|  |  |  |
| --- | --- | --- |
| Tested | Pros | Cons |
| Multithreading.Pool | - Able to create limited number of workers  - If no worker is available, job is queued  - Runs job once a worker is available | - Pool can only be used in the same function where the pool is created (i.e. cannot create an instance of the pool and shared across instances of the app) |
| concurrent.futures.ThreadPoolExecutor | - Able to create limited number of workers  - Able to create instance shared across multiple files  - If no worker is available, job is queued and runs once a worker is available | - Returns future, so returns a result even before the process executes completely |
| Multithreading.Queue |  | - Unable to control size of queue  - output overwrites each processes’ output |
|  |  |  |