

Game Optimization

For fun and for profit

mattdurgavich@gmail.com

Who am I?

- EN 2003
- 10 years professional experience
- Last 5 years in the games industry

Who am I?



SUB ATOMIC
STUDIOS

Agenda

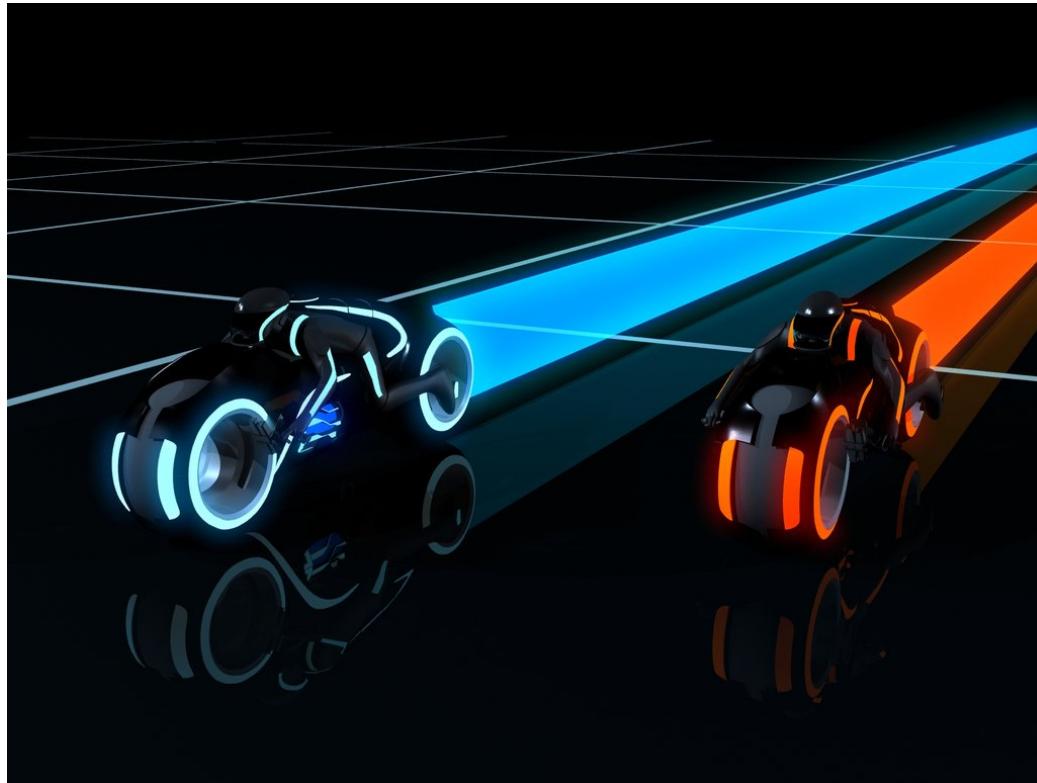
We will discuss:

- Program optimization
- Optimizing games
- Q&A

Optimization



Optimization: Perception?



Optimization: Reality



Optimization

"We should forget about small efficiencies, say about 97% of the time: premature optimization is the root of all evil"^[2]



An easy algorithm

- Measure
- Modify
- Test
- Repeat until done

An easy algorithm

- Measure

How?

What?

An easy algorithm

- Modify

What are you aiming for?

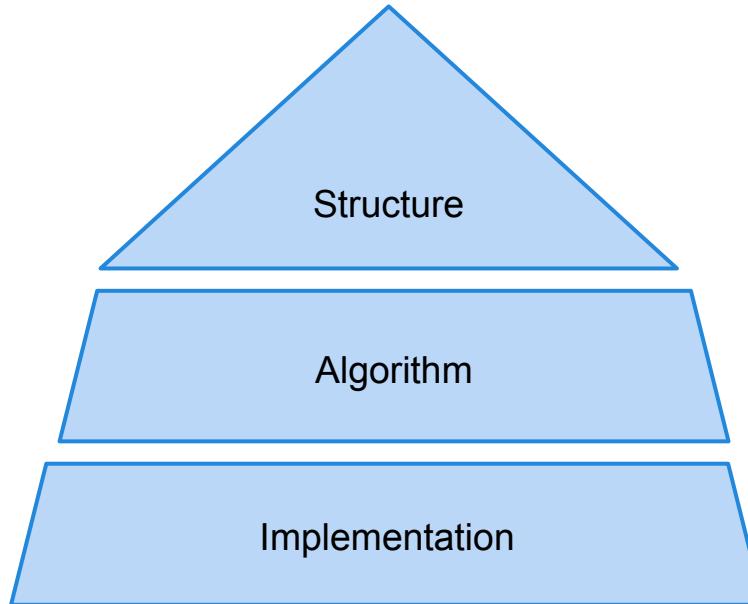
“You can always trade time for space”

An easy algorithm

- Test

Must preserve program integrity

Hierarchy of optimization



Structure



Algorithm



VS



Implementation

“Classic” operations:

- Loop unrolling
- Code inlining
- Constant folding
- Caching

Implementation

New hotness operations:

- Vectorizing
- Multicore support

The Big Idea

If your program is too slow, cache some results.

