Review for Final Exam in Data Structures

Michael T. Gastner (30 March 2023)

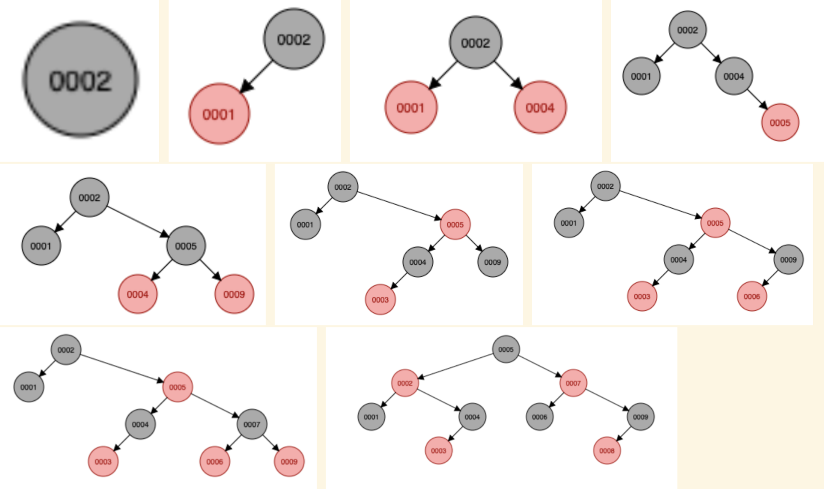
Red-Black Trees: Inserting Nodes

Exercise

Show the red-black trees that result after successively inserting the keys 2, 1, 4, 5, 9, 3, 6, 7 and 8 into an initially empty red-black tree.

