

CÁCH TRÌNH BÀY TỰ LUẬN CÁC THUẬT TOÁN

MÔN: CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

Quang D. C.

dcquang@it.tdt.edu.vn

1) Sắp xếp

Selection Sort, Bubble Sort, Insertion Sort trình bày theo từng pass (kết quả khi kết thúc mỗi vòng lặp ngoài)

VD: Sắp xếp tăng dần dãy số {6, 9, 5, 8, 1} bằng thuật toán Insertion Sort

pass	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
-	6	9	5	8	1	3
1	6	9	5	8	1	3
2	5	6	9	8	1	3
3	5	6	8	9	1	3
4	1	5	6	8	9	3
5	1	3	5	6	8	9

Ghi chú:

- Pass “-” là dòng ghi lại dãy số ban đầu.
- Mỗi pass là kết quả của một vòng lặp ngoài.

Các thuật toán **Merge Sort** và **Quick Sort** trình bày theo slide lý thuyết.

2) Hashing

Trình bày từng bước tính toán và bảng băm kết quả.

VD: Xây dựng bảng băm từ dãy số sau thông qua hàm băm **hash(k) = k % m** với **m = 11**. Giải quyết độ bằng Quadratic Probing

13	11	20	16	27	38	22	47	8
----	----	----	----	----	----	----	----	---

$$13 \% 11 = 2$$

$$11 \% 11 = 0$$

$$20 \% 11 = 9$$

$$16 \% 11 = 5$$

$$27 \% 11 = 5 \rightarrow \text{Đụng độ/Collision}$$

$$((27 \% 11) + 1) \% 11 = 6$$

$$38 \% 11 = 5 \rightarrow \text{Đụng độ/Collision}$$

$$((38 \% 11) + 1) \% 11 = 6 \rightarrow \text{Đụng độ/Collision}$$

$$((38 \% 11) + 4) \% 11 = 9 \rightarrow \text{Đụng độ/Collision}$$

$$((38 \% 11) + 9) \% 11 = 3$$

$$22 \% 11 = 0 \rightarrow \text{Đụng độ/Collision}$$

$$((22 \% 11) + 1) \% 11 = 1$$

$$47 \% 11 = 3 \rightarrow \text{Đụng độ/Collision}$$

$$((47 \% 11) + 1) \% 11 = 4$$

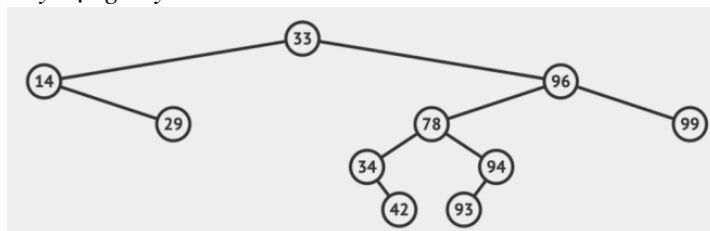
$$8 \% 11 = 8$$

0	11
1	22
2	13
3	38
4	47
5	16
6	27
7	
8	8
9	20
10	

3) Binary Search Tree

VD: Cho dãy số [33, 96, 14, 29, 78, 94, 34, 42, 93, 99]

