The Life of an SWE

<Hetav Pandya/>

Introduction

```
{
        "Name" : "Hetav Pandya",
        "Program" : "Computer Engineering",
        "Year" : [ "2T3", "PEY"],
        "Institution" : "University of Toronto",
        "Experience" : "Intel Corporation, Bell, General Motors"
}
```

Agenda

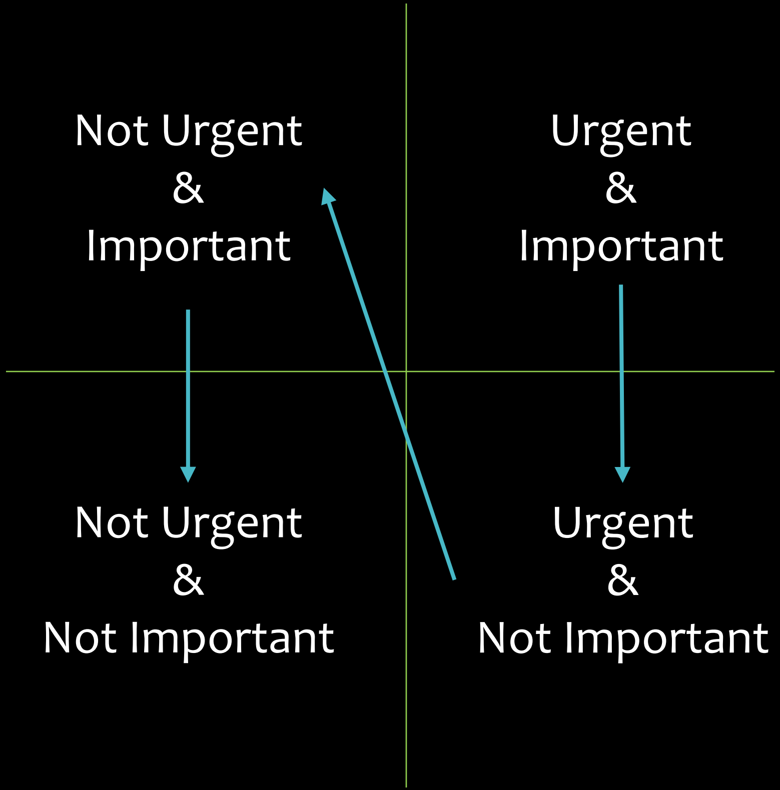
```
{
        “ItemsForToday" : [
                “Peek into a day of a SWE”,
                “Project planning in industry”,
                “How to establish a career in the tech industry?”,
                “How can University education help in the process?”]
}
```

Peek into the life of a Software Engineer

There are primarily two types of task you work on a day-to-day basis

Peek into the life of a Software Engineer

Small bugs that are reported by Quality Assurance team



Not Urgent
&
Important

Urgent
&
Important

Not Urgent
&
Not Important

Urgent
&
Not Important

Eisenhower matrix

Peek into the life of a Software Engineer

Major projects that have been assigned to you.

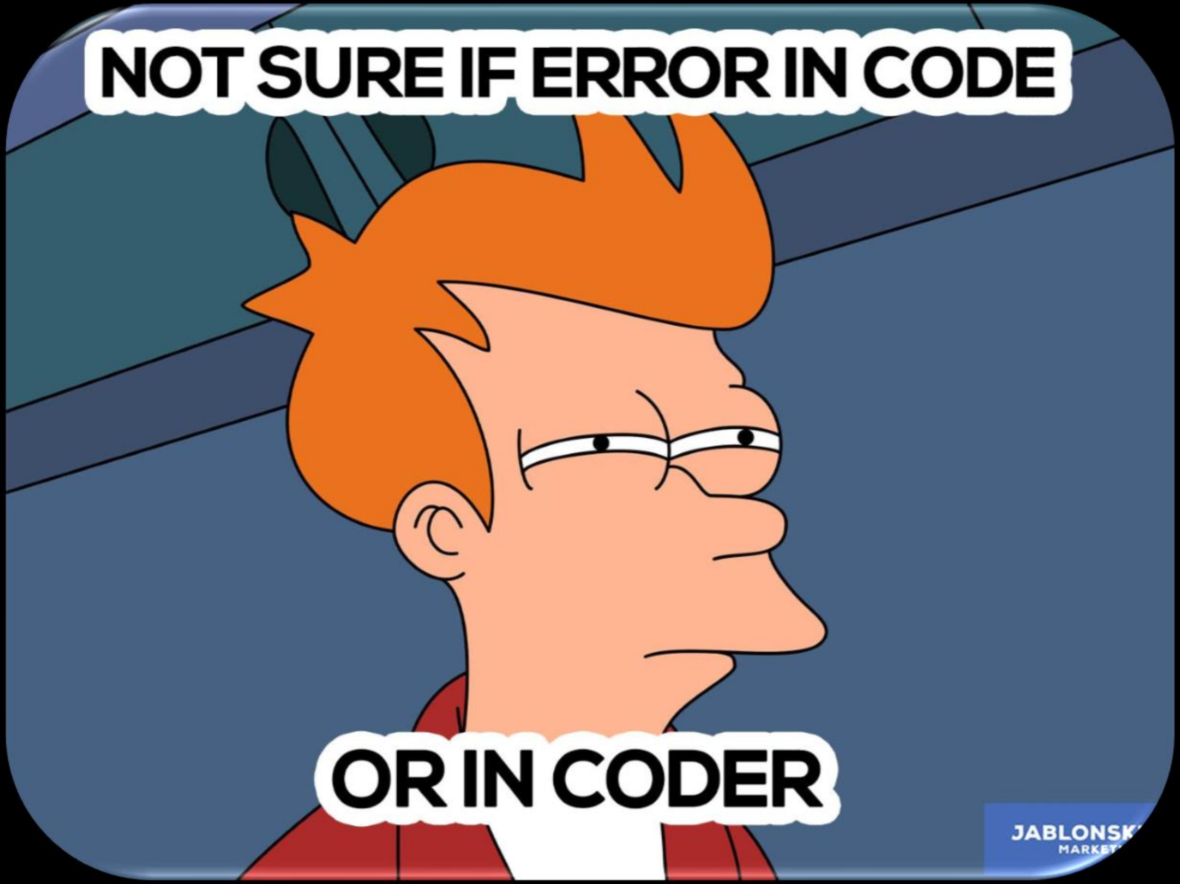
C.E.O
How is it going, Tony?
Developer
I have this headache that comes and goes
Project manager
Steve walks in
Hi, Tony
Developer
There it is again

