[Cheat Sheet] - Interviewing in python

Criterias Evaluated

1. Prior experience
2. Culture fit
3. **Code ability**
4. Analytical abilities

The interview Process

2-6 rounds of interview

ask the job person interviewing you

> HR is more about resume and work experience
> Engineer is more technical

They might call you and make you code on a piece of paper and explain it to the interviewer

coding and algorithm questions

no specific questions or google the questions

technical questions depend on the main focus of the company - database company has more questions about the topic

Four main things a candidate is being evaluated:

1. Experience
2. Culture Fit - be a nice person
3. Coding Skills - solving the questions
4. Analytical Ability - even if you analise the question correctly but fail the exercise, they may give you a good analytical ability review

10 Mistakes on a interview

1. Practice on a computer - most times you will need to code on a piece of paper
2. Not rehearsing a behavioural question
3. Not doing a Mock interview
4. Trying to memorize solutions
5. Not solving problems outloud
6. Rushing
7. Sloppy Coding
8. Not testing
9. Fixing mistakes carelessly
10. Giving up

	Start-Ups	Microsoft, Google, Amazon, Facebook, e.t.c.	Non-Tech Companies (including banks)
Men	Khakis, slacks, or nice jeans. Polo shirt or dress shirt.	Khakis, slacks, or nice jeans. Polo shirt or dress shirt.	Suit, no tie. (Consider bringing a tie just in case.)

Interview Template

Intro: "Hi, my name is Gabriel Lopes Carvalho, a Brazilian student at Temple University in Philadelphia. I'm a Computer Science Sophomore with a GPA of 3.64, named 2 times to the deans list. I want to build a career in Artificial Inteligence but I want to explore other positions whithin the industry"

Company culture Fit:

Questions:

Technical: How much of the day you spend coding

General: OPT/CPT/Visa problems/What do you expect for the perfect candidate or employee/Dress code

Skills: python/sql/java/excel/google data studio

Strenghs: teamplayer, creative, open minded

Weaknesses: insecure(studying abroad to overcome it), procrastination(found out starting the morning with an easy task boosts my performance)

S.A.R Questions(Situation, Action, Result)

Past experience

Logicalis - me and a team of interns had the task of optimizing the cloud assessment service of the company

	Project 1	Project 2	Project 3
Most challenging			
What I learned			
Most interesting			
Hardest bug			
Enjoyed the most			
Conflicts with teammates			

Technical interview

Ask questions to solve ambiguity -

Design a algorithm

time and space complexity

what happens with a lot of data

does the design cause other issues

what if they give you specific data

Write pseudo code first

Test

General Cases: what is expected

User input error: negative value, wrong data type

Extreme Cases: null, 0 div, empty list, maximun and minimun

Here's a list of the absolute must-have knowledge:

Data Structures	Algorithms	Concepts
Linked Lists	Breadth First Search	Bit Manipulation
Binary Trees	Depth First Search	Singleton Design Pattern
Tries	Binary Search	Factory Design Pattern
Stacks	Merge Sort	Memory (Stack vs Heap)
Queues	Quick Sort	Recursion
Vectors / ArrayLists	Tree Insert / Find / etc	Big-O Time
Hash Tables		

Algorithm Aproaches

Exemplify

Start with an example and draw a picture

Example: angle of a clock made by the hour and the minute stamps

Write out specific examples of the problem, and see if you can figure out a general rule.

Example: Given a time, calculate the angle between the hour and minute hands

Start with an example like 3:27 We can draw a picture of a clock by selecting where the 3 hour hand is and where the 27 minute hand is.

$$360 * m / 60 \quad M$$

360*(hour%12)/12 +360*(minutes/60)*(1/12) hour

(hour angle – minute nagle) %360

Pattern Matching

Match what problems the algorithm is similar to

Example: A sorted array has been rotated so that the elements might appear in the order 3 4 5 6 7 1 2 How would you find the minimum element?

Similar Problems:
- ➢ Find the minimum element in an array
- ➢ Find a particular element in an array (eg, binary search)

Simplifly and Generalize

change the constraint(data type, size, etc) to simplify the problem.
Once the base case is solved, generalize the proble again

Base Case and Build

solve the algorith for the base case, then keep adjusting for more complex situations

Example: Design an algorithm to print all permutations of a string. For simplicity, assume all characters are unique.

```
Test String: abcdefg
   Case "a" --> {a}
   Case "ab" --> {ab, ba}
   Case "abc" --> ?
```

Data Structure Brainstorm

With a list of data structures in your mind, try to find the one that best matches with the solution you are trying to find

Example: Numbers are randomly generated and stored into an (expanding) array How would you keep track of the median?

Data Structure Brainstorm:
- Linked List?
- Array?
- Heap?