Red-Black Trees: Inserting Nodes

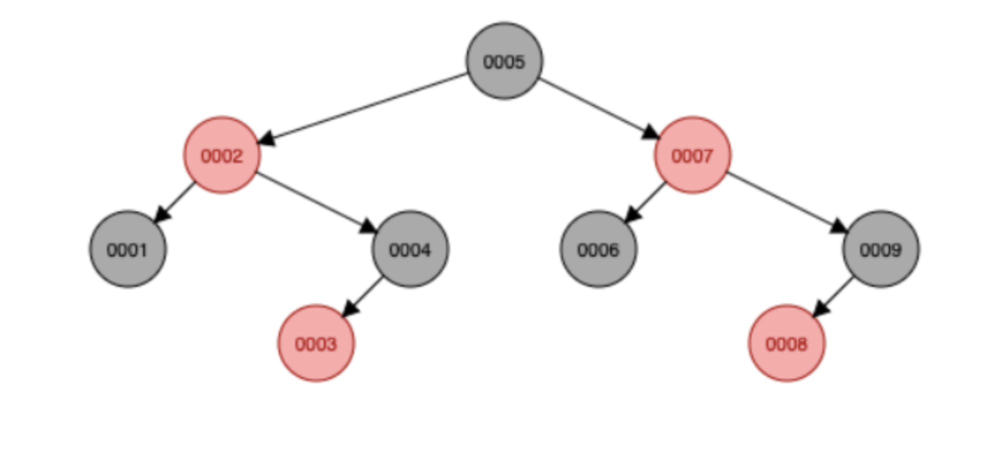
Solutions



Red-Black Trees: Deleting Nodes

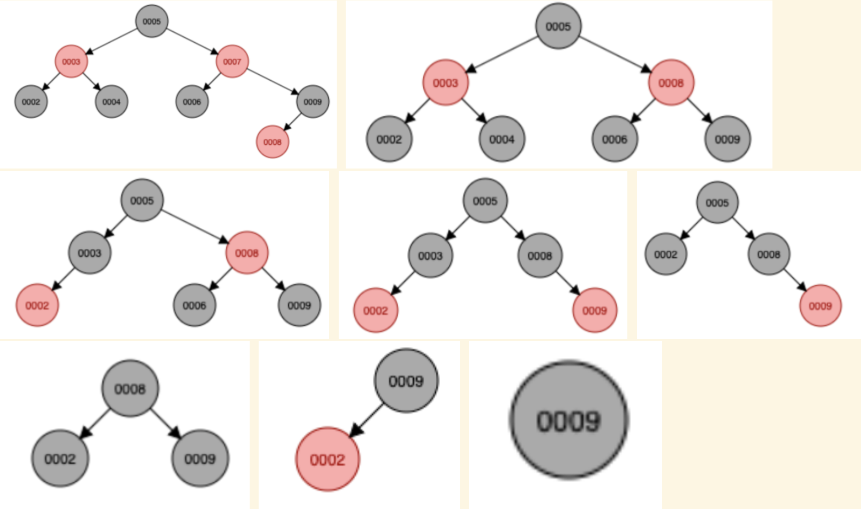
Exercise

Successively delete the nodes 1, 7, 4, 6, 3, 5, 8 and 9.



Red-Black Trees: Deleting Nodes

Solution



Heapsort

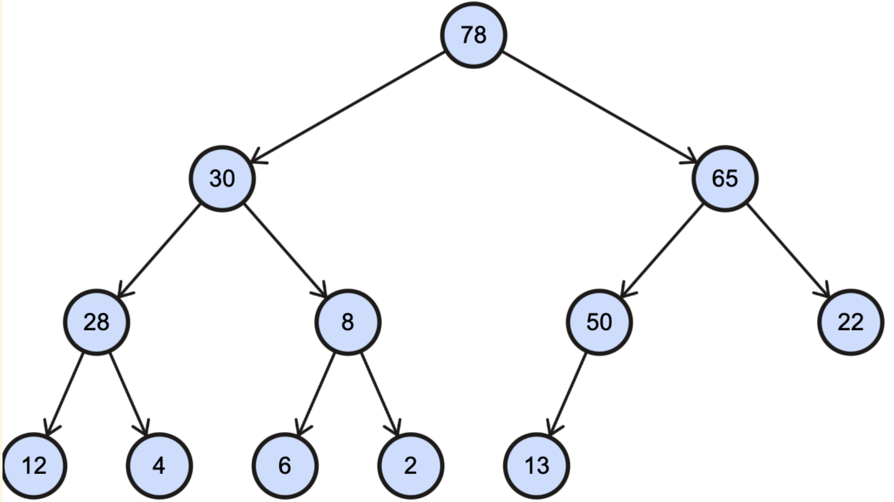
Exercise

(a) Build a max-heap from the array $A = [12, 78, 50, 30, 8, 65, 22, 28, 4, 6, 2, 13]$. Present the end result as a tree.

(b) Apply heapsort to the result of part (a). Show your working.

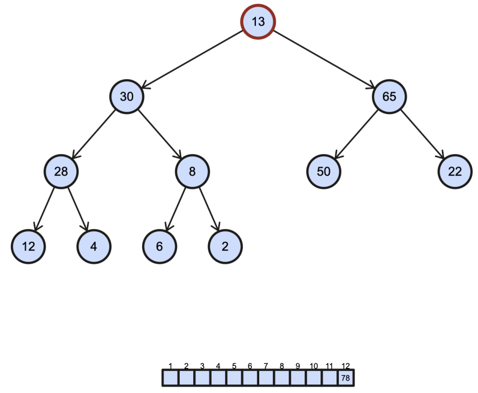
Heapsort

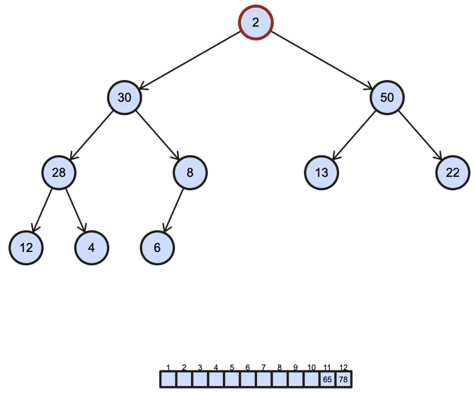
Solution to (a)



