

C++ now

2025

# Program Correctness and its Limitations

David Sankel

# Program Correctness and its Limitations

David Sankel | Principal Scientist  
Software Technology Lab

**Adobe**

Image by Paula García for Adobe Creative Residency



# I don't know what program correctness is...

- We often speak of correctness in the context of programming, but I don't think we know what we're talking about
- I often see assumptions related to code correctness that don't hold up in practice
- I want a more rigorous foundation for correctness
- I want a better understanding of its limitations

*This is supposed to be an interactive journey, please engage. I need your help*

**Some people don't like this  
kind of talk...**

G.F.Y.

# **What does correct mean?**





Complete the bubbles

Test date \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_

State \_\_\_\_\_

Zip Code \_\_\_\_\_

Phone number \_\_\_\_\_

E-mail address \_\_\_\_\_

Test date \_\_\_\_\_

Math \_\_\_\_\_

Critical Reading \_\_\_\_\_

Writing \_\_\_\_\_

Math \_\_\_\_\_

Critical Reading \_\_\_\_\_

Writing \_\_\_\_\_

Math \_\_\_\_\_

Science \_\_\_\_\_

History \_\_\_\_\_

Language \_\_\_\_\_

Princeton Review®

© 1998 Princeton Review, Inc.

1250 Avenue of the Americas, New York, NY 10020

www.princetonreview.com

The Princeton Review®

© 1998 Princeton Review, Inc.

1250 Avenue of the Americas, New York, NY 10020

www.princetonreview.com

The Princeton Review®

© 1998 Princeton Review, Inc.

1250 Avenue of the Americas, New York, NY 10020

www.princetonreview.com

The Princeton Review®

© 1998 Princeton Review, Inc.

1250 Avenue of the Americas, New York, NY 10020

www.princetonreview.com

# Exam correctness

- Each response is said to either be “correct” or “incorrect”
- Correctness does not apply to the entirety of the filled out exam
- An assessment is made based on the proportion of correct responses.  
(sometimes weighted)

(Print) \_\_\_\_\_  
Number and Street \_\_\_\_\_  
City \_\_\_\_\_ State \_\_\_\_\_ Zip Code \_\_\_\_\_ /

IMPORTANT: Please fill in these boxes exactly as shown on the back cover of your test book.

2. TEST FORM		3. TEST CODE		4. REGISTRATION NUMBER	
		111			
		0 A 0 0 0 0 0 0 0 0	1 B 1 1 1 1 1 1 1 1	2 C 2 2 2 2 2 2 2 2 2	3 D 3 3 3 3 3 3 3 3 3
		4 E 4 4 4 4 4 4 4 4 4	5 F 5 5 5 5 5 5 5 5 5	6 G 6 6 6 6 6 6 6 6 6	7 H 7 7 7 7 7 7 7 7 7
		8 I 8 8 8 8 8 8 8 8 8	9 J 9 9 9 9 9 9 9 9 9	10 K 10 10 10 10 10 10 10 10 10	11 L 11 11 11 11 11 11 11 11 11
		12 M 12 12 12 12 12 12 12 12 12	13 N 13 13 13 13 13 13 13 13 13	14 O 14 14 14 14 14 14 14 14 14	15 P 15 15 15 15 15 15 15 15 15
		16 Q 16 16 16 16 16 16 16 16 16	17 R 17 17 17 17 17 17 17 17 17	18 S 18 18 18 18 18 18 18 18 18	19 T 19 19 19 19 19 19 19 19 19
		20 U 20 20 20 20 20 20 20 20 20	21 V 21 21 21 21 21 21 21 21 21	22 W 22 22 22 22 22 22 22 22 22	23 X 23 23 23 23 23 23 23 23 23
		24 Y 24 24 24 24 24 24 24 24 24	25 Z 25 25 25 25 25 25 25 25 25	26 AA 26 26 26 26 26 26 26 26 26	27 BB 27 27 27 27 27 27 27 27 27
		28 CC 28 28 28 28 28 28 28 28 28	29 DD 29 29 29 29 29 29 29 29 29	30 EE 30 30 30 30 30 30 30 30 30	31 FF 31 31 31 31 31 31 31 31 31
		32 GG 32 32 32 32 32 32 32 32 32	33 HH 33 33 33 33 33 33 33 33 33	34 II 34 34 34 34 34 34 34 34 34 34	35 JJ 35 35 35 35 35 35 35 35 35 35
		36 KK 36 36 36 36 36 36 36 36 36 36	37 LL 37 37 37 37 37 37 37 37 37 37	38 MM 38 38 38 38 38 38 38 38 38 38	39 NN 39 39 39 39 39 39 39 39 39 39
		40 OO 40 40 40 40 40 40 40 40 40 40	41 PP 41 41 41 41 41 41 41 41 41 41	42 QQ 42 42 42 42 42 42 42 42 42 42	43 RR 43 43 43 43 43 43 43 43 43 43
		44 SS 44 44 44 44 44 44 44 44 44 44	45 TT 45 45 45 45 45 45 45 45 45 45	46 UU 46 46 46 46 46 46 46 46 46 46	47 VV 47 47 47 47 47 47 47 47 47 47
		48 WW 48 48 48 48 48 48 48 48 48 48	49 XX 49 49 49 49 49 49 49 49 49 49	50 YY 50 50 50 50 50 50 50 50 50 50	51 ZZ 51 51 51 51 51 51 51 51 51 51
		52 AA 52 52 52 52 52 52 52 52 52 52	53 BB 53 53 53 53 53 53 53 53 53 53	54 CC 54 54 54 54 54 54 54 54 54 54	55 DD 55 55 55 55 55 55 55 55 55 55
		56 EE 56 56 56 56 56 56 56 56 56 56	57 FF 57 57 57 57 57 57 57 57 57 57	58 GG 58 58 58 58 58 58 58 58 58 58	59 HH 59 59 59 59 59 59 59 59 59 59
		60 II 60 60 60 60 60 60 60 60 60 60	61 JJ 61 61 61 61 61 61 61 61 61 61	62 KK 62 62 62 62 62 62 62 62 62 62	63 LL 63 63 63 63 63 63 63 63 63 63
		64 MM 64 64 64 64 64 64 64 64 64 64	65 NN 65 65 65 65 65 65 65 65 65 65	66 OO 66 66 66 66 66 66 66 66 66 66	67 PP 67 67 67 67 67 67 67 67 67 67
		68 QQ 68 68 68 68 68 68 68 68 68 68	69 RR 69 69 69 69 69 69 69 69 69 69	70 SS 70 70 70 70 70 70 70 70 70 70	71 TT 71 71 71 71 71 71 71 71 71 71
		72 UU 72 72 72 72 72 72 72 72 72 72	73 VV 73 73 73 73 73 73 73 73 73 73	74 WW 74 74 74 74 74 74 74 74 74 74	75 XX 75 75 75 75 75 75 75 75 75 75
		76 YY 76 76 76 76 76 76 76 76 76 76	77 ZZ 77 77 77 77 77 77 77 77 77 77	78 AA 78 78 78 78 78 78 78 78 78 78	79 BB 79 79 79 79 79 79 79 79 79 79
		80 CC 80 80 80 80 80 80 80 80 80 80	81 DD 81 81 81 81 81 81 81 81 81 81	82 EE 82 82 82 82 82 82 82 82 82 82	83 FF 83 83 83 83 83 83 83 83 83 83
		84 GG 84 84 84 84 84 84 84 84 84 84	85 HH 85 85 85 85 85 85 85 85 85 85	86 II 86 86 86 86 86 86 86 86 86 86	87 JJ 87 87 87 87 87 87 87 87 87 87
		88 KK 88 88 88 88 88 88 88 88 88 88	89 LL 89 89 89 89 89 89 89 89 89 89	90 MM 90 90 90 90 90 90 90 90 90 90	91 NN 91 91 91 91 91 91 91 91 91 91
		92 OO 92 92 92 92 92 92 92 92 92 92	93 PP 93 93 93 93 93 93 93 93 93 93	94 QQ 94 94 94 94 94 94 94 94 94 94	95 RR 95 95 95 95 95 95 95 95 95 95
		96 SS 96 96 96 96 96 96 96 96 96 96	97 TT 97 97 97 97 97 97 97 97 97 97	98 UU 98 98 98 98 98 98 98 98 98 98	99 VV 99 99 99 99 99 99 99 99 99 99
		100 WW 100 100 100 100 100 100 100 100 100 100	101 XX 101 101 101 101 101 101 101 101 101 101	102 YY 102 102 102 102 102 102 102 102 102 102	103 ZZ 103 103 103 103 103 103 103 103 103 103
		104 AA 104 104 104 104 104 104 104 104 104 104	105 BB 105 105 105 105 105 105 105 105 105 105	106 CC 106 106 106 106 106 106 106 106 106 106	107 DD 107 107 107 107 107 107 107 107 107 107