66%

Of all technological projects fail partially or completely.

90%

Of all major technological projects fail.



Prepare a draft problem statement highlighting the existing issue.

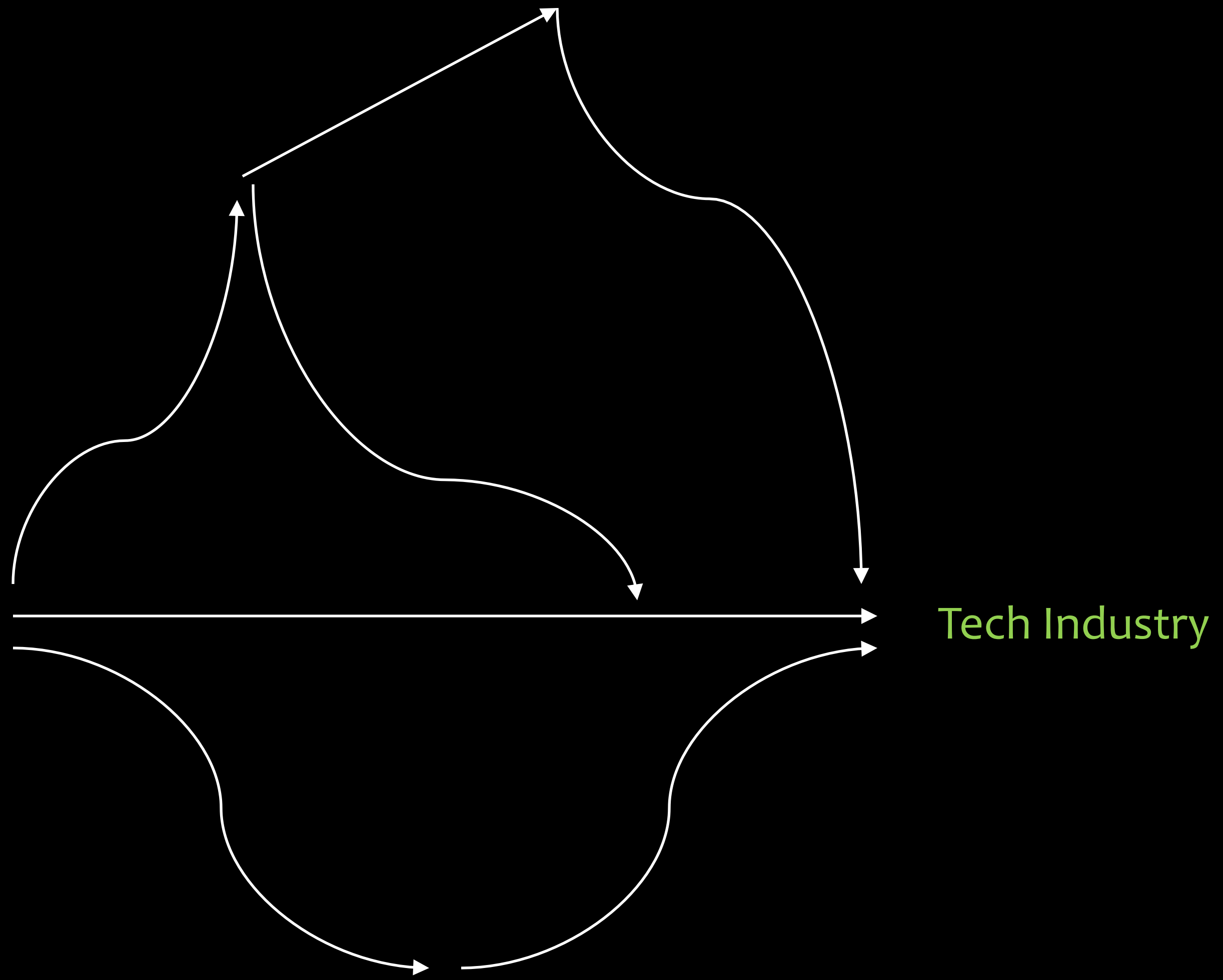
Have data to back up that the problem you are solving actually exists.