Heapsort

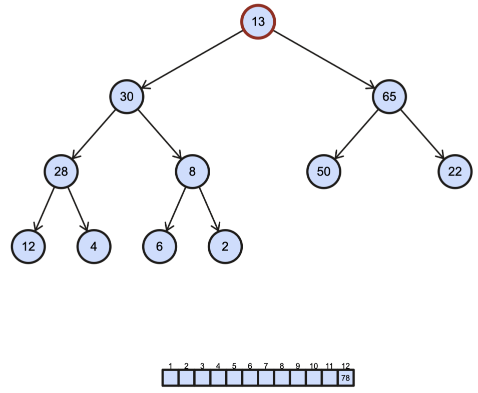
Solution to (b)

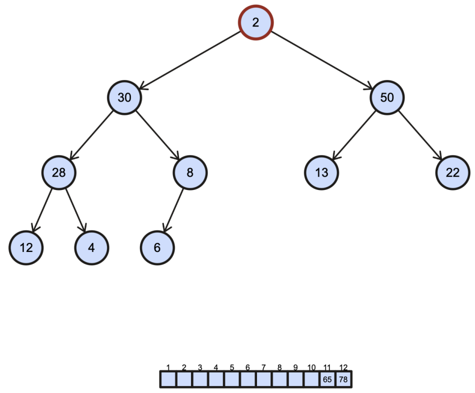




Heapsort

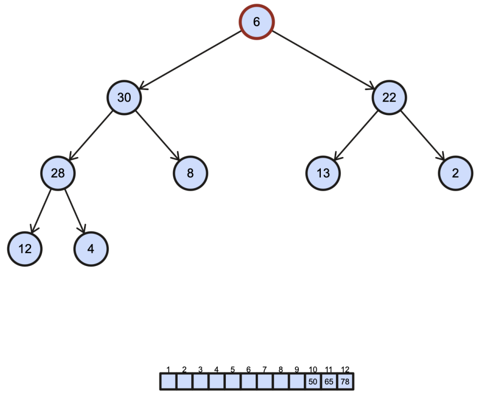
Solution to (b)



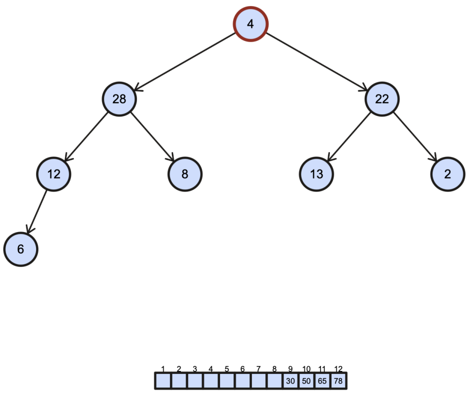


Heapsort

Solution to (b)



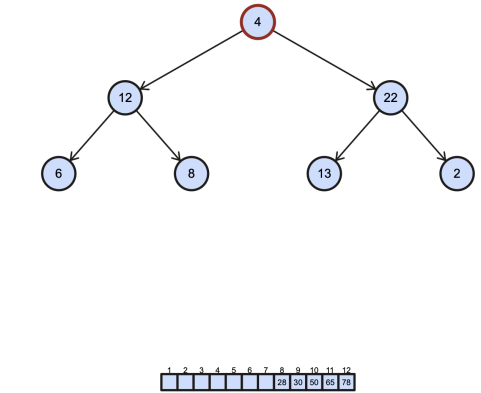
1	2	3	4	5	6	7	8	9	10	11	12
									50	65	78



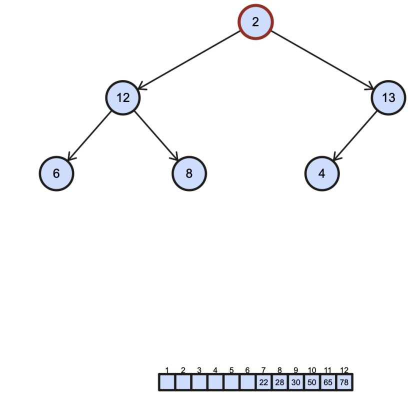
1	2	3	4	5	6	7	8	9	10	11	12
								30	50	65	78

Heapsort

Solution to (b)



