@combinecs The Extra Step

Data Structures & Algo

FREE

मजबुद इरादा | JRF का वादा

LIVE

YouTube

3:30 pm

JULY 18

MOCK-4

like !!

RASHMI PRABHA

Qualified UGCNET, GATE Educator

10+ Years of Experience

CombineCS Schedule

ARE YOU PREPARED?

LIVE MOCK TEST	LIVE DATE	TIMING
~~UGCNET Artificial Intelligence~~	~~Sun, 11th JULY~~	~~@ 3:30 pm~~
UGCNET Data Structure & Algorithm	Sun, 18th JULY	@ 3:30 pm
UGCNET DBMS	Sun, 25th July	@ 3:30 pm

AI
DSA

UPCOMING LIVE SESSIONS	LIVE DATE	TIMING
PAPER – 1 (Computer Science)	DAILY	@ 9am

Subscribe channel to get Job notifications & access to free resources

Last chance to enroll by 25th July

Ready?

Searching

MCQ

Q1) What is the best case for linear search?

a) O(nlogn)
b) O(logn)
c) O(n)
d) O (1)

Q1) What is the best case for linear search?

a) O(nlogn)

b) O(logn)

c) O(n)

d) O (1)

Answer: d

Explanation: Best case, means the element is

at the head of the array, hence O (1).

Pass

Q2) What is the worst case for linear search?

a) O(nlogn)
b) O(logn)
c) O(n)
d) O (1)

Q2) What is the worst case for linear search?

a) O(nlogn)

b) O(logn)

c) O(n)

d) O (1)

Answer: c

Explanation: Worst case means, when the desired element is at the tail of the array or not present at all, in this case you have to traverse till the end of the array, hence the complexity is O(n).

Q3) What is the best case and worst-case complexity of ordered linear search?

a) O(nlogn), O(logn)
b) O(logn), O(nlogn)
c) O(n), O(1)
d) O (1), O(n)

Q3) What is the best case and worst-case complexity of ordered linear search?

a) O(nlogn), O(logn)

b) O(logn), O(nlogn)

c) O(n), O(1)

d) O(1), O(n)

time

ascending → 1, 2, 3, 4, 5

descending → 5, 4, 3, 2, 1

Answer: d

Explanation: Although ordered linear search is better than unordered when the element is not present in the array, the best and worst cases still remain the same, with the key element being found at first position or at last position.

Q4) Which of the following is a disadvantage of linear search?

a) Requires more space
b) Greater time complexities compared to other searching algorithms
c) Not easy to understand
d) Not easy to implement

Q4) Which of the following is a disadvantage of linear search?

a) Requires more space

b) Greater time complexities compared to other searching algorithms

c) Not easy to understand

d) Not easy to implement

Answer: b

Explanation: The complexity of linear search as the name suggests is O(n) which is much greater than other searching techniques like binary search(O(logn)). Linear search is easy to implement and understand than other searching techniques.

CombineCS
The Extra Step

SUBSCRIBE

DSA

August

Q5) What is the recurrence relation for the linear search recursive algorithm?

a) $T(n-2) + c$
b) $2T(n-1) + c$
c) $T(n-1) + c$
d) $T(n+1) + c$

Q5) What is the recurrence relation for the linear search recursive algorithm?

a) T(n-2) +c

b) 2T(n-1) +c

c) T(n-1) +c

d) T(n+1) +c

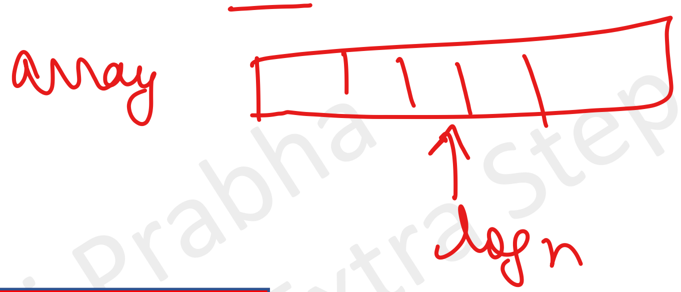
Answer: c

Explanation: After each call in the recursive algorithm, the size of n is reduced by 1. Therefore, the optimal solution is T(n-1) +c.