Identify the stakeholders or people who will be affected by the project.

Come up with different ideas to solve the problem.

Sit down with a more experienced colleague to go over your solution ideas.

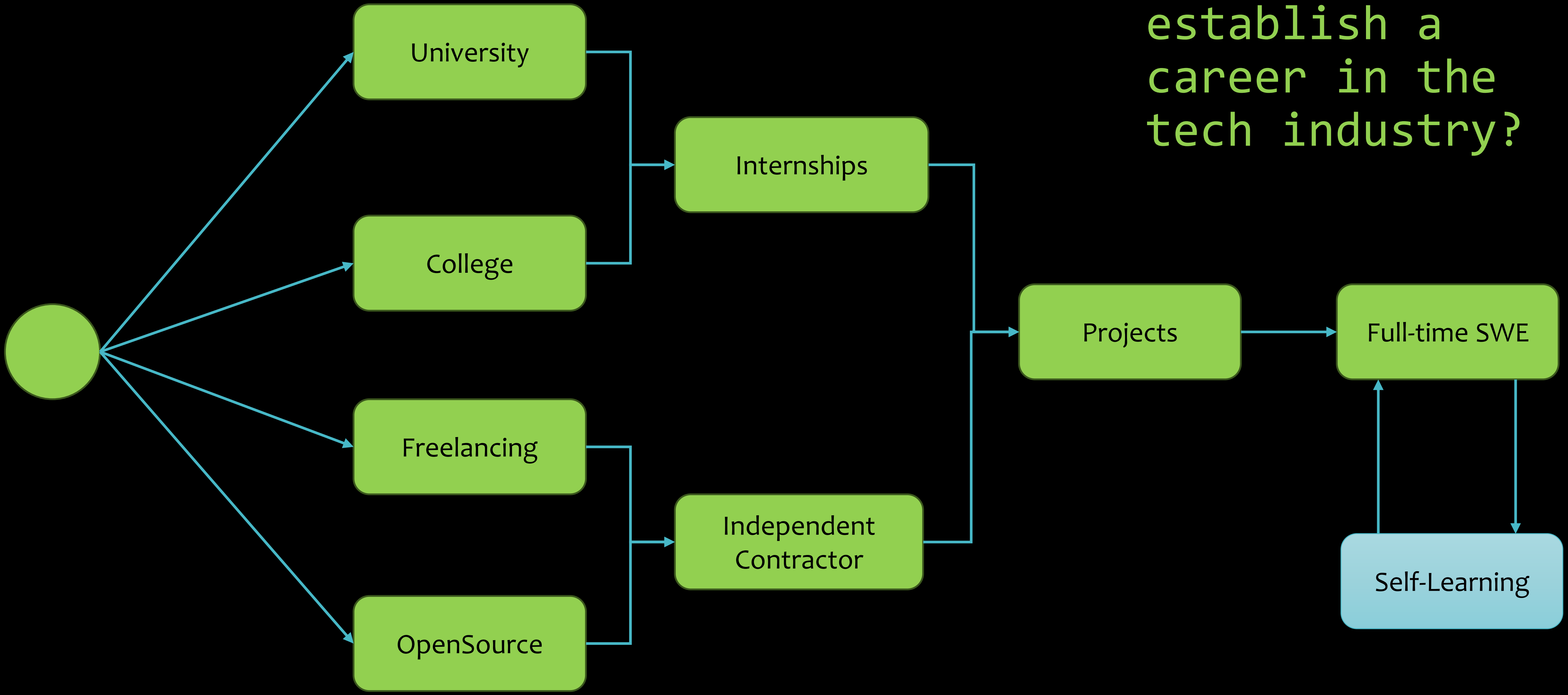
Set up a meeting with the stakeholders to identify the best solution.