The Big Idea

If your program is too big, replace data with calculations.

Profilers

Two major classes:

- 1) Sampling
- 2) Instrumenting

Python profiling

Recall:

- Measure
- Modify
- Test

Python profiling

- Measure

```
python -m profile my.py
```

It's a sampling profiler

Using your results

Find the hot path

Using your results

Eliminate outliers

Game programmer tricks



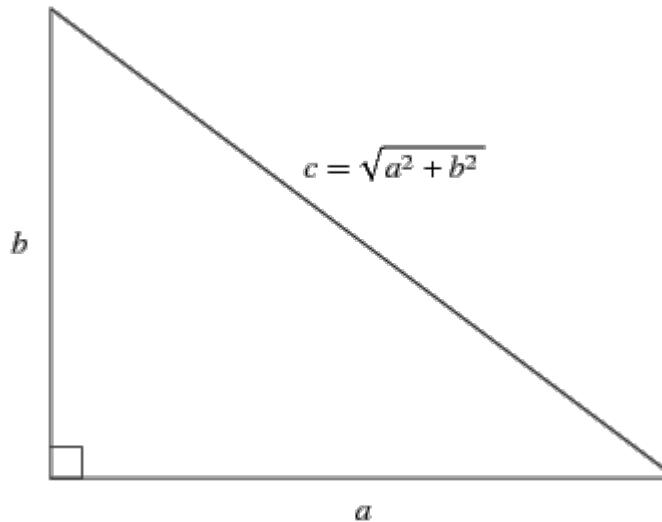
Know your platform

Use vendor libraries. Test, don't trust.

How far away is something?

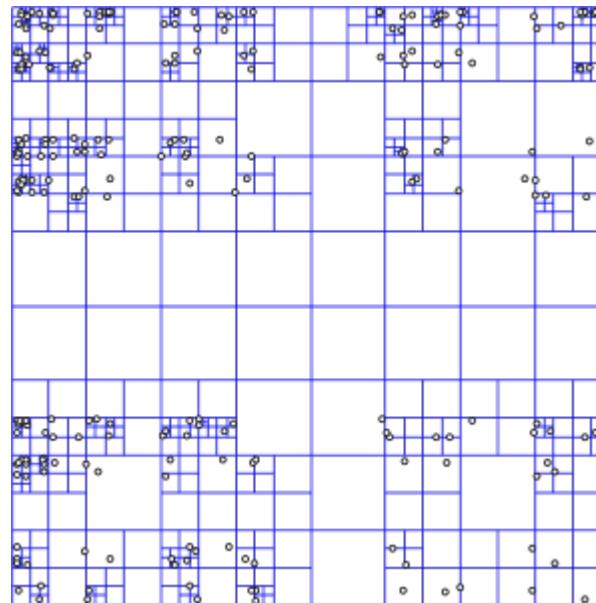
Using the Pythagorean theorem?

Compare **SQUARED** distances, since `math.sqrt()` is **VERY** expensive

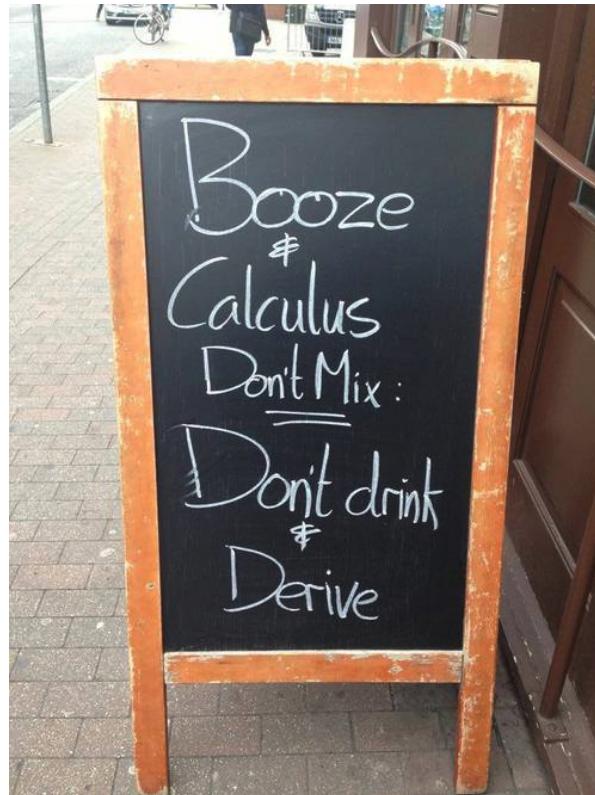


How far away is something?

Spatial partitioning:



My math is slow



My math is slow

Cache things!

- For a function, build a table mapping inputs to outputs
- Use linear interpolation if it makes sense

My math is slow

Fake it. Use a cheap(er) approximation.

Q&A

?