Q6) What is the worst-case complexity of binary search using recursion?

a) O(nlogn)
b) O(logn)
c) O(n)
d) $O(n^2)$

Q6) What is the worst-case complexity of binary search using recursion?



a) O(nlogn)

b) O(logn)

c) O(n)

d) O(n^2)

Same in Average Case

Answer: b

Explanation: Using the divide and conquer master theorem.

Q7) What is the recurrence relation for the binary search recursive algorithm? (ISRO-2017)

O(nlogn) = Merge = heap = Quick (O(n)^2)

Sorting

a) $2T(n/2) + c$

b) $T(n/2) + c$

c) $T(n/2) + \log n$

d) $T(n/2) + n$

Q7) What is the recurrence relation for the binary search recursive algorithm? (ISRO-2017)

a) 2T (n / 2) + c

b) T (n / 2) + c

c) T (n / 2) + log n

d) T (n / 2) + n

Q8) Binary Search can be categorized into which of the following?

a) Brute Force technique
b) Divide and conquer
c) Greedy algorithm
d) Dynamic programming

3:30

3:30 pm

Sunday

half

DAC

Q1 → MCQ

telegram

Assignment

Properties

Q8) Binary Search can be categorized into which of the following?

a) Brute Force technique

b) Divide and conquer

c) Greedy algorithm

d) Dynamic programming

Answer: b

Explanation: Since 'mid' is calculated for every iteration or recursion, we are diving the array into half and then try to solve the problem.

difficult

Basic

linear | binary

Gate PYQ

Q9)Which of the following is correct recurrence for worst case of Binary Search? Gate

don't

1. $T(n) = 2T(n/2) + O(1)$ and $T(1) = T(0) = O(1)$
2. $T(n) = T(n-1) + O(1)$ and $T(1) = T(0) = O(1)$
3. $T(n) = T(n/2) + O(1)$ and $T(1) = T(0) = O(1)$
4. $T(n) = T(n-2) + O(1)$ and $T(1) = T(0) = O(1)$

Combine CS
The Extra Step

SUBSCRIBE

Q9)Which of the following is correct recurrence for worst case of Binary Search?

Best O(1) Worst O(log n) Average O(log n)

merge sort

1. $T(n) = 2T(n/2) + O(1)$ and $T(1) = T(0) = O(1)$
2. $T(n) = T(n-1) + O(1)$ and $T(1) = T(0) = O(1)$
3. $T(n) = T(n/2) + O(1)$ and $T(1) = T(0) = O(1)$
4. $T(n) = T(n-2) + O(1)$ and $T(1) = T(0) = O(1)$

T(n/2) + O(1)

Q10) The average number of key comparisons done in a successful sequential search in a list of length it is …..(GATE CS 1996 / ISRO CS 2016)

1. log n
2. (n-1)/2
3. n/2
4. (n+1)/2

1) take your time

2) Read twice

Q10) The average number of key comparisons done in a successful sequential search in a list of length it is ….. **(GATE CS 1996 / ISRO CS 2016)**

1. log n
2. (n-1)/2
3. n/2
4. (n+1)/2

If element is at 1 position then it requires 1 comparison. If element is at 2 position then it requires 2 comparison. If element is at 3 position then it requires 3 comparison. Similarly , If element is at n position then it requires n comparison.

Total comparison = n(n+1)/2

For average comparison = (n(n+1)/2) / n

= (n+1)/2

Combine CS
The Extra Step

Q11) The average case occurs in the Linear Search Algorithm when:

1. The item to be searched is in some where middle of the Array
2. The item to be searched is not in the array
3. The item to be searched is in the last of the array
4. The item to be searched is either in the last or not in the array

Q11) The average case occurs in the Linear Search Algorithm when:

Best case -> element first index

1. The item to be searched is in some where middle of the Array
2. The item to be searched is not in the array – worst case
3. The item to be searched is in the last of the array – worst case
4. The item to be searched is either in the last or not in the array

worst case

Combine CS
The Extra Step

SUBSCRIBE

Q12) Number of comparisons required for an unsuccessful search of an element in a sequential search, organized, fixed length, symbol table of length L is (ISRO CS 2011)

hint: tree

like !!