a. Xây dựng cây/thêm nút



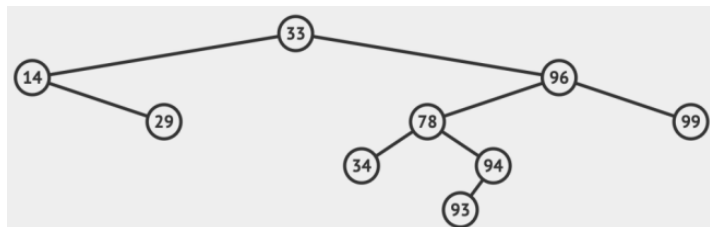
b. Duyệt cây NLR, LNR, LRN

NLR: 33, 14, 29, 96, 78, 34, 42, 94, 93, 99

LNR: 14, 29, 33, 34, 42, 78, 93, 94, 96, 99

LRN: 29, 14, 42, 34, 93, 94, 78, 99, 96, 33

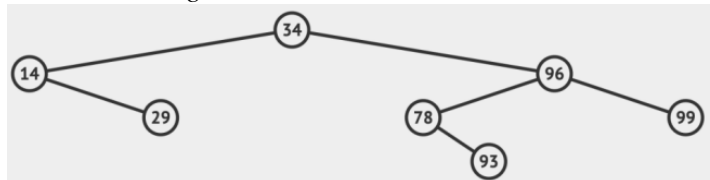
c. Xóa nút 42



d. Xóa nút 94



e. Xóa nút 33 dùng successor

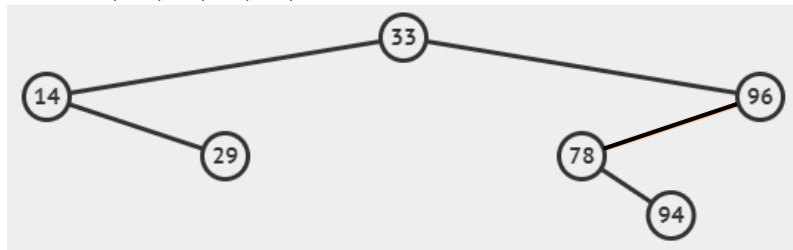


4) AVL

VD: Cho dãy số [33, 96, 14, 29, 78, 94, 34, 42, 93, 99]