How to establish a career in the tech industry?

You have a destination, now all you need is a direction.

How to establish a career in the tech industry?



University

College

Freelancing

OpenSource

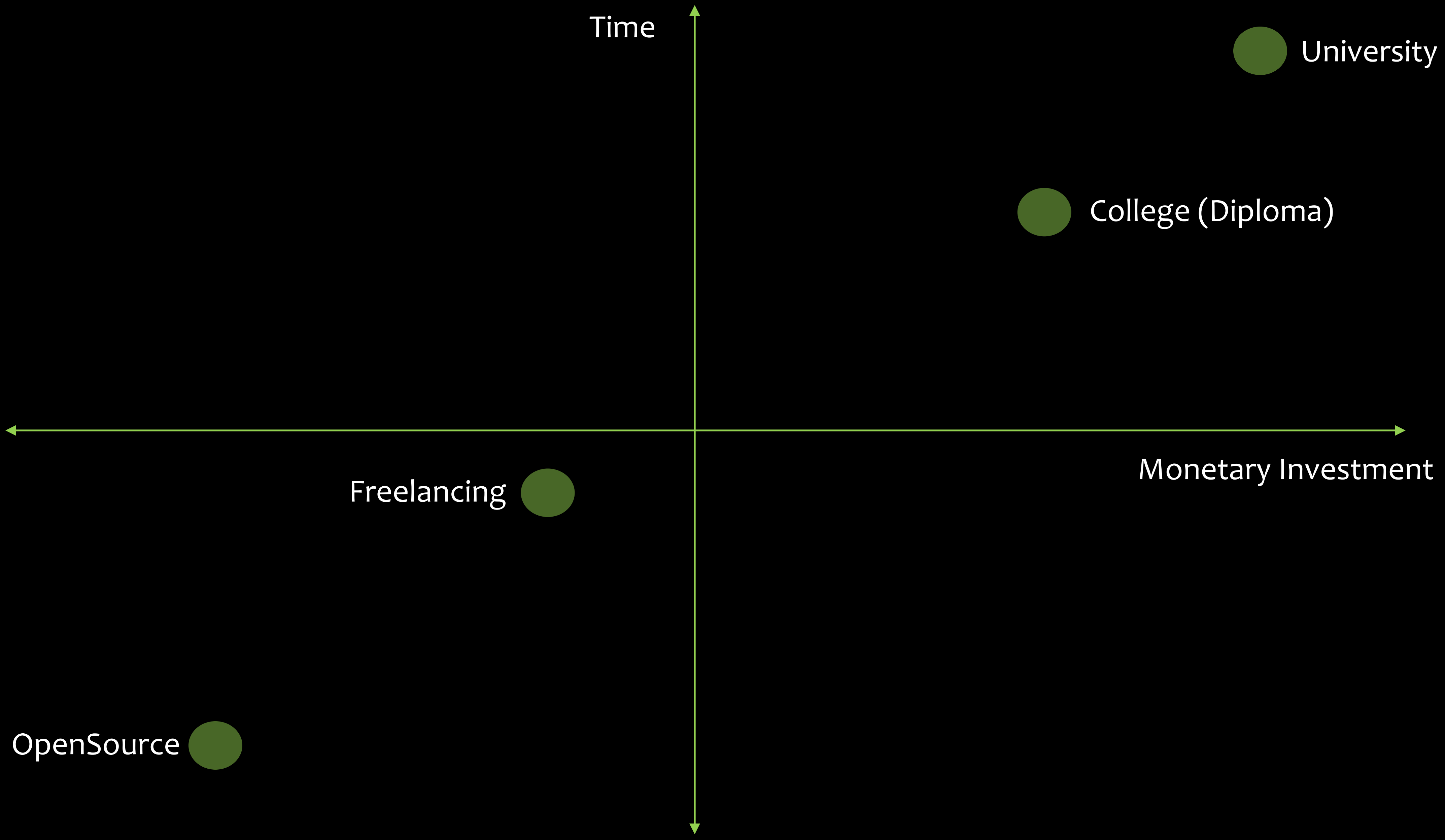
Internships

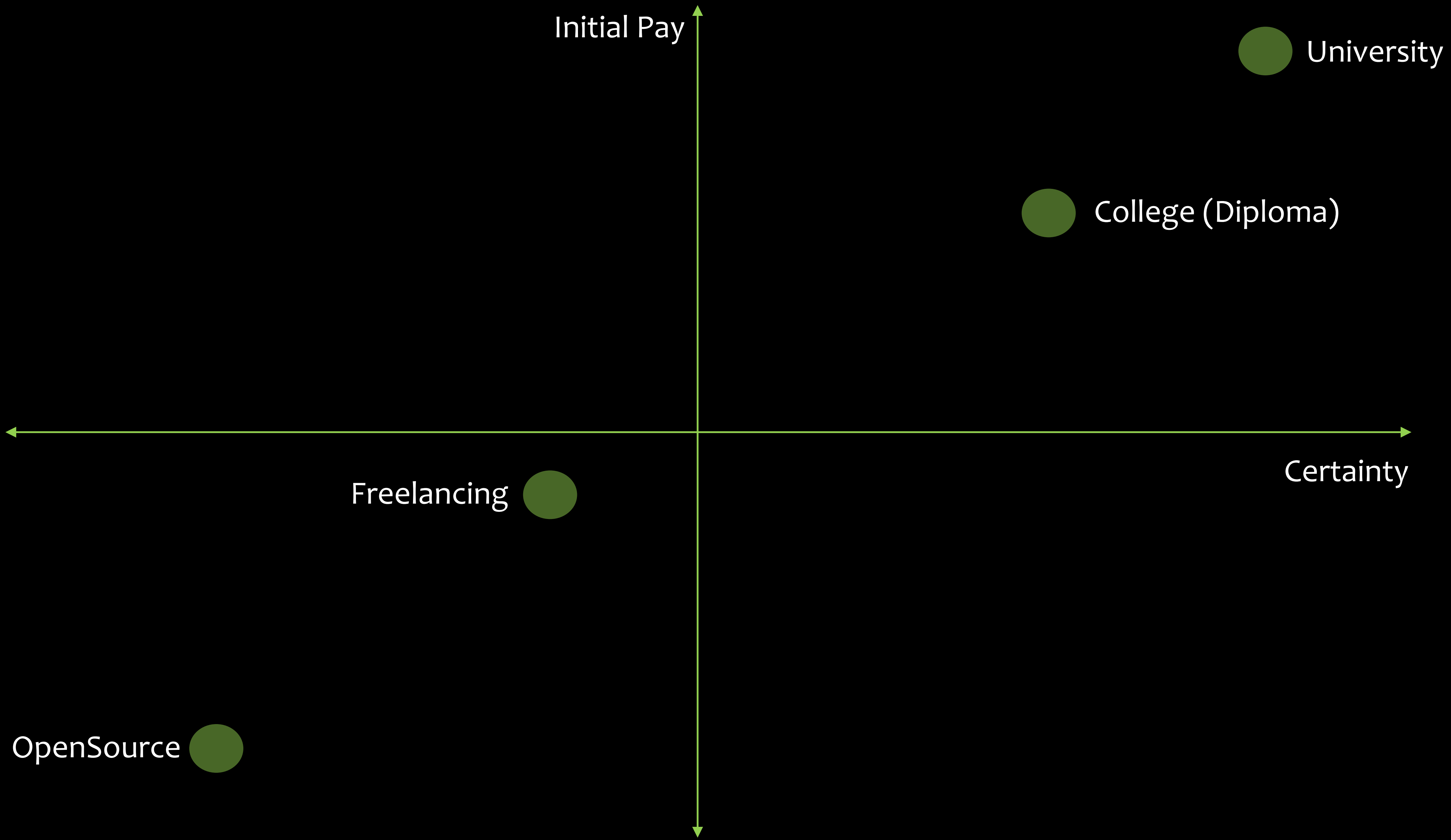
Independent Contractor

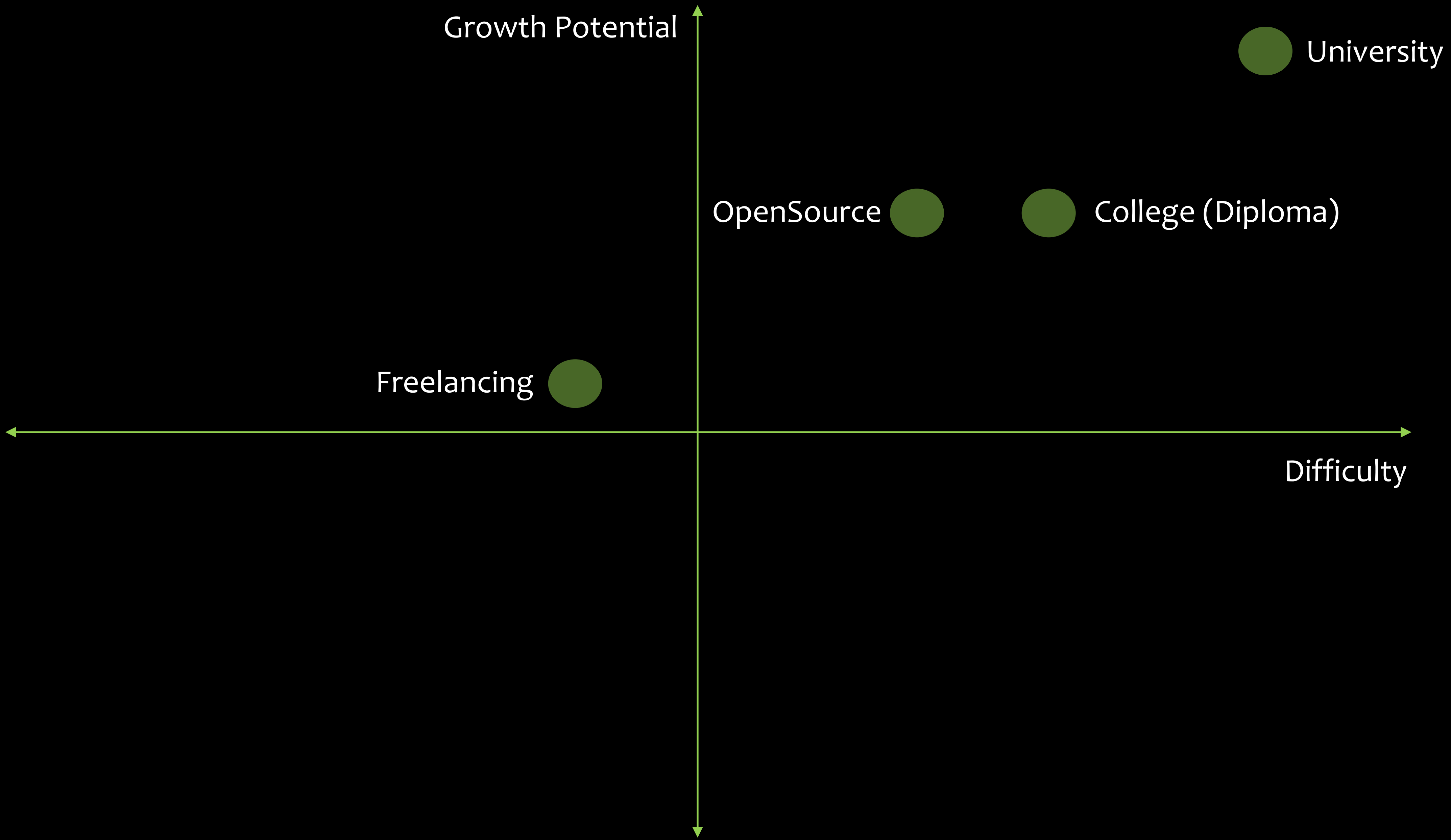
Projects

Full-time SWE