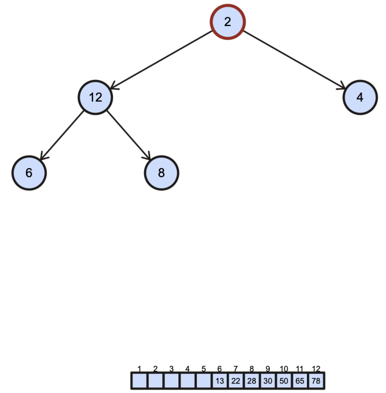
1	2	3	4	5	6	7	8	9	10	11	12
							28	30	50	65	78



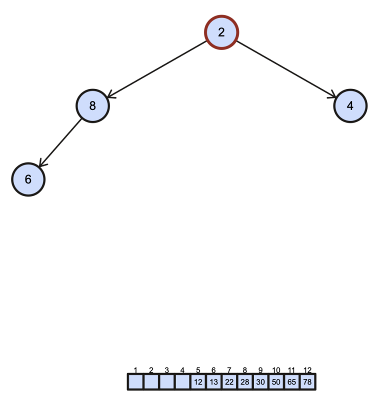
1	2	3	4	5	6	7	8	9	10	11	12
						22	28	30	50	65	78

Heapsort

Solution to (b)



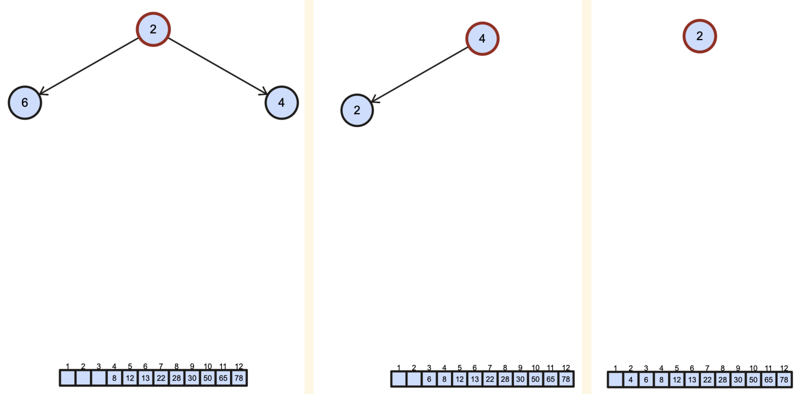
1	2	3	4	5	6	7	8	9	10	11	12
					13	22	28	30	50	65	78



1	2	3	4	5	6	7	8	9	10	11	12
				12	13	22	28	30	50	65	78

Heapsort

Solution to (b)



1	2	3	4	5	6	7	8	9	10	11	12
			8	12	13	22	28	30	50	65	78

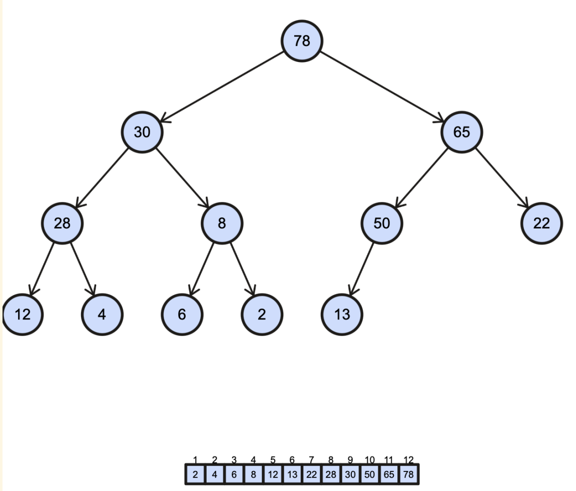
1	2	3	4	5	6	7	8	9	10	11	12
		6	8	12	13	22	28	30	50	65	78

1	2	3	4	5	6	7	8	9	10	11	12
	4	6	8	12	13	22	28	30	50	65	78

Heapsort

Solution to (b)