		108 EE 108 108 108 108 108 108 108 108 108 108	109 FF 109 109 109 109 109 109 109 109 109 109	110 GG 110 110 110 110 110 110 110 110 110 110	111 HH 111 111 111 111 111 111 111 111 111 111
		112 II 112 112 112 112 112 112 112 112 112 112	113 JJ 113 113 113 113 113 113 113 113 113 113	114 KK 114 114 114 114 114 114 114 114 114 114	115 LL 115 115 115 115 115 115 115 115 115 115
		116 MM 116 116 116 116 116 116 116 116 116 116	117 NN 117 117 117 117 117 117 117 117 117 117	118 OO 118 118 118 118 118 118 118 118 118 118	119 PP 119 119 119 119 119 119 119 119 119 119
		120 QQ 120 120 120 120 120 120 120 120 120 120	121 RR 121 121 121 121 121 121 121 121 121 121	122 SS 122 122 122 122 122 122 122 122 122 122	123 TT 123 123 123 123 123 123 123 123 123 123
		124 UU 124 124 124 124 124 124 124 124 124 124	125 VV 125 125 125 125 125 125 125 125 125 125	126 WW 126 126 126 126 126 126 126 126 126 126	127 XX 127 127 127 127 127 127 127 127 127 127
		128 YY 128 128 128 128 128 128 128 128 128 128	129 ZZ 129 129 129 129 129 129 129 129 129 129	130 AA 130 130 130 130 130 130 130 130 130 130	131 BB 131 131 131 131 131 131 131 131 131 131
		132 CC 132 132 132 132 132 132 132 132 132 132	133 DD 133 133 133 133 133 133 133 133 133 133	134 EE 134 134 134 134 134 134 134 134 134 134	135 FF 135 135 135 135 135 135 135 135 135 135
		136 GG 136 136 136 136 136 136 136 136 136 136	137 HH 137 137 137 137 137 137 137 137 137 137	138 II 138 138 138 138 138 138 138 138 138 138	139 JJ 139 139 139 139 139 139 139 139 139 139
		140 KK 140 140 140 140 140 140 140 140 140 140	141 LL 141 141 141 141 141 141 141 141 141 141	142 MM 142 142 142 142 142 142 142 142 142 142	143 NN 143 143 143 143 143 143 143 143 143 143
		144 OO 144 144 144 144 144 144 144 144 144 144	145 PP 145 145 145 145 145 145 145 145 145 145	146 QQ 146 146 146 146 146 146 146 146 146 146	147 RR 147 147 147 147 147 147 147 147 147 147
		148 SS 148 148 148 148 148 148 148 148 148 148	149 TT 149 149 149 149 149 149 149 149 149 149	150 UU 150 150 150 150 150 150 150 150 150 150	151 VV 151 151 151 151 151 151 151 151 151 151
		152 WW 152 152 152 152 152 152 152 152 152 152	153 XX 153 153 153 153 153 153 153 153 153 153	154 YY 154 154 154 154 154 154 154 154 154 154	155 ZZ 155 155 155 155 155 155 155 155 155 155
		156 AA 156 156 156 156 156 156 156 156 156 156	157 BB 157 157 157 157 157 157 157 157 157 157	158 CC 158 158 158 158 158 158 158 158 158 158	159 DD 159 159 159 159 159 159 159 159 159 159
		160 EE 160 160 160 160 160 160 160 160 160 160	161 FF 161 161 161 161 161 161 161 161 161 161	162 GG 162 162 162 162 162 162 162 162 162 162	163 HH 163 163 163 163 163 163 163 163 163 163
		164 GG 164 164 164 164 164 164 164 164 164 164	165 HH 165 165 165 165 165 165 165 165 165 165	166 II 166 166 166 166 166 166 166 166 166 166	167 JJ 167 167 167 167 167 167 167 167 167 167
		168 KK 168 168 168 168 168 168 168 168 168 168	169 LL 169 169 169 169 169 169 169 169 169 169	170 MM 170 170 170 170 170 170 170 170 170 170	171 NN 171 171 171 171 171 171 171 171 171 171
		172 OO 172 172 172 172 172 172 172 172 172 172	173 PP 173 173 173 173 173 173 173 173 173 173	174 QQ 174 174 174 174 174 174 174 174 174 174	175 RR 175 175 175 175 175 175 175 175 175 175
		176 SS 176 176 176 176 176 176 176 176 176 176	177 TT 177 177 177 177 177 177 177 177 177 177	178 UU 178 178 178 178 178 178 178 178 178 178	179 VV 179 179 179 179 179 179 179 179 179 179
		180 WW 180 180 180 180 180 180 180 180 180 180	181 XX 181 181 181 181 181 181 181 181 181 181	182 YY 182 182 182 182 182 182 182 182 182 182	183 ZZ 183 183 183 183 183 183 183 183 183 183
		184 AA 184 184 184 184 184 184 184 184 184 184	185 BB 185 185 185 185 185 185 185 185 185 185	186 CC 186 186 186 186 186 186 186 186 186 186	187 DD 187 187 187 187 187 187 187 187 187 187
		188 EE 188 188 188 188 188 188 188 188 188 188	189 FF 189 189 189 189 189 189 189 189 189 189	190 GG 190 190 190 190 190 190 190 190 190 190	191 HH 191 191 191 191 191 191 191 191 191 191
		192 GG 192 192 192 192 192 192 192 192 192 192	193 HH 193 193 193 193 193 193 193 193 193 193	194 II 194 194 194 194 194 194 194 194 194 194	195 JJ 195 195 195 195 195 195 195 195 195 195
		196 KK 196 196 196 196 196 196 196 196 196 196	197 LL 197 197 197 197 197 197 197 197 197 197	198 MM 198 198 198 198 198 198 198 198 198 198	199 NN 199 199 199 199 199 199 199 199 199 199
		200 OO 200 200 200 200 200 200 200 200 200 200	201 PP 201 201 201 201 201 201 201 201 201 201	202 QQ 202 202 202 202 202 202 202 202 202 202	203 RR 203 203 203 203 203 203 203 203 203 203
		204 SS 204 204 204 204 204 204 204 204 204 204	205 TT 205 205 205 205 205 205 205 205 205 205	206 UU 206 206 206 206 206 206 206 206 206 206	207 VV 207 207 207 207 207 207 207 207 207 207
		208 WW 208 208 208 208 208 208 208 208 208 208	209 XX 209 209 209 209 209 209 209 209 209 209	210 YY 210 210 210 210 210 210 210 210 210 210	211 ZZ 211 211 211 211 211 211 211 211 211 211
		212 AA 212 212 212 212 212 212 212 212 212 212	213 BB 213 213 213 213 213 213 213 213 213 213	214 CC 214 214 214 214 214 214 214 214 214 214	215 DD 215 215 215 215 215 215 215 215 215 215
		216 EE 216 216 216 216 216 216 216 216 216 216	217 FF 217 217 217 217 217 217 217 217 217 217	218 GG 218 218 218 218 218 218 218 218 218 218	219 HH 219 219 219 219 219 219 219 219 219 219
		220 GG 220 220 220 220 220 220 220 220 220 220	221 HH 221 221 221 221 221 221 221 221 221 221	222 II 222 222 222 222 222 222 222 222 222 222	223 JJ 223 223 223 223 223 223 223 223 223 223
		224 KK 224 224 224 224 224 224 224 224 224 224	225 LL 225 225 225 225 225 225 225 225 225 225	226 MM 226 226 226 226 226 226 226 226 226 226	227 NN 227 227 227 227 227 227 227 227 227 227
		228 OO 228 228 228 228 228 228 228 228 228 228	229 PP 229 229 229 229 229 229 229 229 229 229	230 QQ 230 230 230 230 230 230 230 230 230 230	231 RR 231 231 231 231 231 231 231 231 231 231
		232 SS 232 232 232 232 232 232 232 232 232 232	233 TT 233 233 233 233 233 233 233 233 233 233		

