Algorithm 1 Groups datapoints into k clusters. Initalizes centers as random selection of k datapoints within the dataset, assigns data into cluster with nearest center. Recomputes cluster centers with current group membership. Reassigns data to clusters based on nearest center. Repeats last two steps until no changes are made between iterations.

Input: A matrix of datapoints with rows being observations and columns being dimensions, $A \in \mathbb{R}^{n \times d}$. Number of clusters to group data into, $k \in [1, d]$.

Output: Vector of cluster assignments for each observation, $v \in \mathbb{C}^n$.

```
function KMEANS($A$,$k$)
    Initialize $v = \text{ZEROS}(n, 1)$                          ▷ create empty cluster vector assignment
    Initialize $v' = \text{ONES}(n, 1)$      ▷ create empty cluster vector assignment that is different
    Get $r = \text{INT}(\text{RAND}(k) \times n)$                ▷ Pick random $k$ points to set as clusters centers
    Set $c = A(r, :)$                                            ▷ Get initial cluster centers
    while $v'$ is not $v$ do                     ▷ while last two cluster assignments aren't the same
        $v = v'$                                                 ▷ save last cluster assignment
        for $i = 1, \ldots, n$ do                                ▷ for each observation
            Set $d = \text{ZEROS}(k, 1)$                         ▷ set distances from point to centers as zero
            for $j = 1, \ldots, k$ do                            ▷ for each cluster
                $d(j) = \text{DISTANCE}(A(i, :), c(j, :))$       ▷ compute distance
            end for
            $v'(j) = \text{WHERE}(\text{MIN}(d))$                ▷ assign current observation to nearest cluster
        end for
        for $j = 1, \ldots, k$ do
            $c(j, :) = \text{MEAN}(A(\text{WHERE}(v' == j), :))$  ▷ update centers with mean of cluster
        end for
    end while
end function
```