Summary



1	2	3	4	5	6	7	8	9	10	11	12
2	4	6	8	12	13	22	28	30	50	65	78

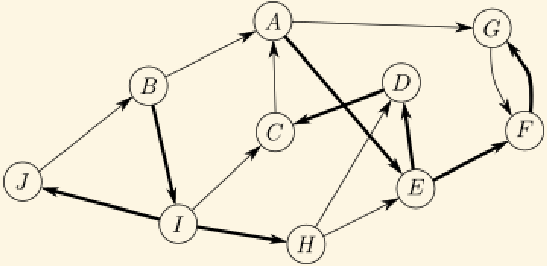
Strongly Connected Components

Exercise (a)

(a) Perform line 1 of $\textsc{Strongly-Connected-Component}(G)$ on the graph G shown below. Insert the discovery and finishing times in the table below.

Assume that the loop of lines 5–7 of DFS considers vertices in alphabetical order and that the adjacency lists are in alphabetical order.

Vertex v	A	B	C	. . .
v.d				. . .
v.f				

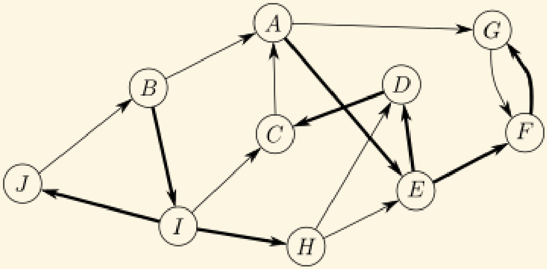


Strongly Connected Components

Exercise (b)

(b) Perform lines 2 and 3 of Strongly-Connected-Component(G) on the same graph G as in part (a). Insert the predecessors (also known as parents) of the vertices in G^{T} in the table below.

Assume that the loop of lines 5–7 of DFS considers vertices in alphabetical order and that the adjacency lists are in alphabetical order.



Vertex v	A	B	C	...
$\mathbf{v}.\pi$				...

Strongly Connected Components

Exercise (c)

(c) Perform line 4 of $\textsc{Strongly-Connected-Component}(G)$. What are the strongly connected components of G?

Strongly Connected Components

Solution to (a) and (b)

(a)

Vertex v	A	B	C	D	E	F	G	H	I	J
v.d	1	13	4	3	2	7	8	15	14	17
v.f	12	20	5	6	11	10	9	16	19	18

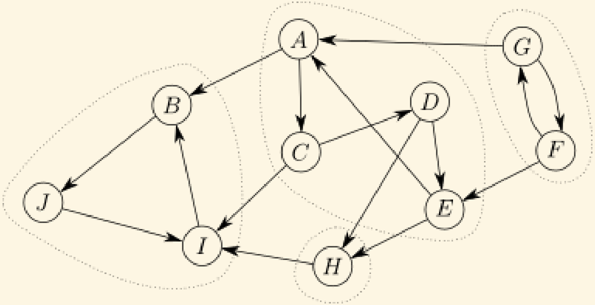
(b)

Vertex v	A	B	C	D	E	F	G	H	I	J
v.π	NIL	NIL	A	C	D	NIL	F	NIL	J	B

Strongly Connected Components

Solution to (c)

(c) The strongly connected components are $\{A, C, D, E\}$, $\{B, I, J\}$, $\{F, G\}$ and $\{H\}$.



Hash Tables

Exercise

We are given an initially empty hash table of length 11. Insert the keys $0, 8, 9, 1, 52, 44, 56, 53, 61, 64$ (in this order) using

