

After receiving blocks and combining them into pieces, you'll want to check the integrity of each piece by comparing its hash with the piece hash value found in the torrent file.

Here's how the tester will execute your program:

```
$ ./your_bittorrent.sh download_piece -o /tmp/test-piece-0 sample.torrent 0
```

and here's the output it expects:

```
Piece 0 downloaded to /tmp/test-piece-0.
```

Optional: To improve download speeds, you can consider pipelining your requests. [BitTorrent Economics Paper](#) recommends having 5 requests pending at once, to avoid a delay between blocks being sent.

[View Code Examples](#)[View Screenscasts](#)[Collapse ↑](#)

Hints

[Filter by Rust](#)[Write](#)[Preview](#)

Found an interesting resource? Share it with the community.

Ready to run tests...

[Show logs](#)