



Internship / Program Analysis

(Helen Zhao 02/08/2026)



This presentation covers:

- Key findings and data trends, including supporting visuals and assumptions.
- Root cause analysis for underperformers in the internship program.
- Recommendations and next steps.



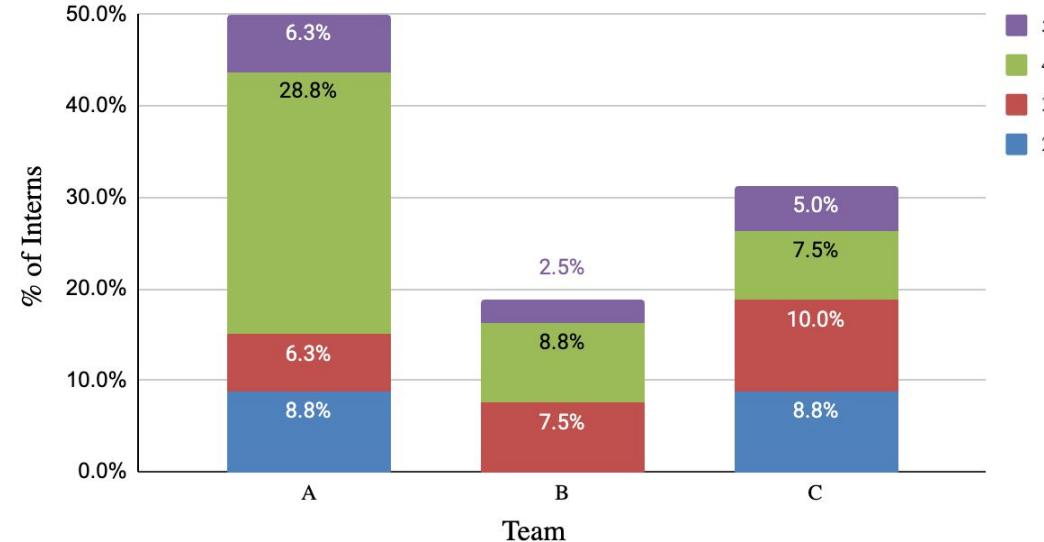
Distribution of Intern Performance Across Teams

The internship program encompasses **80** participants, who were allocated across **three** teams.

The data reveals the following:

- **Team A holds the largest proportion of interns**, representing 50% of the total, followed by Team C with approximately 31% and Team B with about 19%.
- **Final Performance Rating 4 is the most frequent outcome**, constituting 45% of all evaluations, significantly surpassing Rating 5 (13.8%).
- **Team A contributes the greatest share of top performers**, accounting for nearly half of all Rating 5 results, whereas Team B contributes the smallest portion.

Intern Performance Rating Distribution by Team



Team	Final Intern Performance Rating	Number of Interns	% of Interns
A	2	7	8.8%
A	3	5	6.3%
A	4	23	28.8%
A	5	5	6.3%
B	3	6	7.5%
B	4	7	8.8%
B	5	2	2.5%
C	2	7	8.8%
C	3	8	10.0%
C	4	6	7.5%
C	5	4	5.0%

Team Structure

- **Team A and Team B** are **Python-dominant** and focus on software development.
- **Team C** is structurally different, **C++ focused**, and primarily handles **infrastructure**.
- Multi-language interns are assigned to Team C.

Table 1 : Team Structure

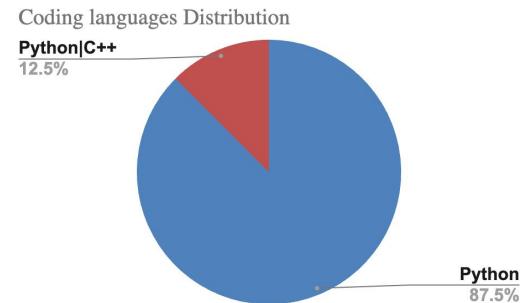
Team Name	Primary Language	Title	Number of Employees
A	Python	SWE - ChatGPT	43
A	Python / C++	SWE - ChatGPT	7
B	Python	SWE - Integrity	42
B	Python / C++	SWE - Integrity	3
C	C++	Infra Engineer	52
C	Python / C++	Infra Engineer	8

Intern Performance

- Interns are mostly fluent in Python.
- In **Team C**,
 - Interns with **Python and C++** coding languages have the highest overall interview scores, team match scores, and final performance ratings.
 - Interns **without C++** experience have the lowest overall performance rating, interview score, and team match score on average.