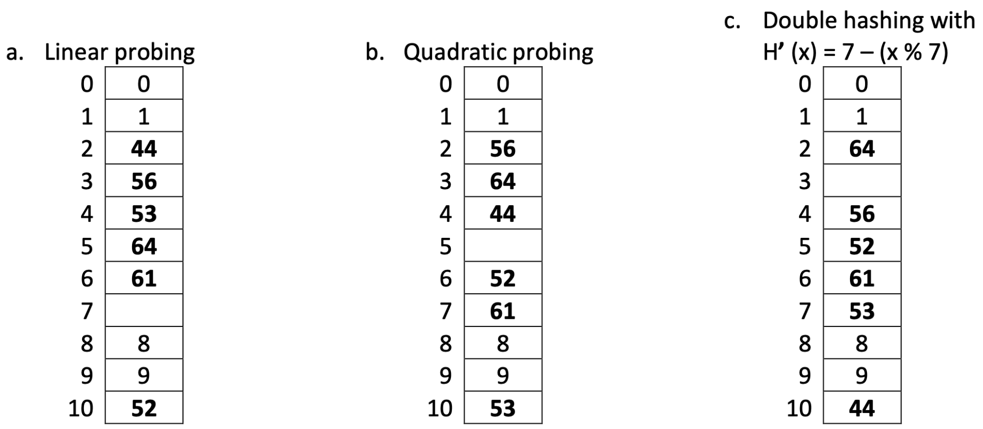
(a) linear hashing based on the auxiliary hash function $h'(k) = k \bmod 11$.

(b) quadratic hashing based on the hash function $h(k, i) = (h'(k) + i^2) \bmod 11$.

(c) double hashing based on the hash function $h(k, i) = (h_1(k) + ih_2(k)) \bmod 11$, where $h_1(k) = h'(k)$ and $h_2(k) = 7 - (k \bmod 7)$.

Hash Tables

Solution



a. Linear probing

0	0
1	1
2	**44**
3	**56**
4	**53**
5	**64**
6	**61**
7	
8	8
9	9
10	**52**

b. Quadratic probing

0	0
1	1
2	**56**
3	**64**
4	**44**
5	
6	**52**
7	**61**
8	8
9	9
10	**53**

c. Double hashing with H’ (x) = 7 – (x % 7)

0	0
1	1
2	**64**
3	
4	**56**
5	**52**
6	**61**
7	**53**
8	8
9	9
10	**44**

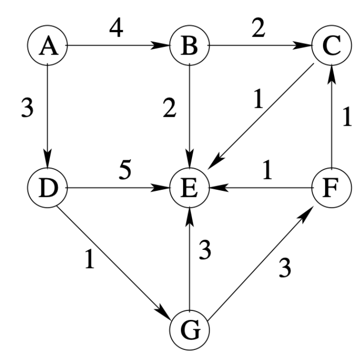
Weighted Shortest Paths

Exercise

(a) Execute Dijkstra's algorithm on the graph shown on the right, starting at vertex A. If there are any ties, the vertex with the lower letter comes first.

Insert your results in the table below.

Vertex v	A	B	C	D	E	F	G
$\mathbf{v.d}$							
$\mathbf{v.}\pi$							



(b) Based on your results in (a), what is a shortest path from A to E?

Weighted Shortest Paths

Solution

(a)

Vertex v	A	B	C	D	E	F	G
v.d	0	4	6	3	6	7	3
$\mathbf{v}.\pi$	NIL	A	B	A	B	G	D

(b) The shortest path from A to E is (A, B, E).

Disjoint-Set Forest

Exercise

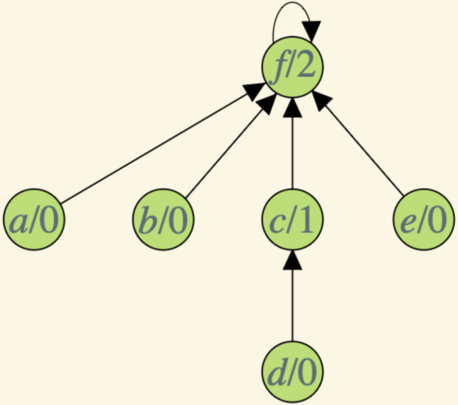
Draw the disjoint-set forest at the end of the following program. Include the ranks in your drawing.

```
Make-Set($a$)
Make-Set($b$)
Make-Set($c$)
Make-Set($d$)
Make-Set($e$)
Make-Set($f$)
Union($b, c$)
Union($b, d$)
Union($e, f$)
Union($b, e$)
Union($b, a$)
```

Disjoint-Set Forest

Solution

Ranks are represented by the numbers inside the nodes.


f/2
a/0
b/0
c/1
e/0
d/0