**1. Introduction**

In this assignment we cover peer-to-peer file distribution where peers share files over a network based on availability without the use of a centralized server. It allows a client to act as a server and vice versa.

The three main parts of the project are; the tracker, the seeder and the leecher, where the tracker maintains a list of active peers, the seeder acts as a peer that’s willing to share the files it has, and the leecher is looking for a specific file.

For this assignment, we used python as it has the built-in socket module that allows you to create and interact with network sockets.

**2. Features of the Application**

* List the **implemented features** and provide a short explanation of why each was included.

Tracker:

Seeder:

Leecher:

* Mention optional features (**file integrity verification, parallel downloads, GUI**), if implemented.

**3. System Architecture & Design**

* Describe how the **Tracker, Seeder, and Leecher** interact.
* Explain how **data transfer and peer discovery** work.
* Include **system diagrams** (architecture overview, workflow).

**4. Protocol Specification**

* **Message formats** (header structure, fields, message types: command, data transfer, control).
* **Communication sequence rules** (how each component transitions between states).
* Include **sequence diagrams** showing interactions between:
  + Leecher ↔ Tracker (peer discovery via UDP).
  + Leecher ↔ Seeder (file transfer via TCP).
  + Seeder ↔ Tracker (registration via UDP).

**5. Implementation Details**

* Explain **socket programming** approach:
  + UDP for tracker communication.
  + TCP for file transfer.
* Describe **threading or multiprocessing** (if used for parallel downloads).
* File chunking mechanism (e.g., **512 KB per chunk, hashing for integrity**).
* Error handling and robustness strategies.

**6. Testing and Results**

* Describe **test scenarios** (e.g., multiple seeders and leechers, network failures).
* Performance analysis (download speeds, number of peers tested).
* Include **screenshots** showing:
  + Tracker managing peers.
  + Leecher downloading chunks.
  + Seeder providing chunks.

**7. Challenges and Solutions**

* Mention **technical difficulties** encountered (e.g., socket handling, parallel downloads).
* Explain how you solved them or optimized the system.

**8. Conclusion and Future Improvements**

* Summarize the key takeaways.
* Discuss possible **enhancements** (e.g., encryption, dynamic chunk sizes, full GUI).

**9. References (if any)**

* Cite any external resources used (networking guides, research papers, etc.).

Conscious leader:

Aware of your surroundings

Being sure to plan ahead

Keeping others informed

Being on alert for any possible problems

Everyone is involved

No one left behind

Being aware of the repercussions of your actions and decisions

I felt that the FYBY movement would allow me the opportunity to build and refine leadership skills while at the same time, explore a career that has recently caught my eye, project management/consulting. Everything from the support that I would receive to the experience that I would walk away with is something that I feel would greatly benefit me at this stage of my career. Being able to be a part of a movement such as this one would also allow me the opportunity to see youths lifting up/building other youths which is something that is not as common outside of school societal settings. Normally, in the real world, it is adults that lead and especially coming from a cultural perspective where youths are sometimes undermined. It would be truly inspiring to be part of a movement that puts so much value to