**What makes one answer  
correct and the others  
incorrect?**





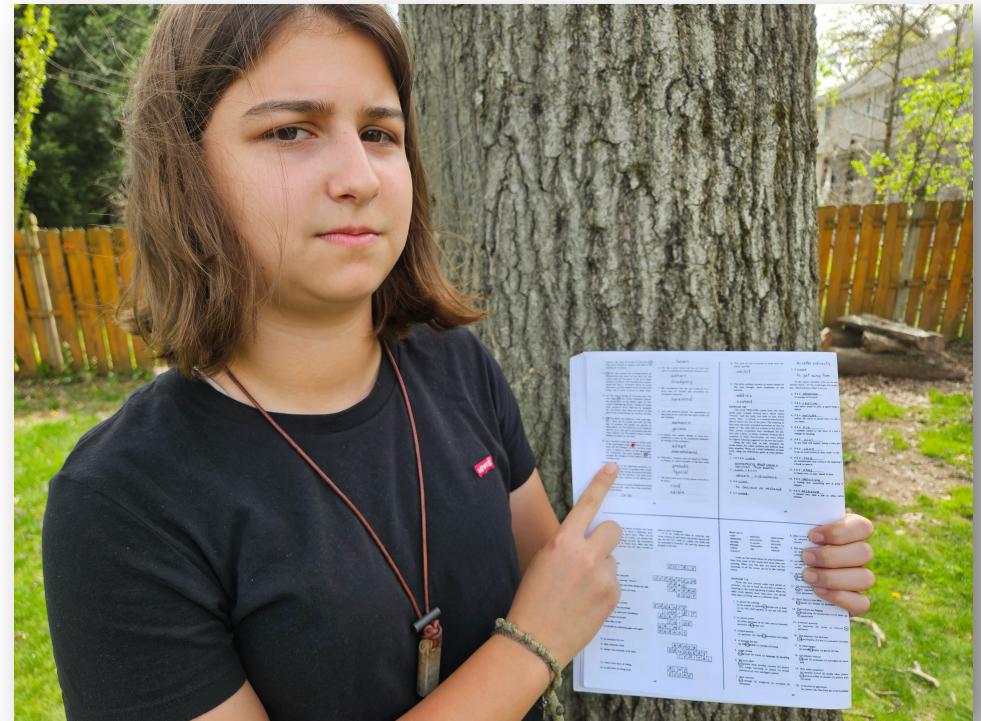
16. (a) I wouldn't mind the *drudgery* of the work if they paid higher wages. (b) Would you clear away all this *drudgery* so that I'll have some room in which to work? (c) He agreed that the housework was pure *drudgery*. (d) He escaped the *drudgery* of his father's farm by running away.

