CS3210: Lazy Page Allocation

Agenda

- Understanding lazy page allocation
- A demo
  - Walk through relevant xv6 functions
- In class exercise:
  - Implementing basic lazy page allocation in xv6
- Quiz discussion
  - Time permitting: continue in office hours

Life without lazy page allocation

- Physical pages immediately allocated (malloc, etc.)
- In xv6, done this way by default
- Potential undesirables
  - Wasted memory (allocated but never used)
  - Performance?
- Example
  - Linux: Does not alloce physical pages until they are used

Lazy page allocation

- Q: What sbrk()?
- Q: What is lazy page allocation?

sbrk in Linux

```
void *sbrk(intptr_t increment);
```

- Increment program data space by `increment` bytes

```
$ strace -e brk ./brk 4
$ perf stat ./brk 4
```

Lazy page allocation, explained

- Do not allocate physical memory in advance
- Allow page fault to occur
- Allocate required pages

```
$ git clone git://tc.gtisc.gatech.edu/cs3210-pub
```

or

```
$ cd cs3210-pub
$ git pull
```

Notes

- Must checkout older revision of xv6 so patches will apply cleanly

```
$ git checkout -b tut6 50edfe14
```

Quiz Discussion

- Quiz on Tuesday, October 4th
- Open book, open notes, open laptop, *closed internet*
- Written exam: bring a pencil(s)!
- Quiz 1 will only cover lectures through Thursday, 9/29, Chapters 0-2, Appendix A/B, and Labs 1 and 2