Self-Learning


Time
University
College (Diploma)
Monetary Investment
Freelancing
OpenSource


Initial Pay
University
College (Diploma)
Certainty
Freelancing
OpenSource


Growth Potential
University
OpenSource
College (Diploma)
Freelancing
Difficulty

GitLab

fiverr®

Take time to talk to people who have chosen different routes.

upwork

UNIVERSITY OF TORONTO

Seneca

The best route if a hybrid one: Formal + Informal education

How can University and College help in your journey?

Gives you a structured education plan.

How can University and College help in your journey?

It’s a treasure trove of opportunities: Good Professors, Great Research, Good Internship Programs

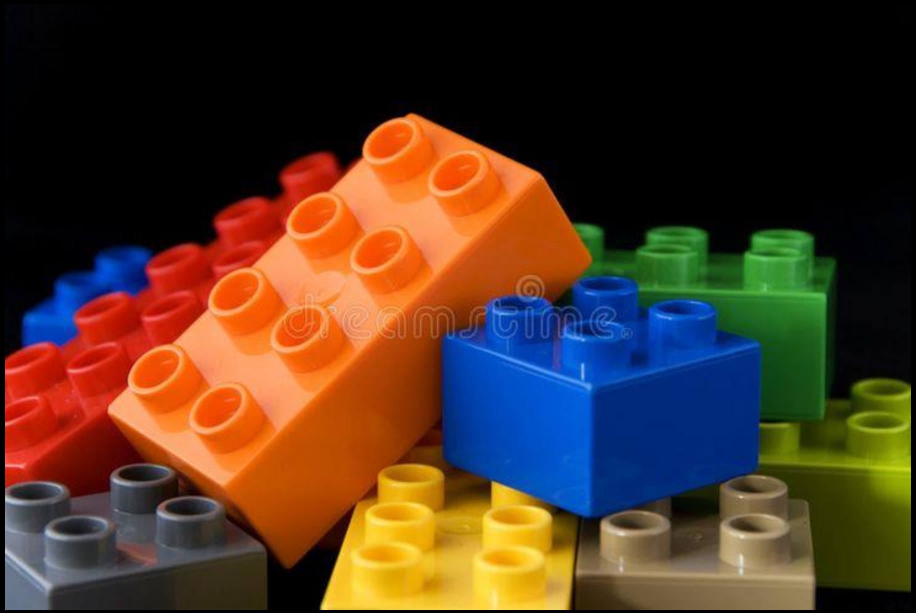
How can University and College help in your journey?