# How is correctness determined?

It isn't determined by "the answer key".

*There's got to be something more to it*

*What?*

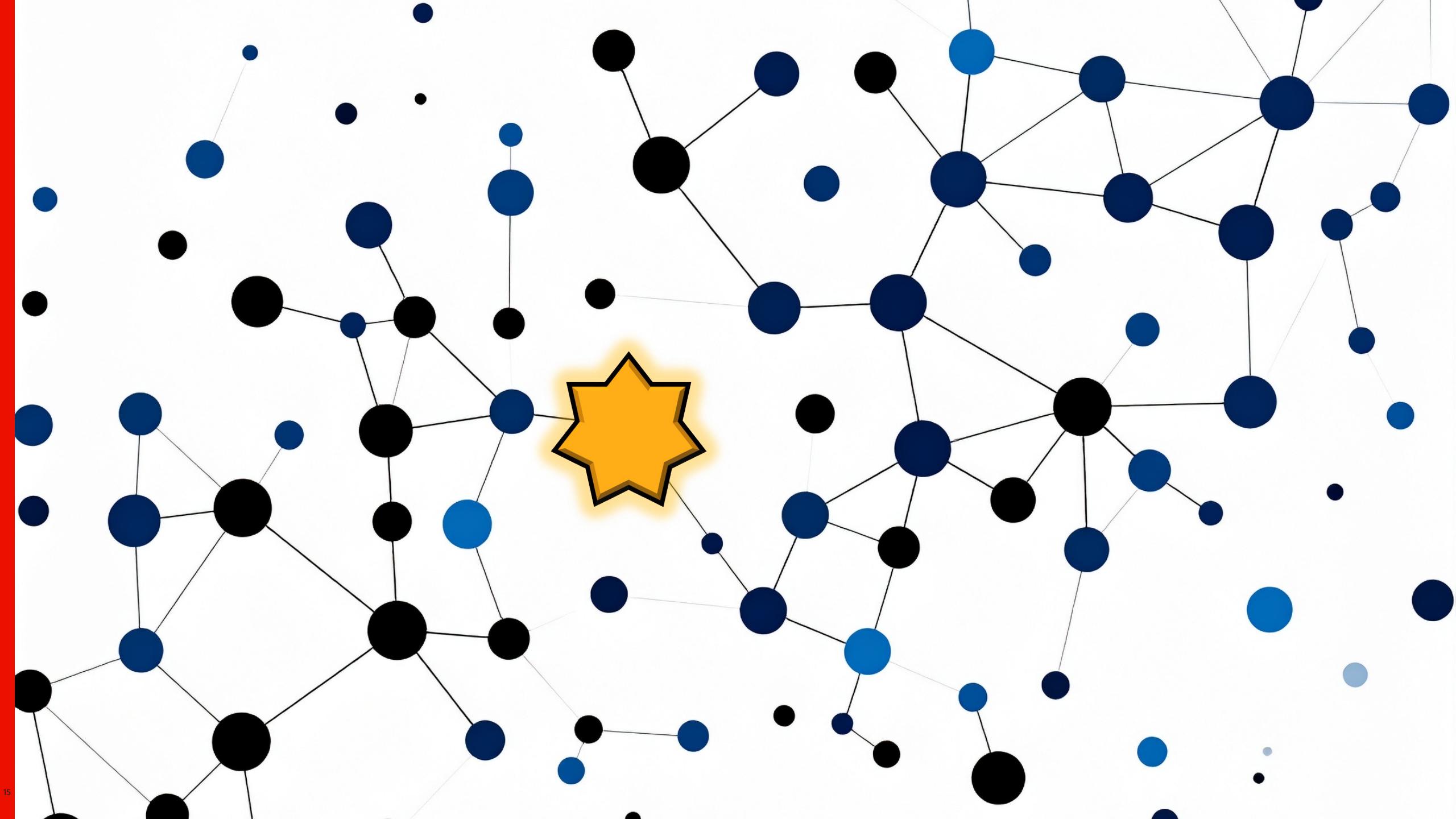


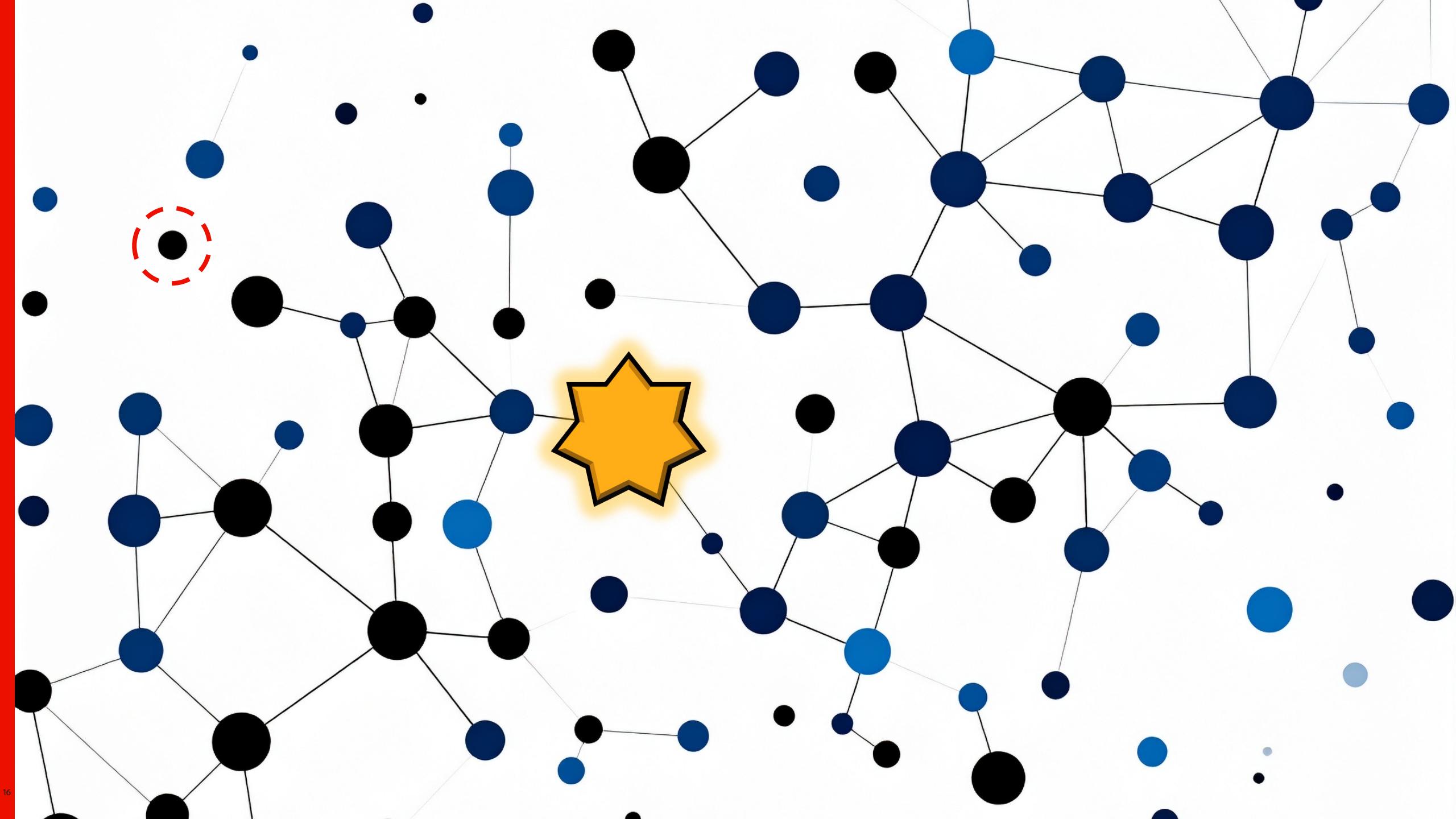
# Correctness

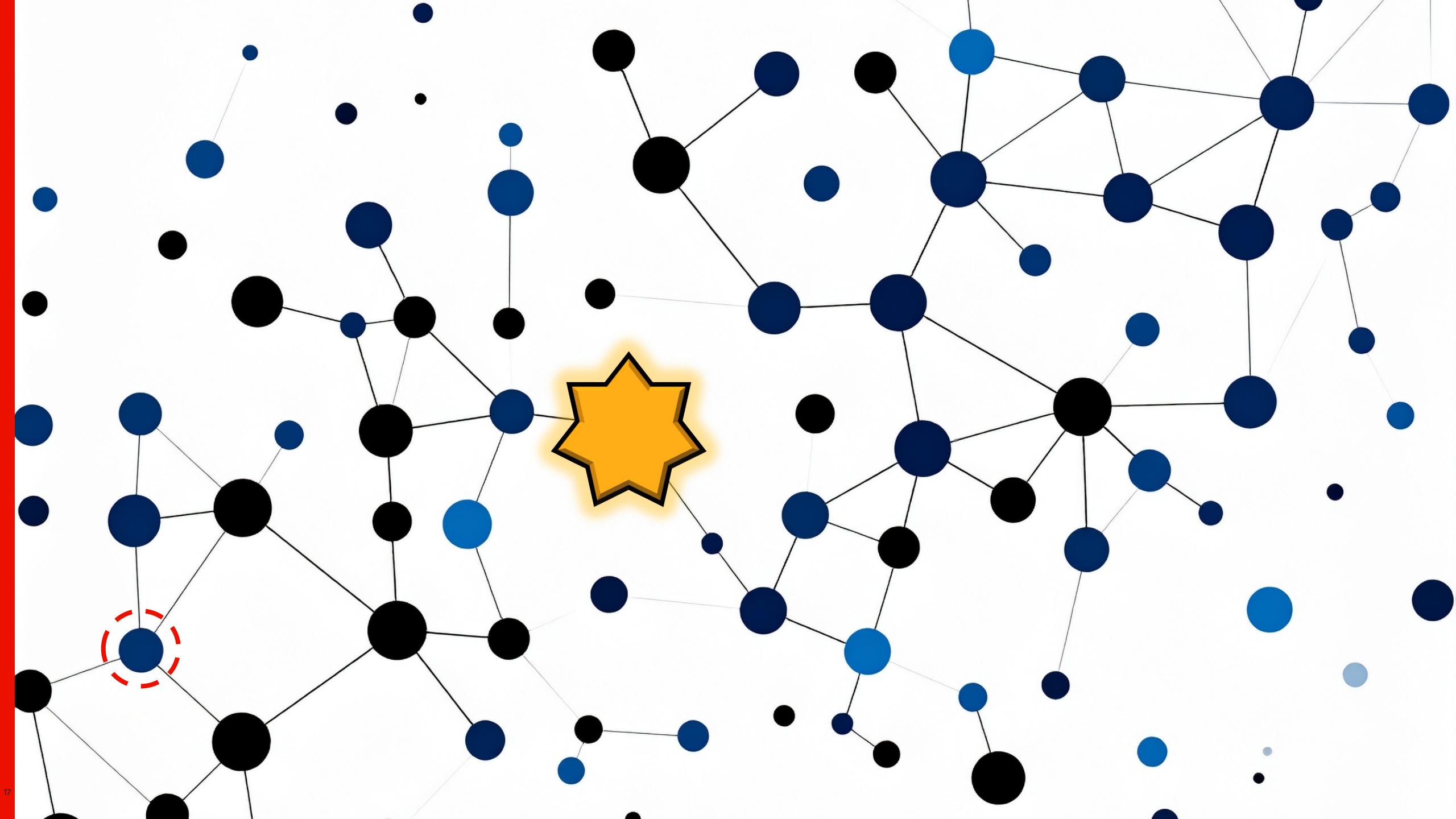
For correctness to be well-defined, it seems like

1. There needs to be an agreed-upon “basis of truth”.
  1. Usually self-evident, but sometimes hypothetical
2. There's usually some way to take claims beyond the basis of truth and demonstrate that the truth supports those claims or contradicts them.

