

Algorithms: Homework 2

Li-Yuan Wei

May 3, 2023

Problem 1

Solution.

$0(\text{initial})$	<i>digit 1</i>	<i>digit 2</i>	<i>digit 3</i>
NOD	EBB	BAN	ASH
HOG	NOD	BAR	BAN
SHY	TED	PAR	BAR
BAN	HOG	EBB	EBB
BAR	FIG	TED	FIG
JET	ASH	JET	HOG
EBB	BAN	PET	JET
PAR	BAR	SHY	NOD
ASH	PAR	FIG	PAR
PET	JET	NOD	PET
TED	PET	HOG	ROT
ROT	ROT	ROT	SHY
FIG	SHY	ASH	TED

■

Problem 2

Solution.

■

Problem 3

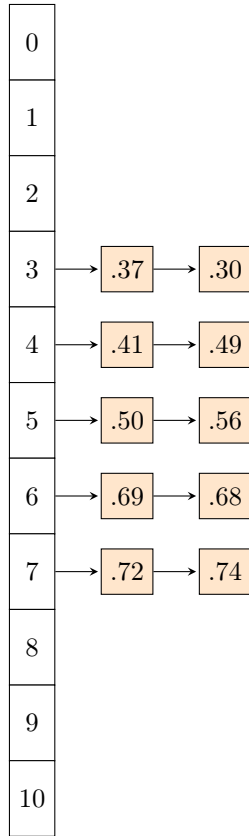


Figure 1: Insert data

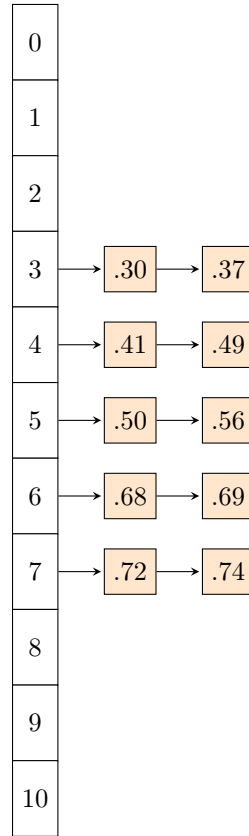


Figure 2: Sort each bucket

Solution. To get sorted data, chain all in Figure 2 from bucket 0 to bucket 10:
 $\langle .30, .37, .41, .49, .50, .56, .68, .69, .72, .74 \rangle$

Problem 4

Solution. Psuedocode for finding the k -th smallest element:

FIND-K-TH-SMALLEST(X, k)

```

1:  if  $i > A.heap-size$ 
2:    error  $A[i]$  does not exist
3:   $key = A[i]$ 
4:   $A[i] = A[A.heap-size]$ 
5:   $A.heap-size = A.heap-size - 1$ 
6:  if  $A[i] < key$ 
7:    MAX-HEAPIFY( $A, i$ )
8:  else
9:     $key1 = A[i]$ 
10:    $A[i] = key$ 
11:   HEAP-INCREASE-KEY( $A, i, key1$ )

```

Problem 5

Solution. Psuedocode for finding the median of arrays X and Y:

FIND-MEDIAN(X, Y, s, n)

```
1:  if  $i > A.heap-size$ 
2:    error  $A[i]$  does not exist
3:   $key = A[i]$ 
4:   $A[i] = A[A.heap-size]$ 
5:   $A.heap-size = A.heap-size - 1$ 
6:  if  $A[i] < key$ 
7:    MAX-HEAPIFY( $A, i$ )
8:  else
9:     $key1 = A[i]$ 
10:    $A[i] = key$ 
11:   HEAP-INCREASE-KEY( $A, i, key1$ )
```

■

Problem 6

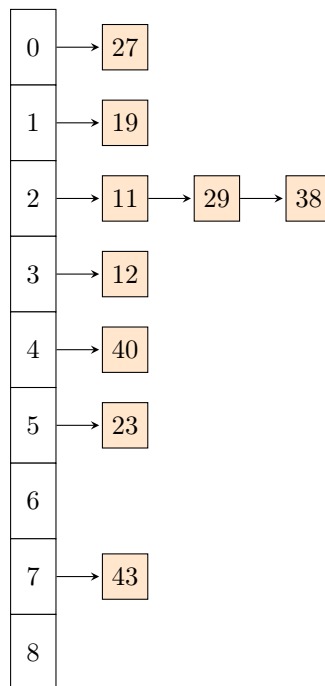


Figure 3: Hash table with chaining

Problem 7

0	87
1	23
2	12
3	58
4	4
5	0
6	28
7	17
8	0
9	31
10	10

Figure 4: linear probing

0	58
1	23
2	87
3	17
4	4
5	12
6	28
7	7
8	8
9	31
10	10

Figure 5: quadratic probing

0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0

Figure 6: double hashing

Problem 8

Solution.



Problem 9

Solution.



Problem 10

Solution.

