CMSC 417 Bit-Torrent Client Project Report
Group: word.exe
Group Member: Zeping He

1) List of supported features:

    a) Communicate with the tracker (with support for compact format):
        i) This client is capable to the first tracker in torrent file
        ii) Send status update to tracker within a tracker defined interval

    b) Download a file from other official clients
        i) Support both single and multi-message in a packet – that is when handshake and bitfield are received in a same packet
        ii) Downloading from multiple peers at the same time (with support from asynico thread)

    c) Display Current progress
        i) Can show both global status and inter-piece status

    d) Save progress!!!
        i) When client exit during the download progress, it will assume where it left and pick it up
        ii) By encode and decode each piece into a file (bencode library)

    e) Download from other replicas
        i) Yes, that also supported
        ii) Basically – multi thread:
            (1) Thread 1: client
            (2) Thread 2: n download connections
            (3) Thread n + 1: server (listen for peer connection)
            (4) Thread n + 2 to X: server connection

    f) Extra credit 1: rare first
        i) Center-Client have a global view of each bitfield from its connections
        ii) Bitfield are up to date by call back functions
        iii) Rare-first index assign to connection first

    g) Extra credit 2: End-Game:
        i) When it comes to 90% of progress, each connection can start download to same piece
        ii) If center received a piece it send cancel to other connections that current downloading the same piece

    h) Extra credit 3: optimistic unchoking

        i) Still a global view, use sleep to select connections

    i) Extra credit 4:
        i) One man group?

2) Design and implementation choices that you made
    a) Python
        i) It's simple
        ii) Internal data-structure is explict
    b) Python asynico library
        i) Support safe threading
        ii) Download speed is not that bad anyway (peek at 600kbs from 15 connections)

3) Problems that you encountered (and if/how you addressed them)
    a) Last piece has different length thank piece_length
        i) Do manual calculation

4) Known bugs in your implementation
    a) Python thread is still io-bound
    b) So it has nature limit of download speed and code complexness

5) Contributions made by each group member
    a) Zeping He – lead