



John Carmack ✓ @ID_AA_Carmack

1h

Everyone knows that when you have similar code repeated several times that you should consolidate into a loop or function, but changing discrete variables into arrays and adding loops can have a small readability cost, so sometimes it can feel like a debatable choice for something small just repeated two or three times.

Usually, optimizing for readability is positive, but a requirement to keep two separate things aligned through future changes is an “invisible constraint” that is quite likely to cause problems eventually.

Many years ago, I would casually copy-paste-paste-paste-modify 4D vector statements, but now I almost never even do two related statements like that.

The hazard is when you combine two things that are close, but not exactly the same, and the strain induces complexity.

Apr 13, 2023 · 10:02 PM UTC

💬 53 ↻ 50 🗨️ 15 ❤️ 758



John Carmack ✓ @ID_AA_Carmack

1h

Sorting out some similar code for “fovea” and “peripheral” reminded me that I had made a mistake here.

💬 ↻ 🗨️ ❤️ 109



FM @ftuuky

1h

Replying to @ID_AA_Carmack

"Don't Repeat Yourself" versus "Repeat Once or Twice"

💬 2 ↻ 🗨️ ❤️ 12



John Carmack ✓ @ID_AA_Carmack

1h

But when things aren't exactly the same, do it three times before trying to infer an abstraction.

💬 3 ↻ 11 🗨️ 1 ❤️ 94



more replies



John Loeber 🚀 @johnloeber

1h

Replying to @ID_AA_Carmack

A lot of junior engineers are very very aggressive about DRY when in practice, a lot of code is similar enough to tempt to abstract, but too different to create a

💬 4 ↻ 2 🗨️ ❤️ 91



John Carmack ✓ @ID_AA_Carmack

1h

It is often still a good trade if an abstraction may induce some extra work where not strictly necessary, but a bad trade when you start introducing conditional behavior in your abstraction.

💬 2 ↻ 🗨️ ❤️ 87

⋮ more replies



Anthony Figueroa @anthfb
Replying to @ID_AA_Carmack

1h

abstract semantics must be the judge to define if two things are the "same"

💬 ↻ 🗨️ ❤️ 1



Olcan @olcan
Replying to @ID_AA_Carmack

1h

agree, though readability can be everything in early/experimental code, and it can actually be a good exercise to go through repetitive fixes and be forced to reconsider for future changes -- i think there is often a bias to refactor too early that is useful to acknowledge


💬 ↻ 🗨️ ❤️ 2




Benjamin De Kraker @BenjaminDEKR
Replying to @ID_AA_Carmack

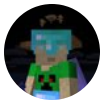
1h

Take this programming tweet from legend John Carmack and translate it to extremely basic caveman grunt language.

< 9/9 > 

 Make code not repeat. Put repeat code in loop or function. But loops and arrays can be hard to read. Readability good, but need to keep things aligned can be problem later. Me do copy-paste before, but now me no do. Bad when things almost same but not really, it make things complex.

💬 1 ↻ 3 🗨️ 1 ❤️ 76



MrHumpty_MC @MrHumpty_

1h

Replying to @ID_AA_Carmack

We have this issue with controls in our UI. Used in many different places and in a couple of them small variations. Makes testing new "features" added to the control for a new requirement a bit of slog.

We know it should be done the way we're doing it. But it can be painful sometimes.

1

more replies



Marcos Pereira @voxelbased

1h

Replying to @ID_AA_Carmack

It's all about sources of truth! If something changes, make sure there's only one place to update

2

Load more

