CS301 All Current
Final Term Paper Subjective 2013
Solved with refernces

Pattren:

Total Questions: 52

Total Marks: 80

Total MCQs: 40 (Each of 1 Mark)
Total Short Questions: 4 (Each of 2 Mark)
Total Short Questions: 4 (Each of 3 Mark)
Total Long Questions: 4 (Each of 5 Mark)
Subjective questions:
Questions of 2 marks

Note: Try to Prepare Mcqz and subjective files of Moaaz brother.

1. In the array representation of union what represents -1?

Answer: Page no 392 Chapter no 34

This shows that this node has no parent. Moreover, this node will be a root that may be a parent of some other node.

2. For smaller lists, linear insertion sort performs well, but for larger lists, quick sort is suitable to apply." Justify why?

Answer:

Since for smaller lists number of comparisons and interchange (if needed) decreases radically in insertion sort than quick sort. For larger lists, opposite occurs in both sorting techniques.

3. If we want to delete the node from BST which has left and right child then which rotation is applied?

Answer:

After deleting the node we traverse up the tree from the deleted node checking the balance of each node at each level up to the root.

4. Collision in hashing definition?

Answer: page no 464 chapter no 42

Collision takes place when two or more keys (data items) produce the same index.

Question of 3 marks:

1. Algorithm union by weight?

Answer: Page no 408

```
//union(i,j):
1.  root1 = find(i);
2.  root2 = find(j);
3.    if (root1 != root2)
4.    if (parent[root1] <= parent[root2])
5.    {
6.        // first tree has more nodes
7.        parent[root1] += parent[root2];
8.       parent[root2] = root1;
9.    }
10.    else
11.    {
12.        // second tree has more nodes
13.        parent[root2] += parent[root1];
14.        parent[root1] = root2;
15.    }
```

2. One tree is given question is it heap or not if it is heap then write its type

Answer: Consult Page no 333-335 Ch:29

Due to the violation, the value of child is less than that of the parent it is not a heap

3. Which data structure is best for priority queue?

Answer: Page no 334 Ch: 29

The use of priority queue with the help of heap is a major application.

The priority queue is itself a major data structure, much-used in operating systems. Similarly priority queue data structure is used in network devices especially in routers. Heap data structure is also employed in sorting algorithms.

Questions of 5 marks:

1. Some numbers are given and using those make BST?

Answer:

Page no: 128

2. One array is given we require to sort it using bubble sort and write only 2 iterations?

Answer:

Page no: 486

3. One tree is given which not the heap but after minimum changes it becomes max heap make it?

Answer:

Chapter no 29 page no 335

==========

1. name of two divide and conquer algorithm

Answer:

Merge Sort

Quick Sort

Heap Sort

2. Height of a tree is 5 find sum of heights

Answer:

Total number of nodes $N = 2^{h+1} - 1 = 2^{5+1} - 1 = 2^6 - 1 = 63$

3. What is a skip List?

Answer: Page no 444 Chapter: 39

- A skip list for a set S of distinct (key, element) items is a series of lists $S_0, S_1, \ldots, S_h$ such that
  - Each list S_i contains the special keys $+\infty$ and $-\infty$
  - List S_0 contains the keys of S in non-decreasing order Each list is a subsequence of the previous one, i.e., $S_0 \supseteq S_1 \supseteq \cdots \supseteq S_h$
  - List S_h contains only the two special keys

What is an Equivalent relation?

Answer: Page no 424 Chapter : 37

Two cells are equivalent if they can be reached from each other.

4 questions of 3 marks

1. What are the properties of equivalence class?

Answer: Page no 385 Chapter : 33

- A binary relation R over a set S is called an *equivalence relation* if it has following propertiesReflexivity: for all element x ∈ S, x R x
  2. Symmetry: for all elements x and y, x R y if and only if y R x
  3. Transitivity: for all elements x, y and z, if x R y and y R z then x R z

2. How heap sort works to set a set of data?

Answer: Chapter:30 Page no 353

When we construct the heap or complete binary tree of the list, the smallest name (considering alphabet 'a' as the smallest) in the list will take the root node place in the tree. The remaining names will

take their places in the nodes below the root node according to their order. Remember, we are not constructing binary search tree but a min-heap to sort the data. After the construction of the min-heap from all the names in the list, we start taking the elements out (deleting) from the heap in order to retrieve the sorted data. The first element that is taken out would be the smallest name in the heap. The remaining heap after taking out the deleted node will be reduced by one element in size and the next name will be at the root position. It is taken out to further reduce the tree size by one.