Scheduling

1. L
2. L/2
3. (L+1)/2
4. 2L

Q12) Number of comparisons required for an unsuccessful search of an element in a sequential search, organized, fixed length, symbol table of length L is (ISRO CS 2011)

1. L
2. L/2
3. (L+1)/2
4. 2L

half → O (log n)

Q13) The time taken by binary search algorithm to search a key in a sorted array of n elements is ..(ISRO 2007)

sorted array

1 2 3 4 5

1. O ($\log_2 n$)
2. O (n)
3. O (n $\log_2 n$)
4. O (n_2)

Combine CS
The Extra Step

SUBSCRIBE

Q13) The time taken by binary search algorithm to search a key in a sorted array of n elements is ..(ISRO 2007)

1. O ($\log_2 n$)
2. O (n)
3. O (n $\log_2 n$)
4. O (n_2)

worst

① Best, Worst, Average

Assume: Worst

O(1)

Q2 Is it mandatory for Binary Search, element should be sorted? Yes NO

Linear? NO

offline
Comment
suggestions
Live Test

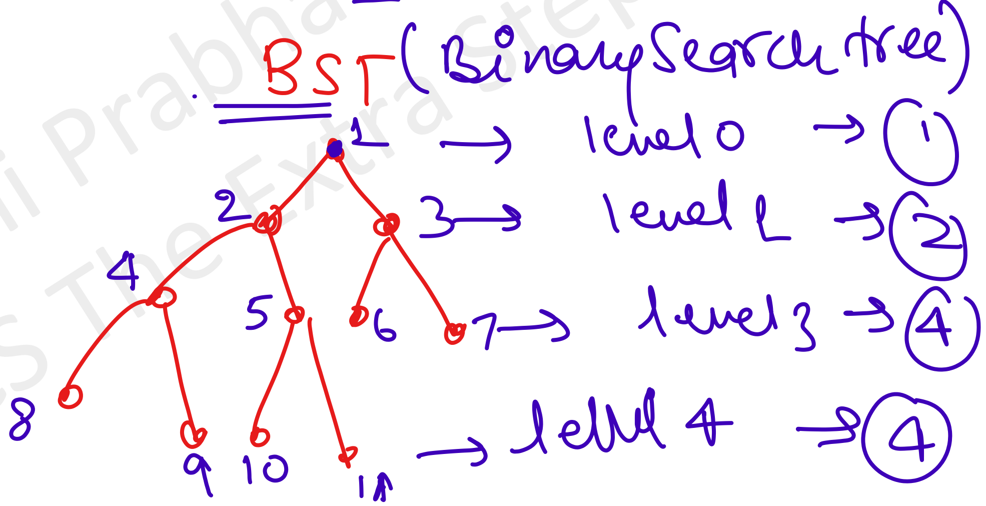
Monday
DS

Q14) Suppose there are 11 items in sorted order in an array. How many searches are required on the average, if binary search is employed and all searches are successful in finding the item? (ISRO CS 2014)

1. 3.00
2. 3.46
3. 2.81
4. 3.33

Q14) Suppose there are 11 items in sorted order in an array, How many searches are required on the average, if binary search is employed and all searches are successful in finding the item? (ISRO CS 2014)

1. 3.00
2. 3.46
3. 2.81
4. 3.33



Total number of caparisons required = 1*1 + 2*2 + 4*3 + 4*4 = 33 Average comparisons required for 11 items = 33/11 = 3

Topic FAQ

August

Q15) Is there any difference in the speed of execution between linear search(recursive) vs linear search(iterative)?

a) Both execute at same speed

b) Linear search(recursive) is faster

c) Linear search (Iterative) is faster

d) Can't be said

Assignment →
& what is the diff. b/w Recursive & Iterative

Q15) Is there any difference in the speed of execution between linear search(recursive) vs linear search(iterative)?

a) Both execute at same speed
b) Linear search(recursive) is faster
c) Linear search (Iterative) is faster
d) Can't be said

Answer: c

Explanation: The Iterative algorithm is faster than the latter as recursive algorithm has overheads like calling function and registering stacks repeatedly.

