CS 61A Structure and Interpretation of Computer Programs

Fall 2025 CONCURRENCY PRACTICE

1. (4.0 points) Downloads

The coroutine function `download` fetches a file with the given name. Its implementation is not shown.

```
async def download(filename: str):
    ...

class Downloader():
    def __init__(self, filenames):
        self.filenames = filenames
        self.current_index = 0

    async def download_file(self):
        if self.current_index < len(self.filenames):
            filename = self.filenames[self.current_index]
            await download(filename)
            self.current_index += 1

    async def download_all(self):
        r = range(self.current_index, len(self.filenames))
        await asyncio.gather(*[self.download_file() for _ in r])

d = Downloader(["a", "b", "c", "d", "e"])
asyncio.run(d.download_all())
```

(a) (3.0 pt) The code above initializes a `Downloader` with a list of 5 files `["a", "b", "c", "d", "e"]`. Which of the following statements are true? Select all that apply.

■ The function `download` will be called exactly 5 times.

□ The function `download` may be called more than 5 times.

□ The function `download` may be called fewer than 5 times.

□ The function `download` will be called on each file exactly once.

■ The function `download` may not be called on all of the files.

■ The function `download` may be called more than once on the same file.

□ The code above may error because the index `self.current_index` is out of range.

(b) (1.0 pt) Assume that the function `download` takes one second to run, and spends all of that time blocked waiting for the file to be downloaded. How many seconds does the code above take to run?

1 second