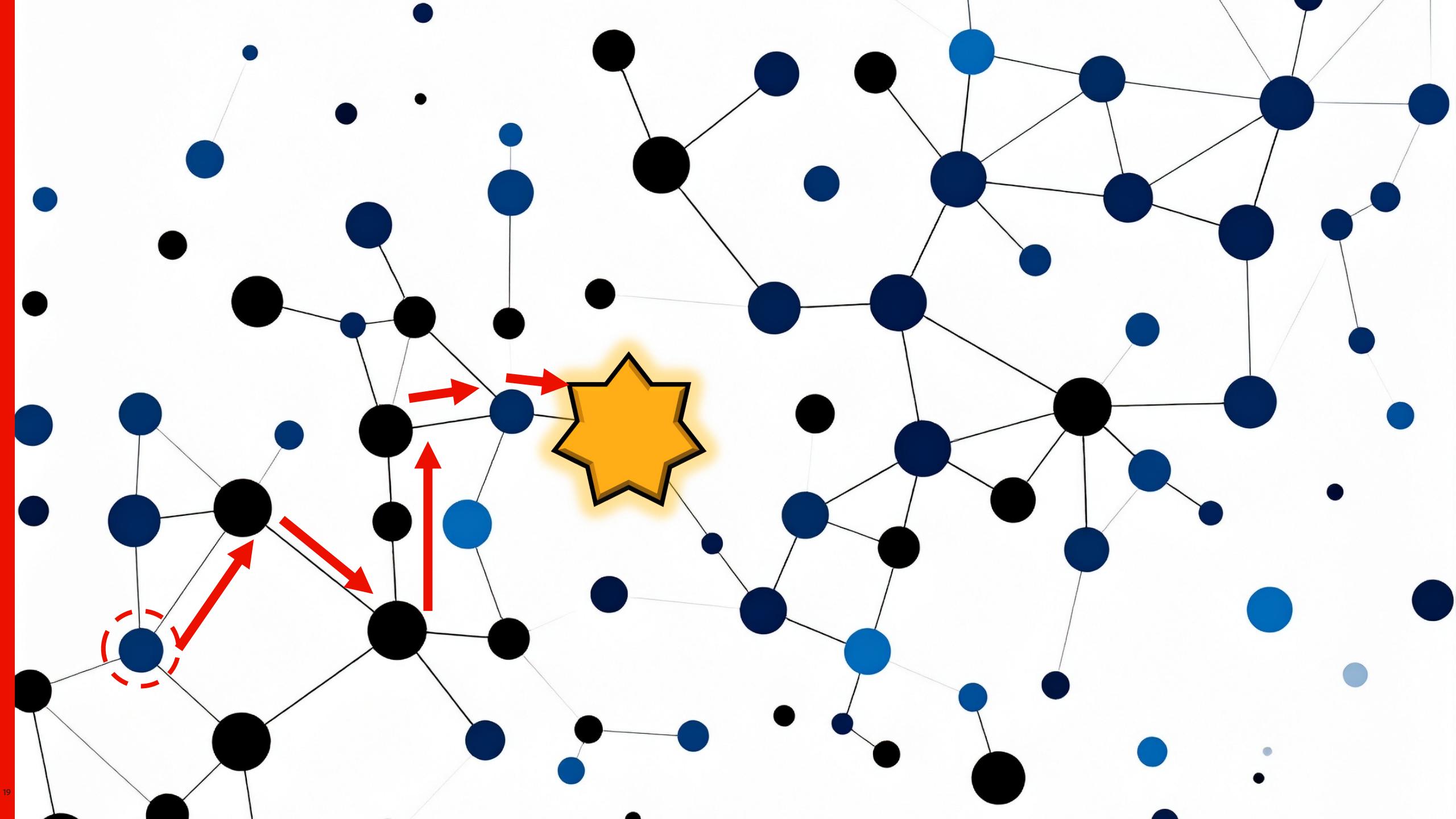












# **What is truth?**

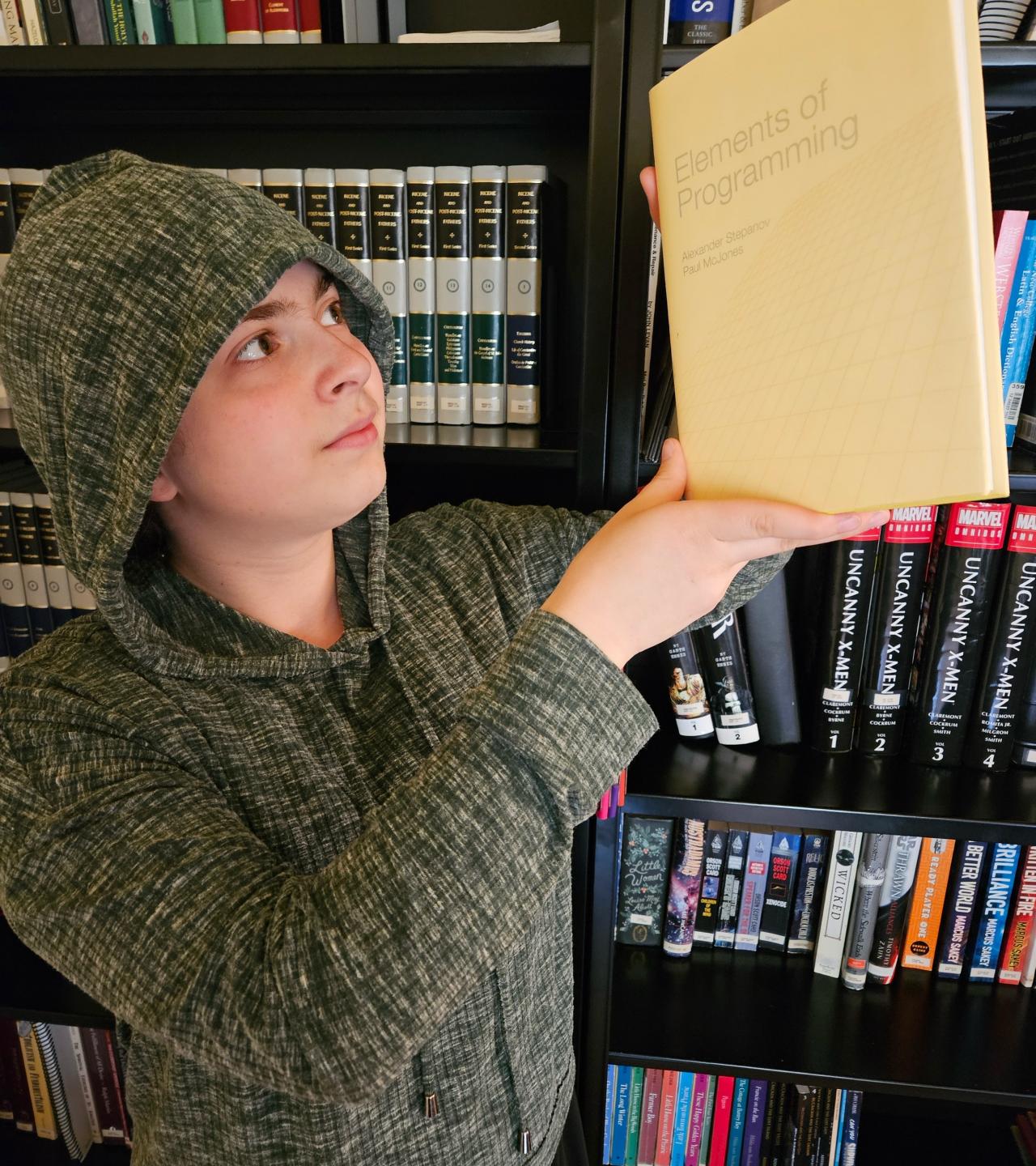
# Use an inspired person

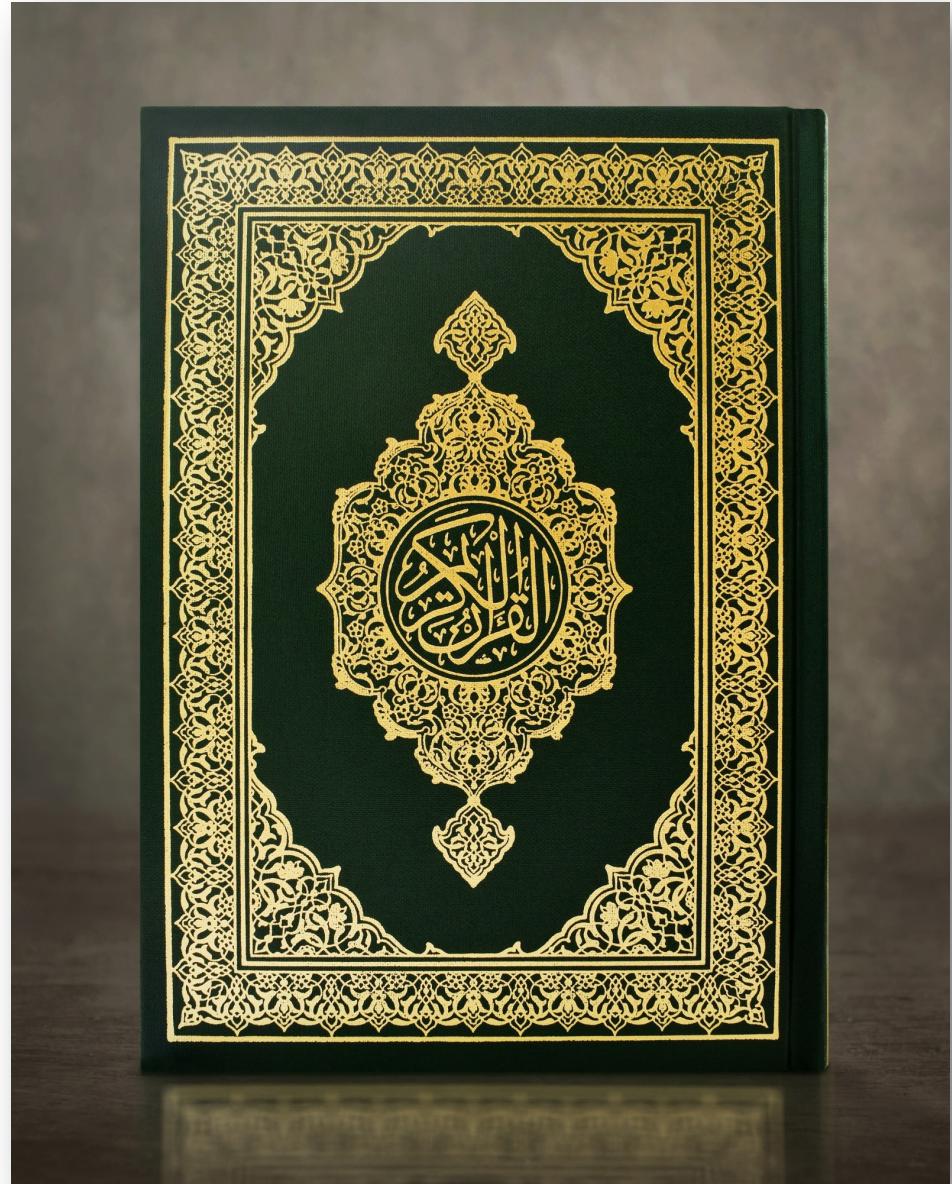
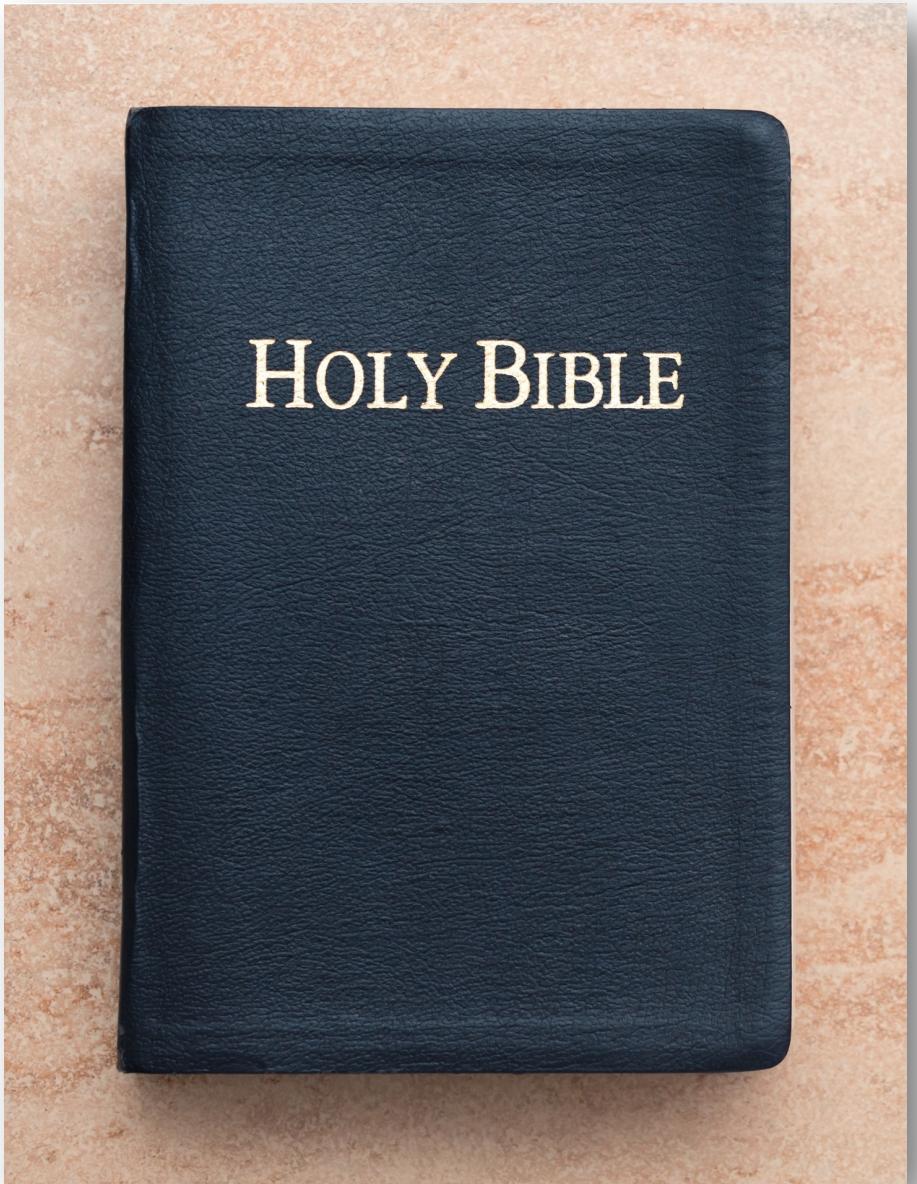
- “basis of truth” defined by a particular person’s authority
- Requires very low mental energy