Specialized student clubs - Machine Learning, PCB chip design, Robotics, Guitar etc.

How can University and College help in your journey?

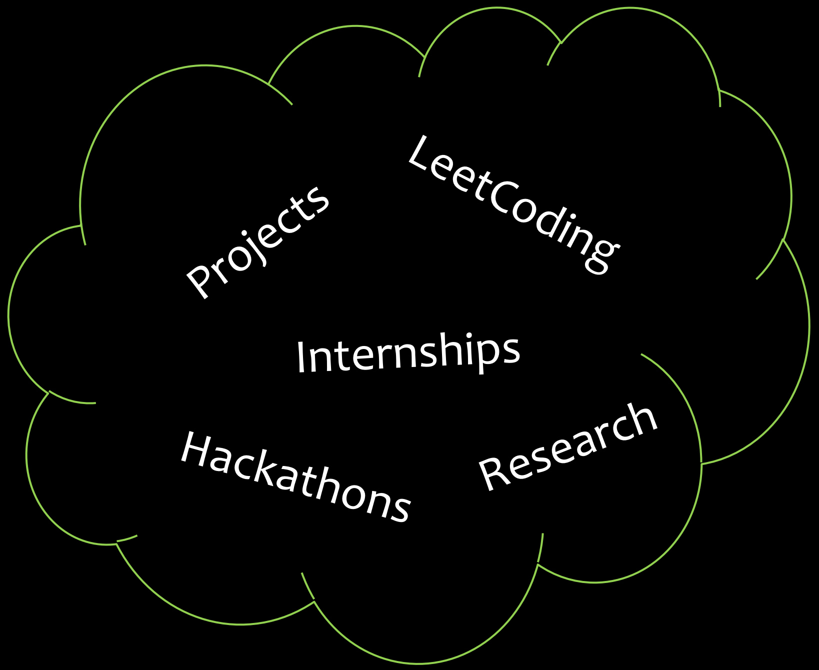
Combining different majors:

- Biotechnology with CS
- Climate specialization with CS
- Cognitive science and CS
- Math and CS
- Finance and CS



How can University help in your journey?