a. Xây dựng cây/thêm nút và xoay cây

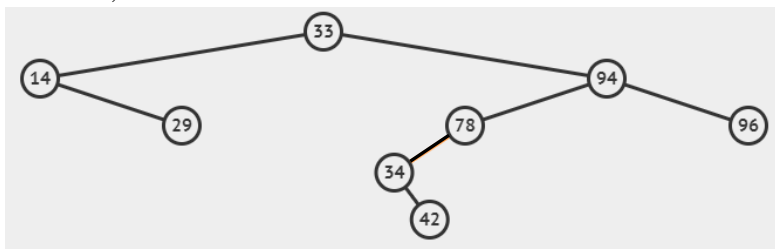
Thêm 33, 96, 14, 29, 78, 94



Xoay trái-phải / Left-right rotation



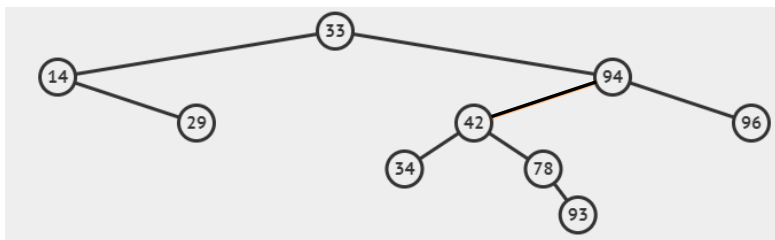
Thêm 34, 42



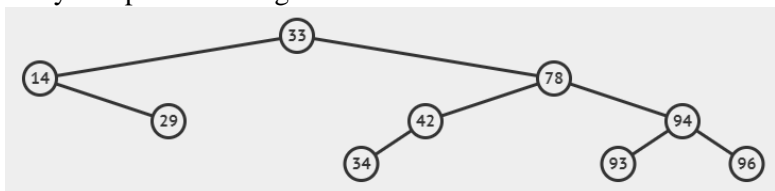
Xoay trái-phải / Left-right rotation



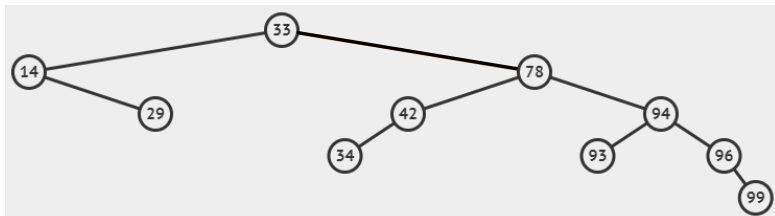
Thêm 93



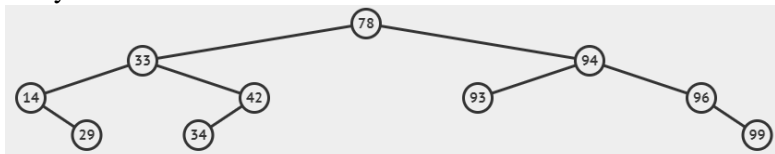
Xoay trái-phải / Left-right rotation



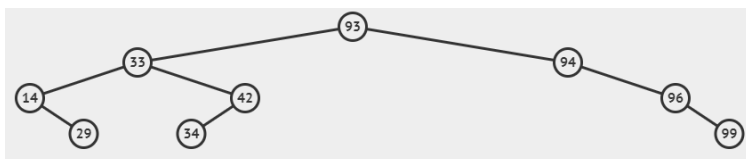
Thêm 99



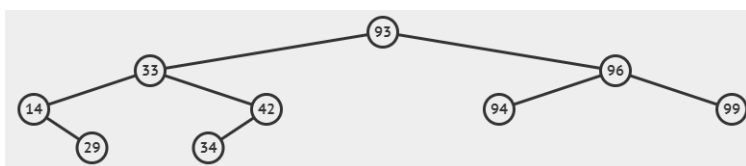
Xoay trái / Left rotation



b. Xóa nút 78 dùng successor



Xoay trái / Left rotation



5) Heap

Cho dãy số:

33 96 14 29 78 94 34 42 93 99

a. Xây dựng Binary Min Heap

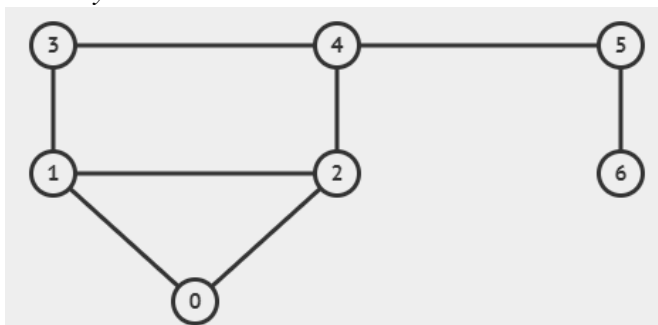
Step	1	2	3	4	5	6	7	8	9	10
1.	33									
2.	33	96								
3.	14	96	33							
4.	14	29	33	96						
5.	14	29	33	96	78					
6.	14	29	33	96	78	94				
7.	14	29	33	96	78	94	34			
8.	14	29	33	42	78	94	34	96		
9.	14	29	33	42	78	94	34	96	93	
10.	14	29	33	42	78	94	34	96	93	99

b. Extract Min

29	42	33	93	78	94	34	96	99
----	----	----	----	----	----	----	----	----

6) Graph

a. Biểu diễn đồ thị trên máy tính



Adjacency Matrix								
	0	1	2	3	4	5	6	
0		1	1	0	0	0	0	
1	1		1	1	0	0	0	
2	1	1		0	1	0	0	
3	0	1	0		1	0	0	
4	0	0	1	1		1	0	
5	0	0	0	0	1		1	
6	0	0	0	0	0	1		