This way, the continuation of the process of taking out the elements from the heap

will ultimately lead to a situation when there is no more node left in the tree.

3. How can we search an element in skip list?
   Answer: Page no 447 Chapter : 40

We search for a key x in the following fashion:

- We start at the first position of the top list
- At the current position p, we compare x with $y \leftarrow key(after(p))$
- $x = y$: we return $element(after(p))$
- $x > y$: we "scan forward"
- $x < y$: we "drop down"
- If we try to drop down past the bottom list, we return NO_SUCH_KEY

4. Give one example of Hashing
   Answer:
   page no 460 Chapter no 41

4 questions of 5 marks.

1. How we can implement Table ADT using linked?

Answer: Page no 441 Chapter no 39

- *TableNodes are again stored consecutively (unsorted or sorted)*
- ***insert**: add to front; (1or n for a sorted list)*
- ***find**: search through potentially all the keys, one at a time; (n for unsorted or for a sorted list*
- ***remove**: find, remove using pointer alterations; (n)*

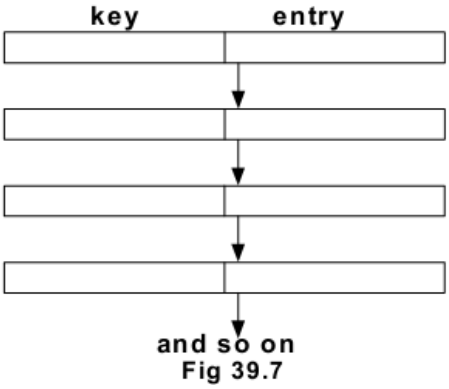


Fig 39.7

2. Suppose we have the following representation child nodes and Parent node of the node D.

Answer:

question is not clear

3. .Consider the following max heap. Add node 24 in it and show the resultant Heap.

Answer:

Page no 337-340 Chapter no 29

4. Give any three characteristics of Union by Weight method

Answer: Page no 408 Chapter no : 36

- Maintain sizes (number of nodes) of all trees, and during *union*.
- Make smaller tree, the subtree of the larger one.
- Implementation: for each root node `i`, instead of setting `parent[i]` to `-1`, set it to `-k` if tree rooted at `i` has `k` nodes.
- This is also called *union-by-weight*.

Note: New Paper (All questions were same)

2. Give the effect of sorted data on Binary Search.

Answer: Page no 205

It gives a look of a linked list

5. Give the operation names that we can perform on Table abstract data type?

Answer: Page no : 429

Insert

As the name shows this method is used to insert (add) a record in a table. For its execution, a field is designated as key. To insert a record (entry) in the table, there is need to know the key and the entry. The insert method puts the key and the other related fields in a table. Thus we add records in a table.

Find

Suppose we have data in the table and want to find some particular information. The find method is given a key and it finds the entry associated with the key. In other words, it finds the whole record that has the same key value as provided to it. For example, in employees table if employee id is the key, we can find the record of an employee whose employee id is, say, 15466.

Remove

Then there is the remove method that is given a value of the key to find and remove the entry associated with that key from the table.

<u>4 questions of 5 marks.</u>

1. Heapify the elements of the following array (reading from left to right) into a Min Heap and show that Min Heap contents in the form of array as shown below,

original array	6	5	3	9	1	2	10	8	-
	1	2	3	4	5	6	7	8	

Answer:

2(i)=2

2(i)+1=4+1=5

2. Give any three characteristics of Union by Weight method.

Answer: Repeat

1. Here is an array with exactly 15 elements: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15. Suppose that we are doing a binary search for an element. Indicate any elements that will be found by examining two or fewer numbers from the array.

Answer:

4. Explain the following terms:

1. Collision

2. Linear Probing

3. Quadratic Probing

Collision:

Answer: page no 464 chapter no 42

Collision takes place when two or more keys (data items) produce the same index.

