

Socket Programming: FTP Client

จงเขียนโปรแกรมด้วยภาษา Python เพื่อพัฒนาโปรแกรมที่เลียนแบบคำสั่ง ftp ในระบบปฏิบัติการ Windows โดยใช้ socket library ในภาษา Python (ห้ามใช้ library สำหรับ FTP สำเร็จรูปอื่นใดนอกจากการใช้ socket library) โปรแกรมดังกล่าวจะต้องสามารถเลียนแบบลักษณะการทำงานและคำสั่งของโปรแกรม ftp ในระบบปฏิบัติการ Windows ต่อไปนี้

- โครงสร้างหลักของโปรแกรมสามารถทำงานแบบ Read-Evaluate-Print-Loop (REPL) และรับคำสั่งได้ (4 คะแนน)
- **ascii** (2 คะแนน)
- **binary** (2 คะแนน)
- **bye** (1 คะแนน)
- **cd** (2 คะแนน)
- **close** (1 คะแนน)
- **delete** (2 คะแนน)
- **disconnect** (1 คะแนน)
- **get** (2 คะแนน)
- **ls** (2 คะแนน)
- **open** (2 คะแนน)
- **put** (2 คะแนน)
- **pwd** (2 คะแนน)
- **quit** (1 คะแนน)
- **rename** (2 คะแนน)
- **user** (2 คะแนน)

โดยผู้เรียนสามารถศึกษาความหมายของคำสั่ง ftp แต่ละคำสั่งเพิ่มเติมได้จาก <https://ss64.com/nt/ftp.html>

การตรวจให้คะแนนโปรแกรมจะเป็นการใช้เครื่องมือตรวจโดยอัตโนมัติ จึงขอให้ผู้เรียนเคร่งครัดกับรูปแบบการรับคำสั่งและ format การแสดงผลให้เหมือนกับคำสั่ง ftp ต้นฉบับ โดยผู้เรียนสามารถติดตั้ง ftp server ที่เป็น open source software เช่น FileZilla (<https://filezilla-project.org/download.php?type=server>) เพื่อใช้ในการทดสอบโปรแกรมที่ตนเองพัฒนา

Submission

ในการส่งงาน ให้ตั้งชื่อไฟล์หลักที่ใช้ในรันโปรแกรมว่า **myftp.py** เท่านั้น และให้ผู้เรียนรวม **source files** ทุกไฟล์ที่ผู้เรียนเขียนขึ้นมาทั้งหมดเอาไว้ใน **zip file** และตั้งชื่อว่า **myftp.zip** โดยให้ **zip** เพียงชั้นเดียวเท่านั้น (หลังจาก **unzip** จะพบไฟล์ **myftp.py** เลย) จากนั้นให้นำส่งบน **Microsoft Teams** และหากรายละเอียดอื่นๆ เพิ่มเติมที่ผู้ทดสอบควรรู้ในการทดสอบหรือใช้งานโปรแกรมของผู้เรียน ให้เขียนรายละเอียดเอาไว้ในไฟล์ **readme.txt** ซึ่งต้องรวมอยู่ใน **myftp.zip**