# Use an inspired book

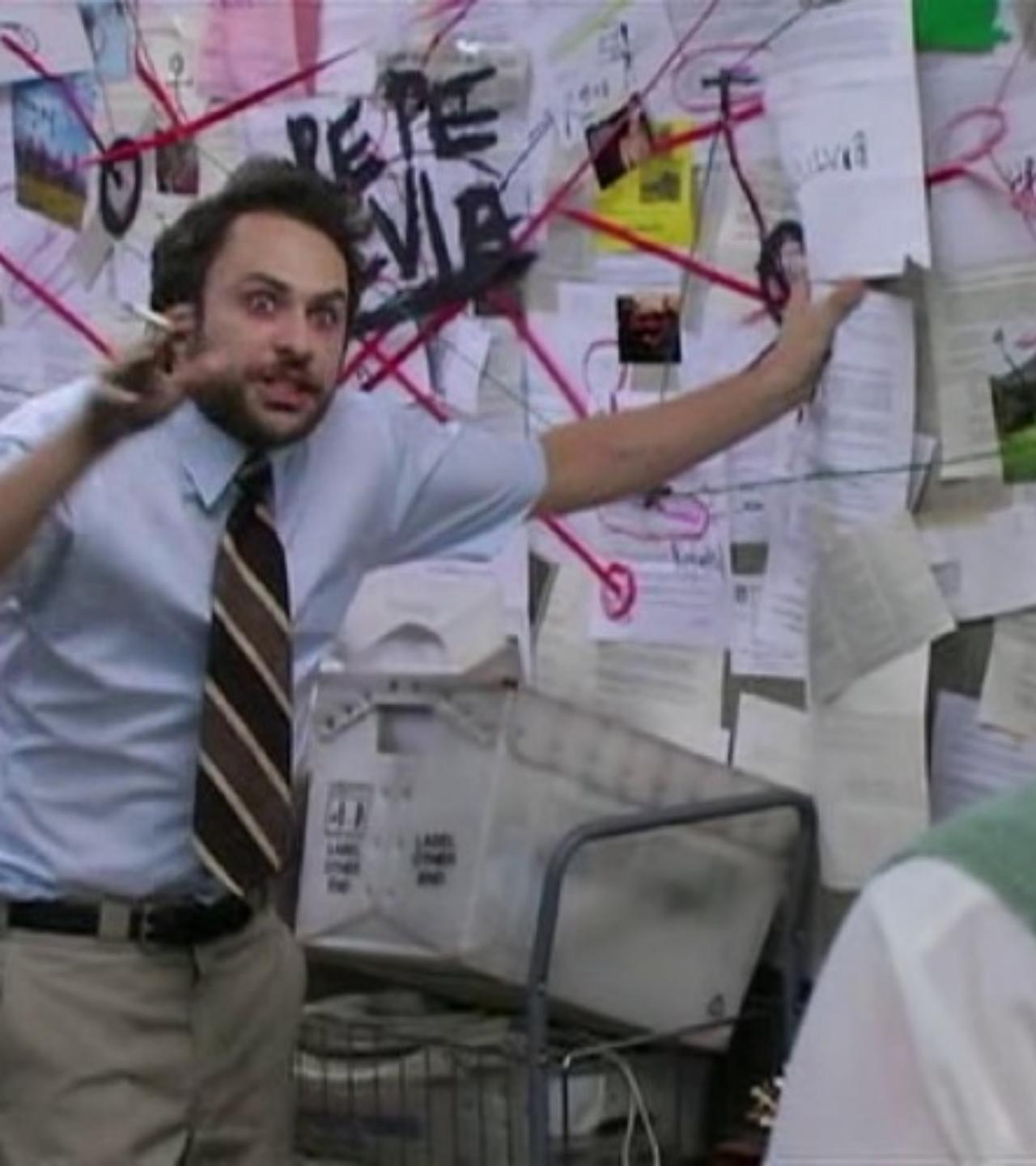
- Most questions can be answered
- Requires derivation work
- Subject to unresolvable contradictions





# Role your own

- Requires a lot of mental energy
- Constant vigilance is required to avoid contradictions
- Frequently coincides with madness



# **Your basis of truth?**

# Is this program correct?

```
int main() {  
    return 0;  
}
```

# Is this program correct?

```
int main(_) {  
    return 0;  
}
```

# Is this program correct?

```
#include <climits>

int main() {
    int i = INT_MAX;
    i++;
    return 0;
}
```

# Is this program correct?

```
#include <cstdlib>

int main() {
    std::abort();
}
```

# Is this program correct?

```
#include <cstdlib>

int main() {
    std::exit(0);
}
```

# **What about function correctness?**

# Is f correct?

```
void f(int &i) {  
    i++;  
}
```

# Is inc2 correct?

```
void f(int &i) {  
    i++;  
}  
  
// Sets x to x+2  
void inc2(int &x) {  
    f(x);  
    f(x);  
}
```

# Is inc2 correct?

```
void f(int &i) {  
    i++;  
}  
  
// Sets x to x+2  
// - precondition: x < INT_MAX-2  
void inc2(int &x) {  
    f(x);  
    f(x);  
}
```

# Is inc3 correct?

```
// Sets x to x+2
// - precondition: x < INT_MAX-2
void inc3(int &x) {
    ++x;
    ++x;
}
```

# Is inc4 correct?

```
// Sets x to x+2
// - precondition: x < INT_MAX-2
void inc3(int &x) {
    ++x;
    ++x;
}

// Sets x to x+4
// - precondition: x < INT_MAX-4
void inc4(int &x) {
    inc3(x);
    inc3(x);
}
```

# Is qfe correct?

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

```
// Returns an x such that ax2+bx+c=0 holds
float qfe(float a, float b, float c) {
    return -b + sqrt(b*b - 4*a*c)/2*a;
}
```

# What can make qfe correct?

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- a. // Returns an x such that  $ax^2+bx+c=0$  holds
- b. // Returns an x such that  $ax^2+bx+c=0$  holds or NaN
- c. // Returns an x such that  $ax^2+bx+c=0$  holds or NaN
  - // if it's outside the range that this function can compute.
- d. // Returns an x such that  $ax^2+bx+c=0$  holds and none of the
  - // operations in ' $-b + \sqrt{b^2 - 4ac}/2a$ ' overflow and
  - //  $b^2 > 4ac$

```
float qfe(float a, float b, float c) {  
    return -b + sqrt(b*b - 4*a*c)/2*a;  
}
```

# Rigor

# Does program correctness have a mathematical basis?

Say a function specification is a predicate  $P$  over C++ functions  $f_x$ .

$P(f_x)$  holds if  $f_x$  adheres to the specification.

- Are these predicates allowed to bifurcate based on non-lexical whitespace?
- What if the predicates consisted of pairs of the world state before the call and the world state after the call?
- Or maybe they consist of relations between before/after world states?
- How would you deal with randomness?

# Complexity in contracts

```
// Returns an index to the first element of 'v' not ordered before 'i', or -1 if no such element is found.  
// Runs in O(log(n)) time.  
// - precondition: v is sorted  
size_t lower_bound(int i, std::span<int> v);
```

- Runtime is not expressible if predicates are relations between before/after world states.
- What about using the C++ abstract machine?
- The as-if rule allows compilers to change complexity. What to do?

# Favorite talk of the week: Computing Correctness



## Computing Correctness

What can formal verification do for us?

Beginner

Intermediate

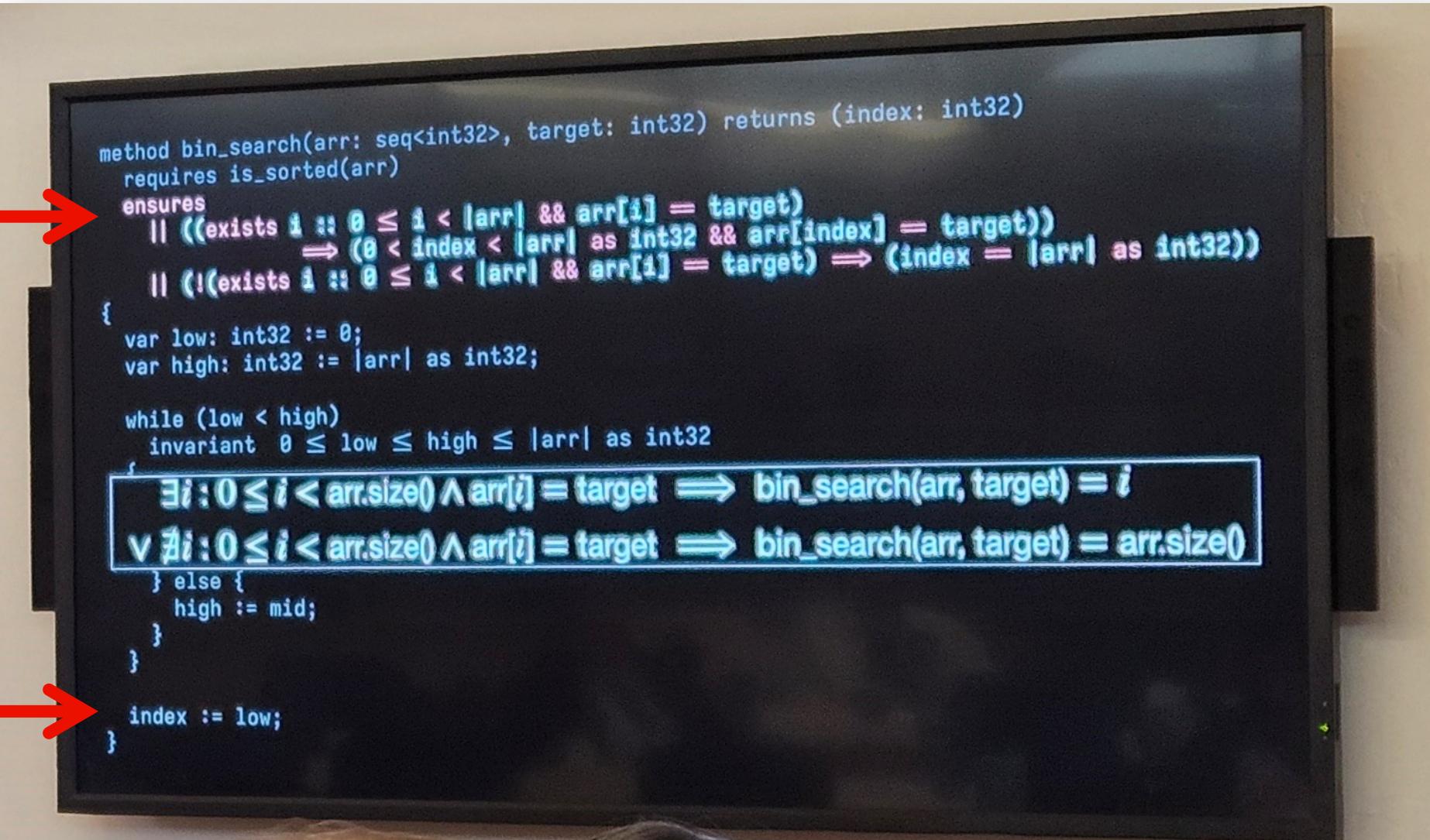
Math/Analysis



Do the words "*formal verification*" send shivers of concern down your spine? This talk will introduce the concept of automated formal verification, and explore how learning about automated formal verification provides a foundation for writing clean code that can be reasoned about. While formally verifying an entire code base is not needed for many projects, the principles behind formal verification apply to all projects and can improve your code. This talk specifically explores model checking and deductive reasoning techniques, demonstrates how we can reason "like a computer", and aims to enable better software design.