You still have to go beyond your courses…



Projects

LeetCoding

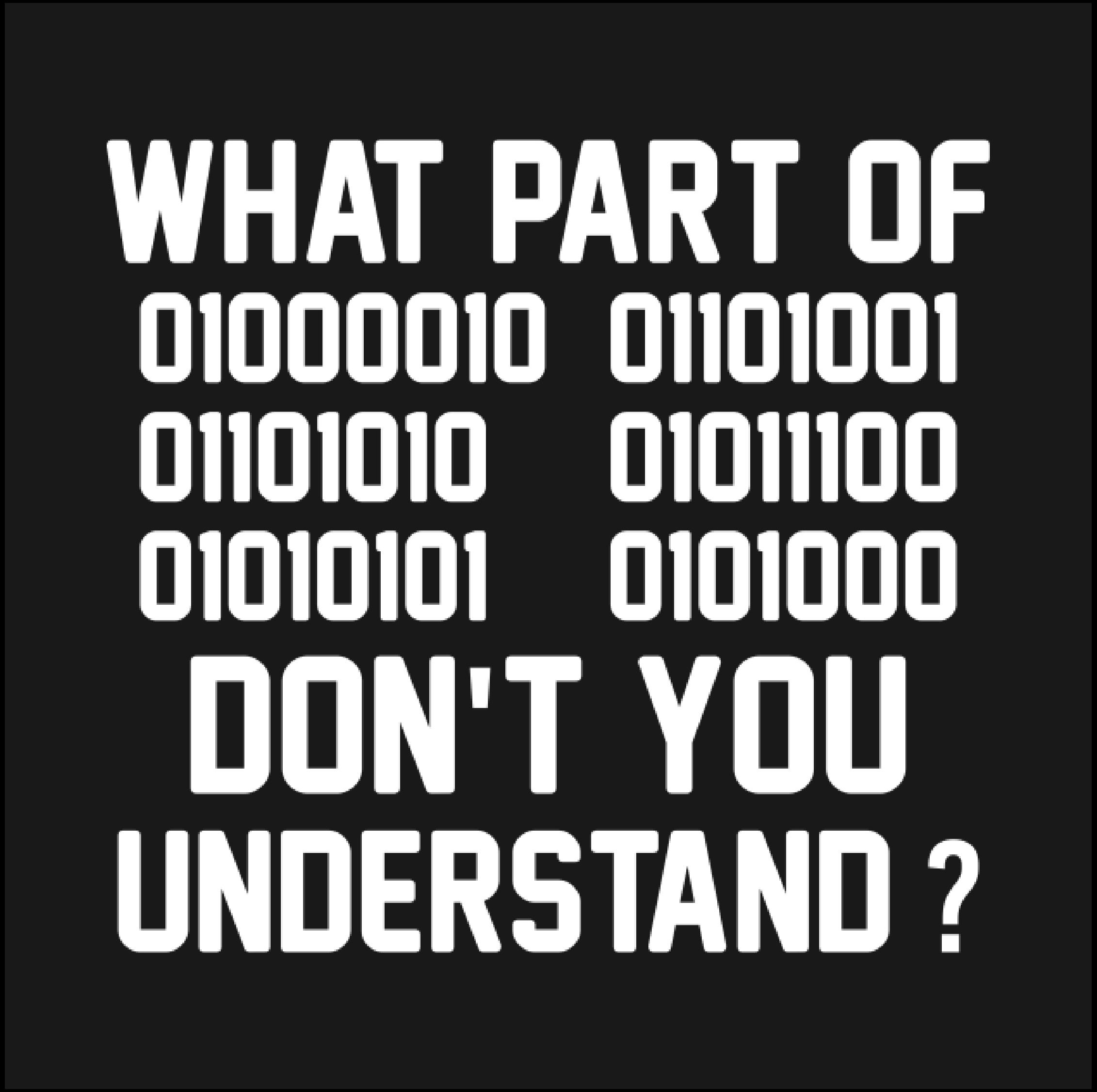
Internships

Hackathons

Research

Thank you!

(If time permits)



WHAT PART OF

01000010 01101001
01101010 01011100
01010101 01010000

DON'T YOU UNDERSTAND ?

How to write maintainable code?

The subtle art of not messing it up.

Learn what unit tests are and how different languages support it.

There are methods to automate testing using scripting languages like bash, Perl etc.

Python has a module called unittest dedicated to testing.

Testcases before code.

- It makes you consider extreme scenarios of your use case at the very beginning.
- It provides a quantitative method to measure progress.

C++ has the standard template libraries. Give it a thorough read!

Python is known for its rich module support for variety of tasks ranging from Finance to AI.

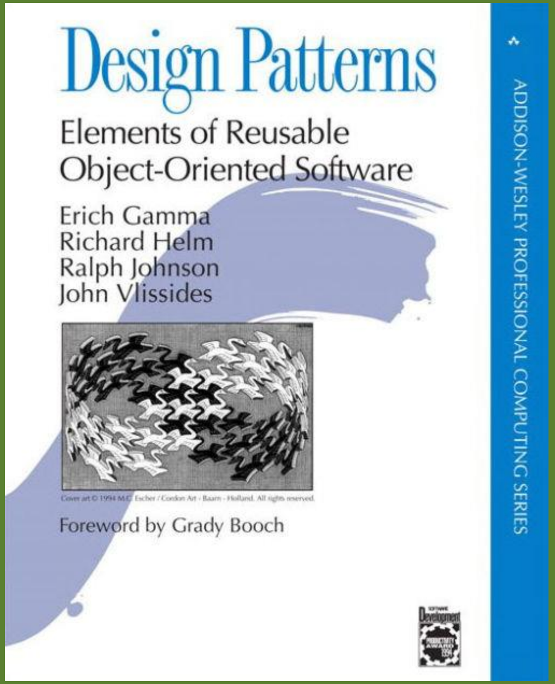
Do not reinvent the wheel.

- Consider using pre-created functionalities.
- It saves time and helps build secure software.

Learn how and where to use design patterns.

- Design patterns can help you write reusable code that is scalable in the future.
- These patterns have been tried and tested by thousands of developers.

To learn design patterns you can read the book: “Design Patterns: Elements of Reusable Object-Oriented Software”



Design Patterns

Elements of Reusable Object-Oriented Software

Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides

Cover art © 1994 M.C. Escher / Cordon Art - Baarn - Holland. All rights reserved.

Foreword by Grady Booch

ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES

Products like Doxygen helps you automatically generate documentation from comments in the code.

Products like GitBook help you manage multiple people working on documentation.

Document the code as you write it.

- Don’t make self-explanatory remarks. Instead explain what the code is intending to do on a high level.
- Stop manually documenting code, instead use some automation tools.

Learn how to generate flamegraphs for CPU usage. They help spot bottlenecks in code.

Tools like valgrind can help you check memory usage and identify any memory leaks in your code.

Learn how to check the performance of your code.

- Learn to use performance tools that help you check CPU time of your code. It will help you build faster code.
- Learn to use tools to check the memory usage of your code.

Git is one of the most popular version control systems. It is highly advised to have an account on GitHub and play around with it.

Having a well polished GitHub profile will boost your chances in the recruitment process.

Learn to use version control tools.

- You must know how to efficiently use your company’s version control software.
- Knowing how to navigate your version control system can streamline your coding process.