CombineCS
The Extra Step

Last chance to enroll by 25th July

SUBSCRIBE

Highlights

1. Live session
2. 100% doubts resolution
3. 1-to-1
4. Notes provided
5. 4 hrs live class

Recursion – LOC
↳ method call overhead
space & time (high)

← Call on this no

LMN

extra

1 2 3

(2)

array → last index

Algorithm	Best Time	Average	Worst	Space Complexity
Linear Search	O(1)	O(n)	O(n)	O(1)
Binary Search	O(1)	O(log n)	O(log n)	O(1)

same

array list first

(1) (2)

DAC approach

sorted order

Combine CS
The Extra Step

SUBSCRIBE

Algorithm	Best Time Complexity	Average Time Complexity	Worst Time Complexity	Worst Space Complexity
Linear Search	O(1)	O(n)	O(n)	O(1)
Binary Search	O(1)	O(log n)	O(log n)	O(1)
Bubble Sort	O(n)	O(n^2)	O(n^2)	O(1)
Selection Sort	O(n^2)	O(n^2)	O(n^2)	O(1)
Insertion Sort	O(n)	O(n^2)	O(n^2)	O(1)
Merge Sort	O(nlogn)	O(nlogn)	O(nlogn)	O(n)
Quick Sort	O(nlogn)	O(nlogn)	O(n^2)	O(log n)
Heap Sort	O(nlogn)	O(nlogn)	O(nlogn)	O(n)
Bucket Sort	O(n+k)	O(n+k)	O(n^2)	O(n)
Radix Sort	O(nk)	O(nk)	O(nk)	O(n+k)
Tim Sort	O(n)	O(nlogn)	O(nlogn)	O(n)
Shell Sort	O(n)	O((nlog(n))^2)	O((nlog(n))^2)	O(1)

covered on 18 July 2021

Will be covering in next week

Stay Subscribed

CombineCS Schedule

ARE YOU PREPARED?

LIVE MOCK TEST	LIVE DATE	TIMING
~~UGCNET Artificial Intelligence~~	~~Sun, 11th JULY~~	~~@ 3:30 pm~~
UGCNET Data Structure & Algorithm	Sun, 18th JULY	@ 3:30 pm
UGCNET DBMS	Sun, 25th July	@ 3:30 pm

telegram, whatsapp

concept

next week's Monday

UPCOMING LIVE SESSIONS	LIVE DATE	TIMING
PAPER – 1 (Computer Science)	DAILY	@ ~~9am~~ 2:30 pm (free)

Subscribe channel to get Job notifications & access to free resources

searching

👉 Follow us on Social media:

▶ YouTube : https://www.youtube.com/c/CombineCSTheExtraStep
Facebook : https://www.facebook.com/groups/combinecs
Instagram : https://www.instagram.com/combinecs/
Telegram Group : https://t.me/RashmiCCS
Telegram Channel : https://t.me/combinecs

👉 Join our WhatsApp group for (NET/SET/GATE):
https://chat.whatsapp.com/GruovhRvste1nL8L2X1YQ3
👉 Join our WhatsApp group for (SCHOOL/JOB Notifications) :
https://chat.whatsapp.com/ExM4CZ2ZKxzEgPvSfOXNFb

For any query regarding notes, pdf, feedback, suggestions
Mail us: combinecs2020@gmail.com
For all our latest courses launched
visit: 🌍 combinecs.com
==
"Effort Never Dies"
👍 Like || Share || Comment || Subscribe

Thank you

offline

live class
Computer Graphics
Projections
DSA

Join Batch

doubt?

course now!

Post your doubts in comment section.
Stay subscribed for all updates.