Nick Waddoups

# The Dafny example



```
method bin_search(arr: seq<int32>, target: int32) returns (index: int32)
  requires is_sorted(arr)
  ensures
    || ((exists i :: 0 ≤ i < |arr| && arr[i] = target) ⇒ (0 < index < |arr| as int32 && arr[index] = target))
    || (!exists i :: 0 ≤ i < |arr| && arr[i] = target) ⇒ (index = |arr| as int32)
  {
    var low: int32 := 0;
    var high: int32 := |arr| as int32;

    while (low < high)
      invariant 0 ≤ low ≤ high ≤ |arr| as int32
      {
         $\exists i : 0 \leq i < \text{arr.size}() \wedge \text{arr}[i] = \text{target} \Rightarrow \text{bin\_search}(\text{arr}, \text{target}) = i$ 
         $\vee \forall i : 0 \leq i < \text{arr.size}() \wedge \text{arr}[i] = \text{target} \Rightarrow \text{bin\_search}(\text{arr}, \text{target}) = \text{arr.size}()$ 
        }
        {
          if arr[low] > target
            then
              high := low;
            else
              low := high + 1;
        }
      }
    index := low;
  }
```

**Does correctness have a solid mathematical foundation?**

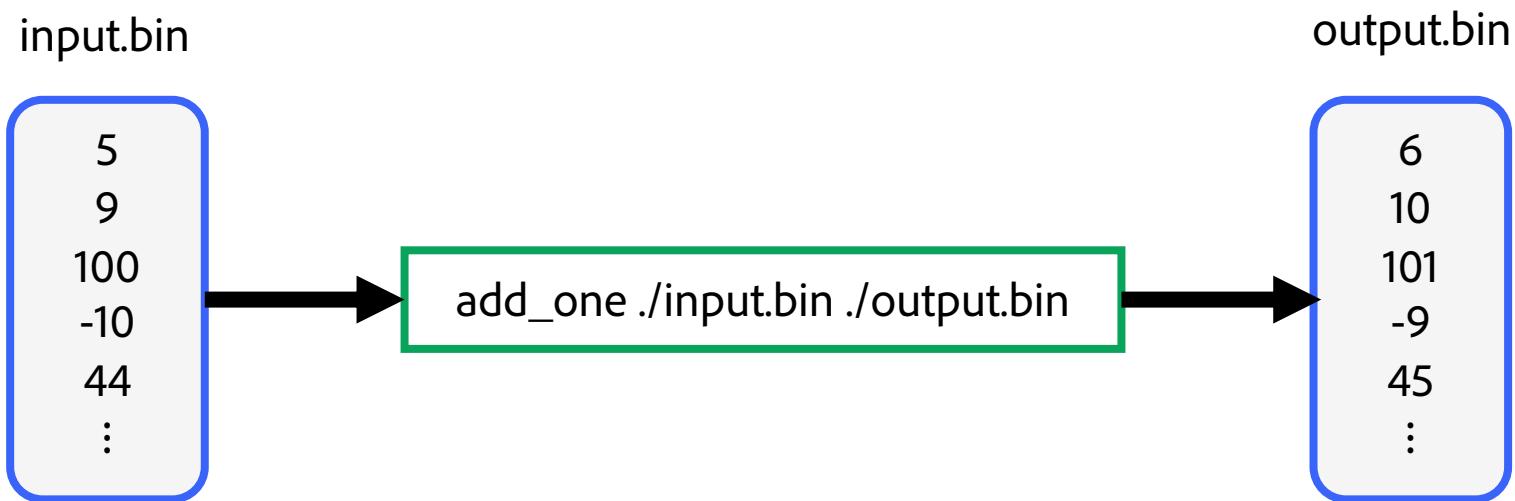
**Does correctness have a *satisfying* mathematical foundation?**

# Does correctness have a *satisfying* mathematical foundation?

*Not that I have seen.*

- Correctness cannot be separated from human communication
  - Varying degrees of specificity and alignment.
  - All communication counts: including function names.
  - Basis of truth, assumptions, is an essential element
- Correctness and requirements are closely related
  - It is nonsensical to proclaim a solution correct that does not meet agreed-upon requirements
  - “Performance optimized” can be a more useful as a contract than a complexity specification
- Basis of truth must be shared between human participants
  - Using whatever is common will result in more opportunities for discourse

# **add\_one**

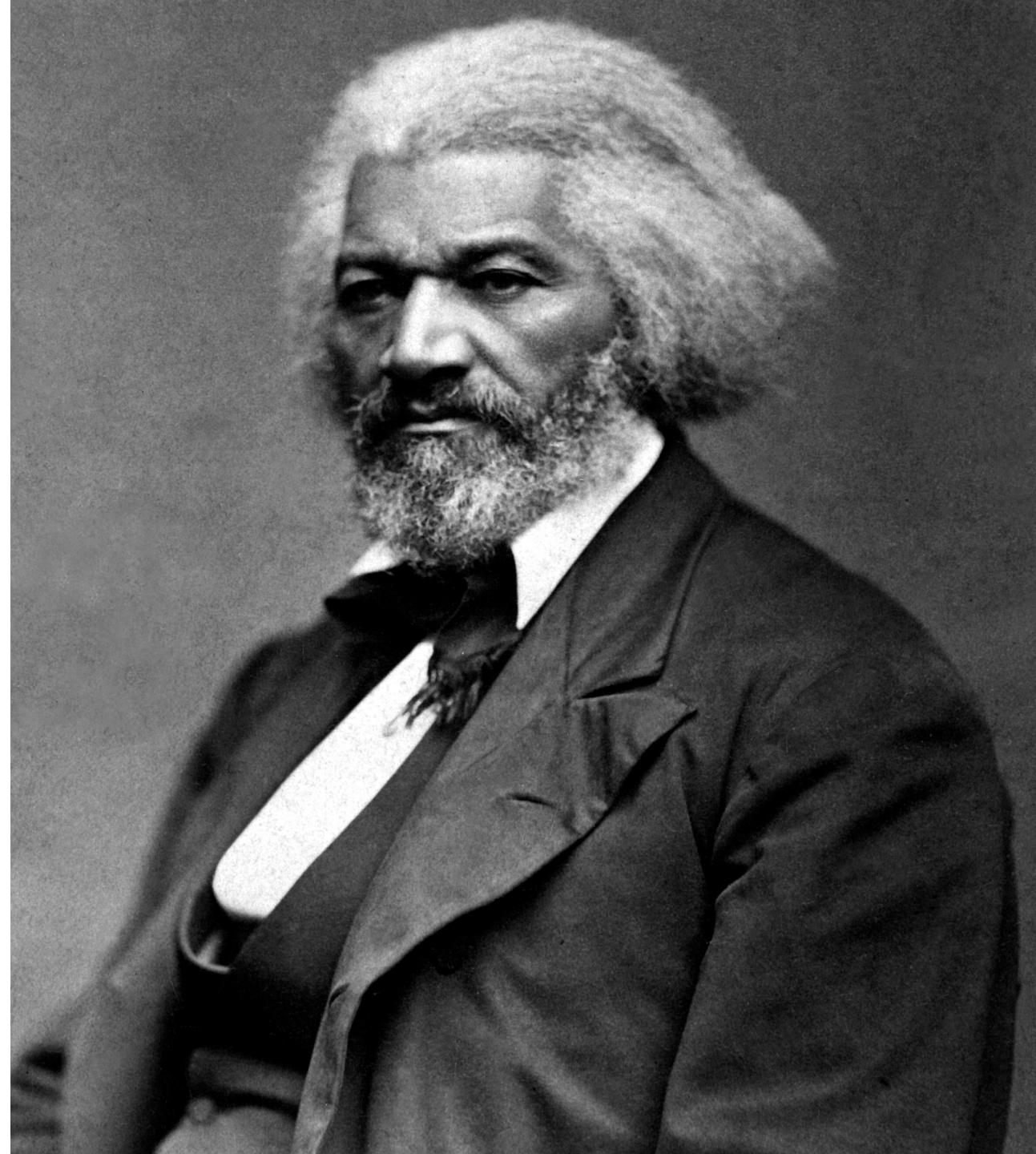


- Files are on an ultra-high performance drive so IO speed is negligible.
- This is being run on a highly specialized embedded processor that isn't superscalar.
- This is a “never enough performance” use-case.
- The `input.bin` files will never have ints nearing `INT_MAX`