Quadratic Probing:

Answer: Page no 466 chapter no 42

Quadratic probing uses different formula:

- Use $F(i) = i^2$ (square of i) to resolve collisions
- If hash function resolves to H and a search in cell H is inconclusive, try $H + 1^2$, $H + 2^2$, $H + 3^2$, …

In the quadratic probing when a collision happens we try to find the empty location at index + $1^{2.}$ If it is filled then we add 2^2 and so on. Let's take the above example. We want to insert seagull. The hash function generates the value 143 and this location is already occupied. Now we will start probing. First of all, we add 1^2 in the 143 and have 144. It is also not empty. Now we will add 2^2 in the 143 and have 147. If it is also occupied, we will add 3^2 in 143 and we have 152. The data is now getting scattered. Unfortunately, there are some problems with quadratic probing also.

Linear Probing:

Answer: Page no 470 chapter no 42

The first solution is known as open addressing. When there is a collision, we try to find some other place in our array. This approach of handling collisions is called *open addressing*; it is also known as *closed hashing*. The word open is used with addressing and the word closed is used with hashing. Be careful when naming these. More formally, cells at $h_0(x)$, $h_1(x)$, $h_2(x)$, … are tried in succession where

$hi(x) = (hash(x) + f(i)) \bmod TableSize$, with $f(0) = 0$.

Here *hash* is our hash function. If there is some collision, we add *f(i)* value to it before taking its mod with *TableSize*. The function, *f*, is the collision resolution strategy.

We use $f(i) = i$, i.e., *f* is a linear function of *i*. Thus

$location(x) = (hash(x) + i) \bmod TableSize$

The function *f* can be any function. So our first implementation of *f* is that whatever integer (i) we gave to it, it returns it back. That is f(i) is a linear function. We can implement function *f* as it returns the square of *i*. That will be quadratic function. Similarly, we can have cubic functions. We are free to implement it.

Write the C++ code which will print the integers from 1 to 100 through the recursive function. (5)

Suppose the hash table of height 7 (index 0 to 6), hash function H (key) = (2 *key + 5) Mode 7 and pass these numbers [5, 23, 17, 14, 44] from the hash function to resolve the collision by the linear probing. (5)

Answer:

H (key) = (2 *key + 5) Mode 7

Mode = 15mode7= 7(2) + 1, in which 1 is the remainder. And this reminder is mode.

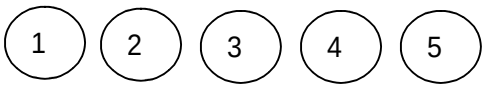
H(5)=(2*5+5)mode7=2*5+5=15mode7=1

H(23)= (2*23+5)mode7=2

....so on

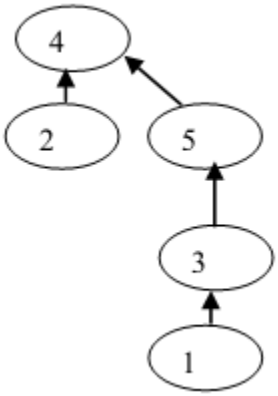
An array was given and we have to draw maximum-heap and also show the resultant array. (5)

Consider the following sequence of union commands on the set of elements {1,2,3,4, 5}:



union(4,2)
union(3,1)
union(5,4)
union(5,3)

Show the result when the unions are performed (5)
Answer:



How can we delete an element from the skip list? (3)

AnsweR:

How we can generate a maze? Draw the algorithm of it. (3)

Answer:- (Page 424)

How can we generate maze? The algorithm is as follows:

- Randomly remove walls until the entrance and exit cells are in the same set.
- Removal of the wall is the same as doing a union operation.
- Do not remove a randomly chosen wall if the cells it separates are already I the same set.

Why binary search not use with link list? (2)

Answer:

Because it store sorted data.

Q2.

What is heap data structure.Write its types.

Repeat

Q3.

When the size of hash table is big enough that the size of input data, then there is less chance for cllision hashing. Write the drawback of usuing this.

Q4.

What is the use of *this* pointer?

Q5.

Writa three characteristics of union by weight.

Answer:- (Page 408)

Following are the salient characteristics of this method:

- Maintain sizes (number of nodes) of all trees, and during *union*.
- Make smaller tree, the sub-tree of the larger one.
- Implementation: for each root node i, instead of setting parent[i] to -1, set it to -k if tree rooted at i has k nodes.

Q6.

Let's call the node as "A" that require rebalancig, considered the two cases below

- An insertion into left subtree of the left child of A
- An insertion into the right subtree of the right child of A

which statement is true in these cases;

Single rotation

Double rotation

Answer: (page 230)

Insertion occurs on the *inside* in *cases 2* and *3* which a single rotation cannot fix.