Adjacency List			
0:	1	2	
1:	0	2	3
2:	0	1	4
3:	1	4	
4:	2	3	5
5:	4	6	
6:	5		

Edge List			
0:	0	1	
1:	0	2	
2:	1	2	
3:	1	3	
4:	2	4	
5:	3	4	
6:	4	5	
7:	5	6	

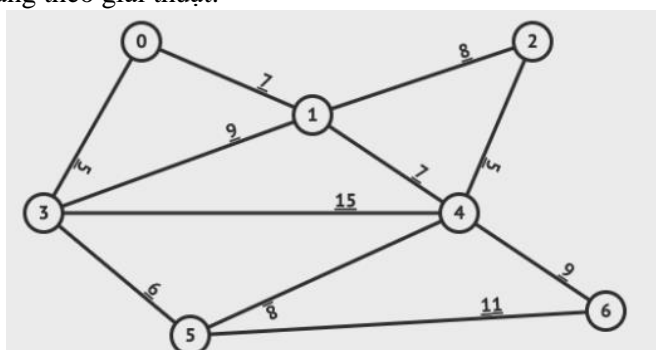
b. BFS/DFS từ đỉnh 0, ưu tiên đỉnh có số thứ tự nhỏ

BFS: 0 1 2 3 4 5 6

DFS: 0 1 2 4 3 5 6

c. Minimum Spanning Tree (MST)

- Trình bày bảng theo giải thuật:



i. Kruskal's

Edge	Selected?	Vertices of MST
5, 0-3	Yes	{0, 3}
5, 2-4	Yes	{0, 2, 3, 4}
6, 3-5	Yes	{0, 2, 3, 4, 5}
7, 0-1	Yes	{0, 1, 2, 3, 4, 5}
7, 1-4	Yes	{0, 1, 2, 3, 4, 5}
8, 1-2	No	{0, 1, 2, 3, 4, 5}
8, 4-5	No	{0, 1, 2, 3, 4, 5}
9, 1-3	No	{0, 1, 2, 3, 4, 5}
9, 4-6	Yes	{0, 1, 2, 3, 4, 5, 6}
11, 5-6	No	{0, 1, 2, 3, 4, 5, 6}
15, 3-4	No	{0, 1, 2, 3, 4, 5, 6}

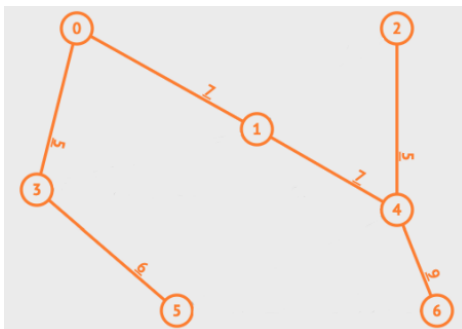
ii. Prim's

Xuất phát từ đỉnh 0

Priority Queue	Dequeued item	Added edge to MST
(5, 0-3), (7, 0-1)	5, 0-3	5, 0-3
(6, 3-5), (7, 0-1), (9, 3-1), (15, 3-4)	6, 3-5	6, 3-5
(7, 0-1), (8, 5-4), (9, 3-1), (11, 5-6), (15, 3-4)	7, 0-1	7, 0-1
(7, 1-4), (8, 1-2), (8, 5-4), (9, 3-1), (11, 5-6), (15, 3-4)	7, 1-4	7, 1-4
(5, 4-2), (8, 1-2), (8, 5-4), (9, 3-1), (9, 4-6), (11, 5-6), (15, 3-4)	5, 4-2	5, 4-2
(8, 1-2), (8, 5-4), (9, 3-1), (9, 4-6), (11, 5-6), (15, 3-4)	8, 1-2	-
(8, 5-4), (9, 3-1), (9, 4-6), (11, 5-6), (15, 3-4)	8, 5-4	-