Table 2 : Intern Performance by Programming Language and Team

Intern Fluent Coding Language	assigned_team	avg match_score	avg final_rating	Average Interview Rating
Python	A	3.6	3.7	3.7
Python	B	4.0	3.7	3.9
Python	C	2.2	2.5	3.4
Python C++	C	4.7	4.4	4.7



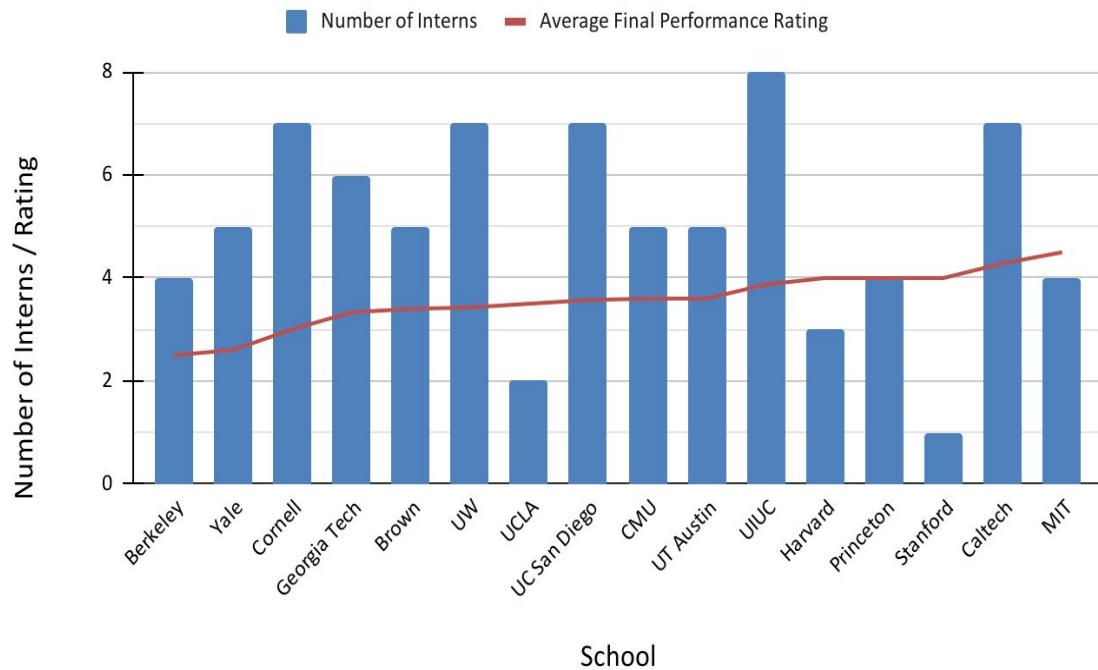
Where Our Best Intern Talent Comes From

Our interns were sourced from **16** different universities.

The data reveals the following:

- Top performance (average rating greater than 4) is observed among interns from **Harvard, Princeton, Stanford, Caltech, and MIT**. These institutions show the highest performance, despite having relatively small intern counts.
- Overall, **UIUC** demonstrates the best balance, having the largest intern population while maintaining solid performance.
- **Only few institutions manage to successfully balance both quality and scale.**

Number of Interns and Average Final Rating by School



School	Number of Interns	Average Final Performance Rating
Berkeley	4	2.5
Yale	5	2.6
Cornell	7	3.0
Georgia Tech	6	3.3
Brown	5	3.4
UW	7	3.4
UCLA	2	3.5
UC San Diego	7	3.6
CMU	5	3.6
UT Austin	5	3.6
UIUC	8	3.9
Harvard	3	4.0
Princeton	4	4.0
Stanford	1	4.0
Caltech	7	4.3
MIT	4	4.5

Systemic Skill–Team Mismatch in Python-Only Interns

- These interns were from **Team A** and **Team C**. **14 interns** received a low performance rating (final performance rating = 2).
- **They are all Python-Only interns.**
- They had a low team match score and an overall moderate to low post-interview rating.
- The interns from Team C specifically lacked C++ coding experience, **indicating a misalignment with infrastructure needs.**
- Despite some receiving interview ratings over 3.0, the final rating remained 2, **suggesting interview performance may have overestimated on-the-job success.**

candidate_id	school	fluent_languages	assigned_team	match_score	final_rating	Average_Interview_Rating
C008	Berkeley	Python	A	2	2	3.0
C010	Yale	Python	A	2	2	3.0
C013	CMU	Python	A	2	2	3.0
C015	Yale	Python	A	2	2	3.0
C016	CMU	Python	A	2	2	3.0
C030	Berkeley	Python	A	2	2	3.0
C036	Cornell	Python	A	2	2	3.0
C058	Georgia Tech	Python	C	2	2	3.5
C063	UC San Diego	Python	C	2	2	3.3
C068	Georgia Tech	Python	C	2	2	3.3
C070	Cornell	Python	C	2	2	3.5
C073	Berkeley	Python	C	2	2	3.5
C074	UT Austin	Python	C	2	2	3.0
C077	Brown	Python	C	2	2	3.5

Recommendations and Next Steps:

1. Enhance skill alignment : Avoid placing interns with Python-only programming skills on teams that primarily use non-Python languages or have high C++ requirements. If assigned to a C++ team, provide a one-week C++ or relevant coding training to facilitate their integration.
2. Establish core technical prerequisites: Require exposure to Python and C++ or equivalent systems-level programming as a baseline for assignment to highly complex teams.
3. Consider increasing the intern headcount from schools that consistently demonstrate a top average performance (> 4.0).
4. Implement a stricter role-based screening process, hiring interns based on the potential team's needs and required skills, and adding role-specific assessments to the interview process.
5. Introduce role-specific intern tracks to prevent misalignment between intern skill sets and team needs, reducing the risk of underperformance.
6. Evaluate intern performance using metrics that extend beyond the singular final performance rating.

Thank you!