Q7

Array is given

3 5 9 8 1 7 0 2 6 4

Draw after 1st iteration of main loop in selection sort algorithm.

AnsweR:

0 5 9 8 1 7 3 2 6 4

Q8

How does heap sort works?

Answer:- (Page 334)

Min heap and max heap are used to sort the data. Min heap convert the data in ascending sorted order and max heap convert the in descending order.

Q9

Write public method of remove_first() into the linked list,that remove the 1st item of the list.

Q10

Sort the data items given below with insertion sort algorithm.Write each step

77 33 44 88 11 22 66 55

AnsweR:

33 77 44 88 11 22 66 55

33 44 77 88 11 22 66 55

33 44 77 88 11 22 66 55

33 44 11 77 88 22 66 55

11 33 44 77 88 22 66 55

11 33 44 77 22 88 66 55

11 33 44 22 77 88 66 55

.

.

11 22 33 44 55 66 77 88 = final

My CS301 paper dated 24 feb 2013

Total questions 52

40 MCQ's mostly from past paers specially ye waly 2

If there are N external nodes in a binary tree then what will be the no. of internal nodes in this binary tree?

Answer: N-1

If there are N internal nodes in a binary tree then what will be the no. of external nodes in this binary tree?

Answer: N+1

Heap and it's type.

Suppose we have the following representation for a complete Binary Search Tree, tell the Left and Right child nodes and Parent node of the node D

A B C D E F G H I J K L M N O P Q R S T ...

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 ...

Answer:

2(i)+1=right node=2*4+1=9=I

2(i)=left node=2*4=8=H

i/2=parent node=4/2=2=B

My Paper:

Q:1 A complete binary tre was given we will have to find out

- Left child
- Right child
- Parent

Answer:

right node =2(i)+1

left node =2(i)

parent node i/2=

Q:2 Hashing function h(x)= (x*2)%table size the values. Table size=10

Values: 11, 29, 22,36,7 Handle collusion with chaining.

0	
1.	
2.	11
3.	
4.	
5.	
6.	

7.	
8.	
9.	
10.	

ANSWER:

h(x)= (x*2)%table size

h(11)=(11*2)%10=22%10=2 position => 22Mode10=10*2+2=22 so 2 is mod

If any collusion occur use

Quardatic function=1^2+H for collusion.

Do remaining by yourself.

Q:3 Write a non recursive procedure a pointer to the node of a binary search tree of integers and a pointer to a new created node which insert the new node as a leaf node in right place.

Answer:

Q:4 Find out union(mooaz bhaiya)

AnsweR: Repeat

Q:5 A hash function was given and f(h)=keymod12 identify key that may cause collusion.

36 ,30 ,84

AnsweR:

f(h)=36mod12=12*3+0=0

f(h)=30mod12=12*2+6=6

f(h)=84mod12=12*7+0=0

so key 36 and 84 collide each other.

Q:6 A program was given the function of BST_Fun was asked to write.

Q:7 Suppose you have 8 nodes with following intilization after union show resultant array. Union of (3,7)(5,6)(3,8)

1	-1
2	-1
3	-1
4	-1
5	-1
6	-1
7	-1
8	-1

Q:8 Properties of good hash function?

Answer:

If the keys are integers then *key%T* is generally a good hash function unless the data has some undesirable features.

Q:9 What is heap write down its types?

Answer:- (Page 333)

Heap is a data structure of big use and benefit. It is used in priority queue. "The definition of heap is that it is a complete binary tree that conforms to the heap order". Here heap order means the min and max heap. Max heap: In max heap, each node has a value greater than the value of its left and right child nodes. Moreover, the value of the root will be largest and will become lesser at downward levels. Min heap: in a (min) heap for every node X, the key in the parent is smaller than (or equal to) the key in X. This means that the value in the parent node is less than the value on its children nodes.

Q:10 Show resultant heap in the form of array after insertion of 86

1	97

2	76
3	61
4	42
5	54
6	59
7	31
8	32
9	17
10	44
11	19
12	

1	97
2	76
3	86
4	42
5	54
6	61
7	31
8	32
9	17
10	44
11	19
12	59

Q: 11: What is image segmentation?

Answer:

Image Segmentation

In image segmentation, we will divide the image into different parts. An image may be segmented with regard to the intensity of the pixels.

Q: 12 In which traversal method recusion is not allowed?

AnsweR: not sure

In post order traversal method