
FURTHEST-IN-FUTURE(R, C)

```
1  for  $i = 1$  to  $R.length$ 
2      if  $R[i] \in C$ 
3          Cache Hit
4      else
5          Cache Miss
6          if Cache is not full
7              add  $R[i]$  to Cache
8      else
9          //  $tmp$  is used to keep track of the latest
10         // appearance of cache item in the sequence
11          $tmp = i$ 
12         //  $pos$  is used to keep track of
13         // the position in cache to be replaced
14          $pos = 1$ 
15         for  $j = 1$  to  $C.length$ 
16              $p = i$ 
17             // find the first appearance of the specified
18             // cache item in the sequence
19             while _____
20                  $p = p + 1$ 
21             // if  $p$  reaches the end of sequence without
22             // finding the specified cache item
23             if _____
24                 //  $p = Infinity$ 
25                  $pos = j$ 
26                 break from the inner for loop
27             // update  $tmp$ 
28             if _____
29                 _____
30                 _____
31          $C[pos] = R[i]$ 
```

See slides 48 - 51 for more detailed analysis of optimality at: <http://www.cs.princeton.edu/~wayne/kleinberg-tardos/pdf/04GreedyAlgorithmsI.pdf>