**Is it sometimes better to write incorrect code?**

# **Frederick Douglass**

- 1889-1891
- Former slave
- Famous speechwriter and abolitionist



# What to the Slave Is the Fourth of July?

## FREDERICK DOUGLASS'S "FOURTH OF JULY" SPEECH (1852)

July 5, 1852

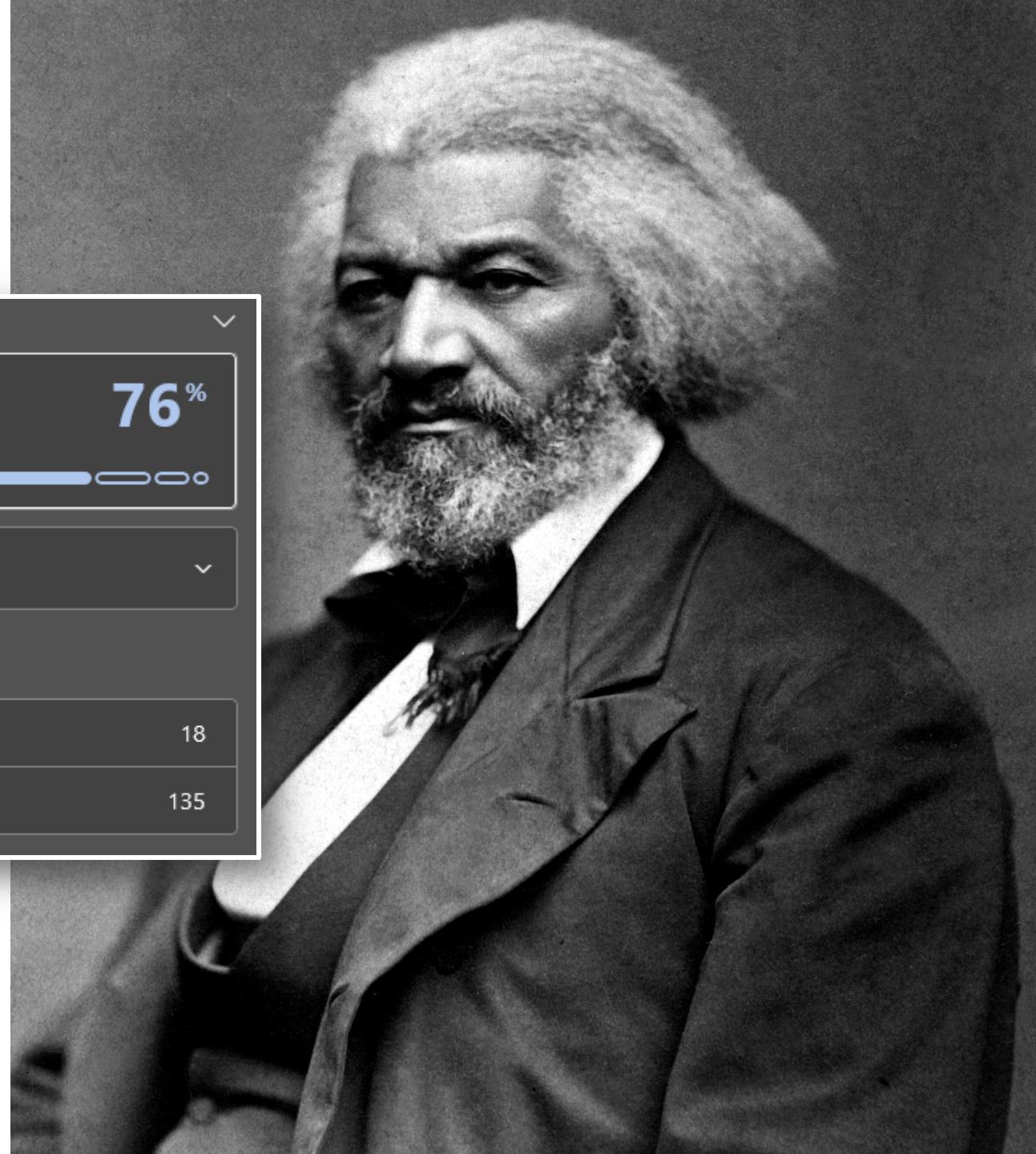
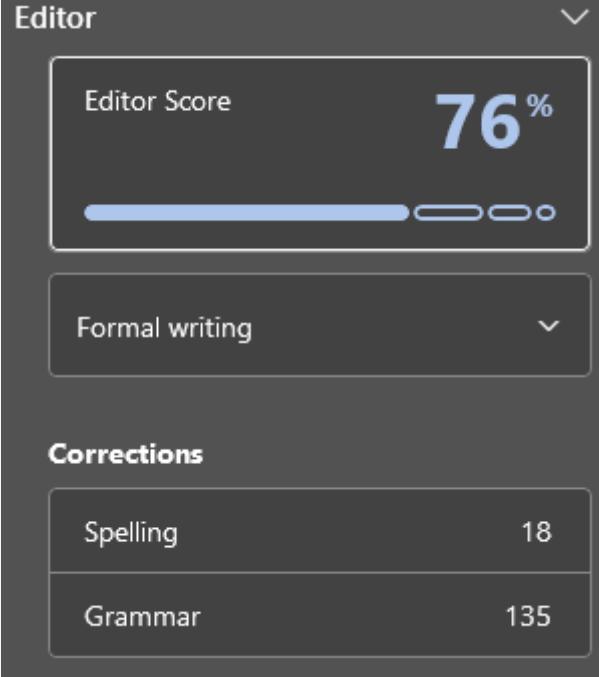
Mr. President, Friends and Fellow Citizens:

He who could address this audience without a quailing sensation, has stronger nerves than I have. I do not remember ever to have appeared as a speaker before any assembly more shrinkingly, nor with greater distrust of my ability, than I do this day. A feeling has crept over me, quite unfavorable to the exercise of my limited powers of speech. The task before me is one which requires much previous thought and study for its proper performance. I know that apologies of this sort are generally considered flat and unmeaning. I trust, however, that mine will not be so considered. Should I seem at ease, my appearance would much misrepresent me. The little experience I have had in addressing public meetings, in country school houses, avails me nothing on the present occasion.

The papers and placards say, that I am to deliver a 4th July oration. This certainly sounds large, and out of the common way, for me. It is true that I have often had the privilege to speak in this beautiful Hall, and to address many who now honor me with their presence. But neither their familiar faces, nor the perfect gage I think I have of Corinthian Hall, seems to free me from embarrassment.

The fact is, ladies and gentlemen, the distance between this platform and the slave plantation, from which I escaped, is considerable—and the difficulties to be overcome in getting from the latter to the former, are by no means slight. That I am here today, is, to me, a matter of astonishment as well as of gratitude. You will not, therefore, be surprised, if in what I have to say, I evince no elaborate preparation, nor grace my speech with any high sounding exordium. With little experience and with less learning, I have been able to throw my thoughts hastily and imperfectly together; and trusting to your patient and generous indulgence, I will proceed to lay them before you.

This, for the purpose of this celebration, is the 4th of July. It is the birthday of your National Independence, and of your political freedom. This, to you, is what the Passover was to the emancipated people of God. It carries your minds back to the day, and to the act of your great deliverance; and to the signs, and to the wonders, associated with that act that day. This celebration also marks the beginning of another year of your national life; and reminds you that the Republic of America is now 76 years old. I am glad, fellow-citizens, that your nation is so young. Seventy-six years, though a good old age for a man, is but a mere speck in the life of a nation. Three score years and ten is the allotted time for individual men; but nations number their years by thousands. According to this fact, you are, even now only in the beginning of you national career, still lingering in the period of childhood. I repeat, I am glad this is so. There is hope in the thought, and hope is much needed, under the dark clouds which lower above the horizon. The eye of the reformer is met with angry flashes, portending disastrous times; but his heart may well bear lighter at the thought that America is young, and that she is still in the impressible stage of her existence. May he not hope that high lessons of wisdom, of justice and of truth, will yet give direction to her destiny? Were the nation older, the patriot's heart might be sadder, and the reformer's brow heavier. Its future might be shrouded in gloom, and the hope of its prophets go out in sorrow. There is consolation in the thought, that America is young. Great streams are not easily turned from channels, worn deep in the course of ages. They may sometimes rise in quiet and stately majesty, and inundate the land, refreshing and fertilizing the earth with their mysterious properties. They may also rise in wrath and fury, and bear away, on their angry waves, the accumulated wealth of years of toil and hardship. They, however, gradually flow back to the same old channel, and flow on as serenely as ever. But, while the river may not be turned aside, it may dry up, and leave nothing behind but the withered branch, and the unsightly rock, to howl in the abyss-sweeping wind, the sad tale of departed glory. As with rivers so with nations.





# How does correctness relate to quality?

- *What* is incorrect makes a difference.
  - Did they take 1 point or 10 points off?
- There's frequently a curve
  - How did the rest of the class do?
- Sometimes correctness is a very small component
  - Essay question vs. multiple choice
  - Interface design

# **Are we writing correct programs?**

Say we write 30-line functions correctly 99.9% of the time.

Consider a 1-million-line program. How likely is it that it is correct?

$$(0.999)^{33333} \approx 2.195 \times 10^{-15}$$

Or 0.00000000002195% of the time

**We've made the decision to write incorrect code.**

# We've made the decision to write incorrect code.

- I'm not sure we could if we wanted to
- Sometimes we talk as if
  - Writing correct code is our goal
  - Correct code trumps other quality considerations

# Let's be real

- Admit the tradeoffs we're making between correctness and other quality factors (e.g. cost, time to market, staffing limitations)
- There's a difference between consequential bugs and non-consequential bugs
- We need to share and learn more about the other, frequently more important, aspects of software quality

# Conclusion



Click to add text



# Adobe