HACK YALE

INTERVIEW PREP

WWW.HACKYALE.COM

GETTING AN INTERVIEW

- Big Companies
  - Attend on campus session/career fair. Get their email address and/or give them your resume.
- Startups
  - Be proactive. Look at hnhiring.com to see startup jobs.
  - Actively email people at those companies, send resume with a brief description of why you're applying.
  - Github/prior experience helps a ton!

HACK YALE

INTERVIEW STRUCTURE

- Normal interview (90%+)
  - Two questions, 45 minutes.
  - Algorithm and data structure oriented.
  - Phone/in-person, Google Doc or collabedit.com
- Alternative interview
  - Language specific (How does C++ implement virtual functions?)
  - Class/system specific (Describe Linux VMM)
  - Experience specific (Describe a side project/tough homework assignment)

HACK YALE

GENERAL APPROACH

- Use a checklist (more on this later).
- Pick one language and be proficient at it. Preferably C,C++, or Java.
- Talk/think out loud, so they know you’re not drawing a blank.
- Develop an approach, always test it out with small cases/edge cases. Much harder to backtrack after you’ve written code.

HACK YALE

HOW TO APPROACH PROBLEMS

- Every problem is a combination of algorithm + data structure.
- When deciding how to solve a problem, use a checklist to decide on the data structure, and the algorithm flows from there.

HACK YALE

WORDS OF WISDOM

"Data dominates. If you've chosen the right data structures and organized things well, the algorithms will almost always be self-evident. Data structures, not algorithms, are central to programming."

–Rob Pike

HACK YALE

CHECKLIST FOR DATA STRUCTURES

- Array
- Stack
- Queue
- Heap
- Binary Tree
- Hash Table

HACK YALE

CHECKLIST FOR ALGORITHMS

- Brute Force
- Greedy
- Divide & Conquer (sorting)
- Backtracking (DFS/BFS)
- Dynamic Programming
- Randomized ("wat?")

HACK YALE

OTHER THINGS TO KNOW

- Bitwise operations
- Network Flows (Min-cut Max-flow)
- OOP patterns
- Databases (ACID, transactions)
- Networks (“How does a browser work?”)
- Compilers (“Stack vs. Heap allocation”)

HACK YALE

PROBLEMS

- In an integer array with N elements (N is large), find the minimum k elements (k << N).
- Write a function to convert a string to an integer (atoi).
- Write a function to find the height of a balanced binary tree.
- Find the minimum element in a binary tree.
- Write a program to calculate pow(x,n).
- Write strtok(char *s, char *delim).
- Write an in-order tree traversal iteratively and recursively.

HACK YALE

SOME TAKEAWAYS

- Recursion can always be simulated with a stack.
- Always start with small, easy examples, then consider edge cases. You usually don't need big examples.
- Ways to get edge cases: NULL input, negative input, corrupt pointers.
- Always have error checking code.
- Clarify before writing code!
- Have a rough idea of the solution runtime. It's okay to discuss O(n^2) or O(n) but make sure that O(logn) isn't possible.
- Sorting algorithms can be no faster than O(nlogn) for generic sorts. Recognize situations where sorting can/can't help, and adjust expectations accordingly.

HACK YALE

RESOURCES

- http://dus.cs.yale.edu/Advice/interviews.html (1st 2 books are great, 3rd is overkill). Practice from both!
- http://www.geeksforgeeks.org/
- http://www.programminginterview.com/content/pdf-version-ultimate-guide
- “The C Companion” (A. Holub) and “Advanced C Tips and Techniques” (Anderson) are great for improving as a programmer.

HACK YALE

THANKS!

HACK YALE