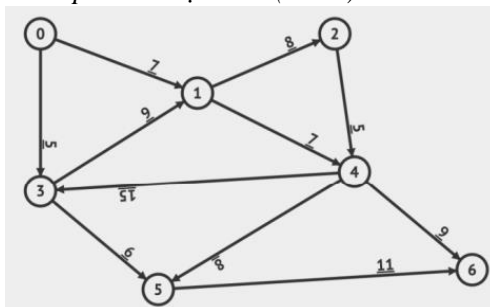
(9, 3-1), (9, 4-6), (11, 5-6), (15, 3-4)	9, 3-1	-
(9, 4-6), (11, 5-6), (15, 3-4)	9, 4-6	9, 4-6
(11, 5-6), (15, 3-4)	11, 5-6	-
(15, 3-4)	15, 3-4	-

- Vẽ lại cây khung nhỏ nhất



- Tính tổng trọng số
 $W = 5 + 5 + 6 + 7 + 7 + 9 = 39$

d. Tìm đường đi ngắn nhất xuất phát từ một đỉnh (SSSP)



- Trình bày bảng theo giải thuật
Xuất phát từ đỉnh 0

i. Bellman-Ford's

Liệt kê thứ tự cạnh chạy giải thuật, nên xếp theo thứ tự đã được hướng dẫn trên lớp lý thuyết (liệt kê cạnh theo thứ tự tăng dần đỉnh source tăng dần đỉnh destination)

	<u>0</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>
-	0, 0	-1, ∞	-1, ∞	-1, ∞	-1, ∞	-1, ∞	-1, ∞
Pass 1	0, 0	0, 7	1, 15	0, 5	1, 14	3, 11	5, 22
Pass 2	0, 0	0, 7	1, 15	0, 5	1, 14	3, 11	5, 22
Pass 3	0, 0	0, 7	1, 15	0, 5	1, 14	3, 11	5, 22
Pass 4	0, 0	0, 7	1, 15	0, 5	1, 14	3, 11	5, 22
Pass 5	0, 0	0, 7	1, 15	0, 5	1, 14	3, 11	5, 22
Pass 6	0, 0	0, 7	1, 15	0, 5	1, 14	3, 11	5, 22

Lưu ý:

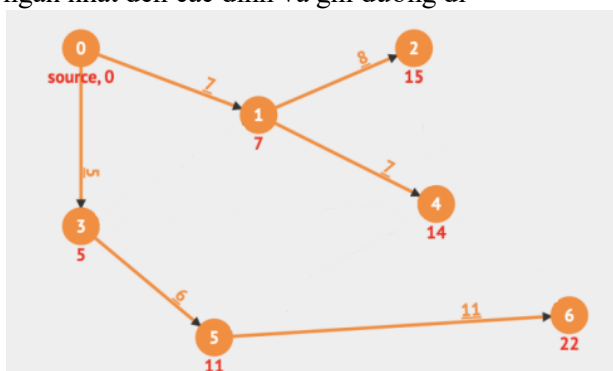
- Chỉ dừng sớm khi đã thể hiện 2 pass liên tiếp giống nhau hoặc có thể ghi đủ V-1 pass.
- Mỗi pass chỉ cần thể hiện kết quả ở cuối mỗi pass (đã relax sau khi xét hết tất cả cạnh).

ii. Dijkstra's

	<u>0</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>
-	0, 0	-1, ∞	-1, ∞	-1, ∞	-1, ∞	-1, ∞	-1, ∞
Step 1				<u>0, 5</u>			
Step 2		<u>0, 7</u>					
Step 3						<u>3, 11</u>	
Step 4					<u>1, 14</u>		
Step 5			<u>1, 15</u>				
Step 6							<u>5, 22</u>

Lưu ý:

- Chú ý cẩn thận các bước có relax những đỉnh chưa được chọn.
- Vẽ lại đường đi ngắn nhất đến các đỉnh và ghi đường đi



0 – 1 – 2

0 – 1 – 4

0 – 3 – 5 – 6

- Vẽ bảng thể hiện khoảng cách từ đỉnh xuất phát đến các đỉnh còn lại

V	D
0	0
1	7
2	15
3	5
4	14
5	11
6	22