
                distTo$[y]$ = weight($x$—$y$)
                DECREASE-KEY($y$, distTo$[y]$)
            end if
        end if
    end for
end while
```

The algorithm uses arrays and a priority queue indexed and keyed by vertex thus each array is at most size n. Thus the space complexity is $4 \times O(n) = O(n)$.