

The background features a dark blue gradient with large, overlapping, semi-transparent shapes in shades of purple and magenta. Two thin, light blue lines intersect to form a large 'A' shape on the right side of the image.

# AWS re:Invent

DECEMBER 2 - 6, 2024 | LAS VEGAS, NV

AIM393

# Introducing Automated Reasoning checks in Amazon Bedrock Guardrails

**Stefano Buliani**

(he/him)

Product Manager

Amazon Web Services

**Byron Cook**

(he/him)

VP, Distinguished Scientist

Amazon Web Services



# Agenda

- 01 How we got here – finding a gen AI strategy
- 02 Automated Reasoning checks
- 03 What is Automated Reasoning anyway?
- 04 Getting started with Automated Reasoning checks and demo
- 05 Next steps

# How we got here





**Ninety-five percent** of CIOs surveyed believe in the moderate or extensive potential value of gen AI, with top areas of value being in productivity, customer experience and digital business transformation

Gartner®

Gartner, Key findings from the 2024 Gartner CIO Generative AI Survey, August 2024.  
GARTNER is a registered trademark and service mark of Gartner, Inc. and/or its affiliates in the U.S. and internationally and is used herein with permission. All rights reserved.



# Leaders are asking – what is our gen AI strategy?

**Understand** large language models (LLMs)

**Educate** leaders

Identify **use cases**

# Experiment, experiment, experiment

Understand LLMs

Educate leaders

Identify **use cases**

**Onboard** employees

Simplify **customer support**

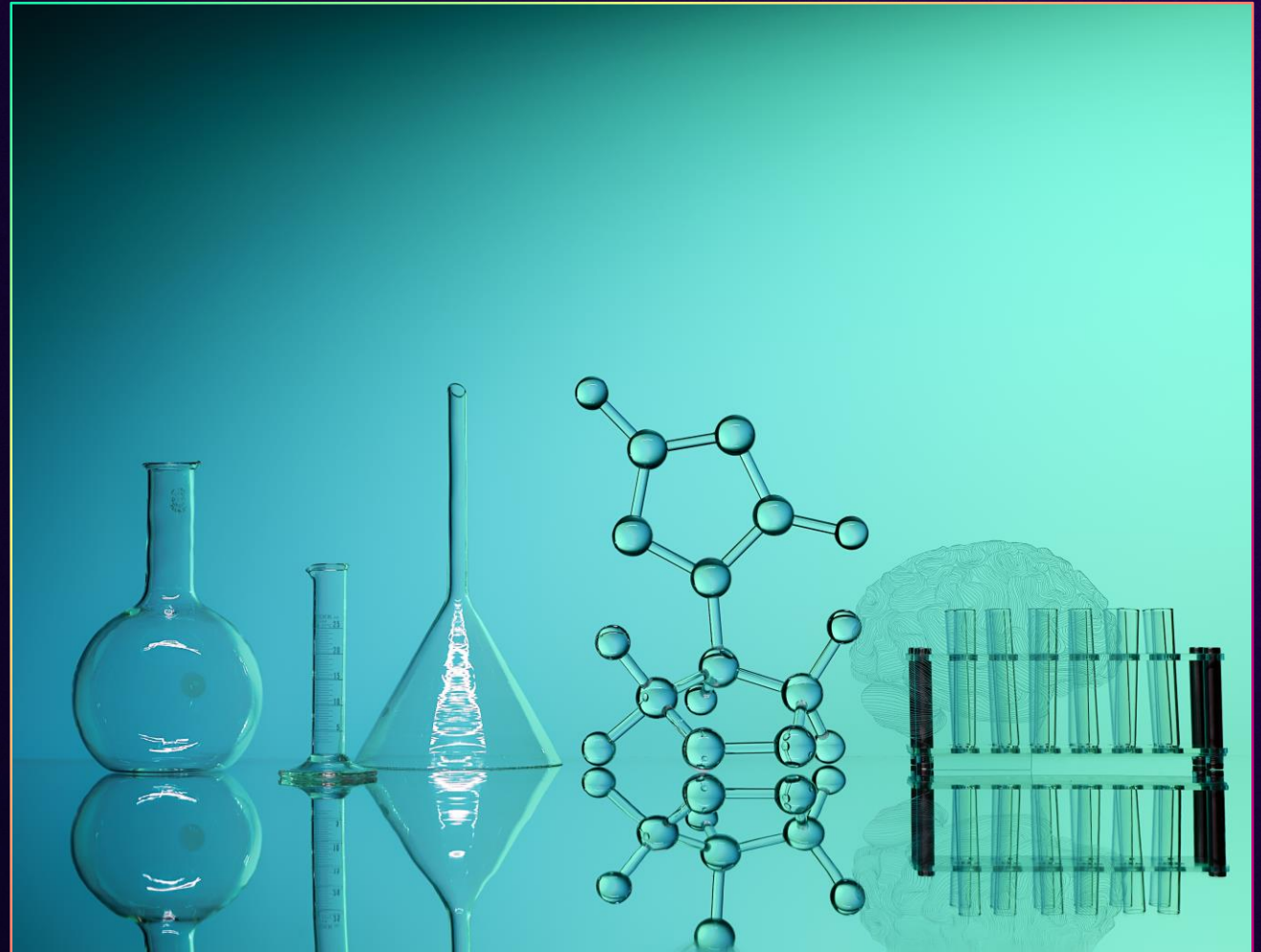
**Automate** decisions



# Wow comes first when experimenting

First reaction is always **wow**

We can build so fast





# Then we test for production

From experiment to production

Lacks accuracy



# Hallucinations can be subtle

## Ground truth

My friend Sam and I enjoy solving Advent of Code puzzles. We spend hours on Slack discussing the trade offs between different algorithms to solve the problem. Our passion for this activity brings us closer as good friends.

## LLM summary

Ben and I love solving Advent of Code puzzles, and this makes us good friends.

# Hallucinations can be subtle

## Ground truth

My friend **Sam** and I enjoy solving Advent of Code puzzles. We spend hours on Slack discussing the trade offs between different algorithms to solve the problem. Our passion for this activity brings us closer as good friends.

## LLM summary

**Ben** and I love solving Advent of Code puzzles, and this makes us good friends.

**Guess who is not going to production?**



The **concern of hallucinations** that result in reasoning errors was the top-rated potential risk (**59%**), followed by bad actors creating misinformation (48%) and privacy assurances (44%).

Gartner®

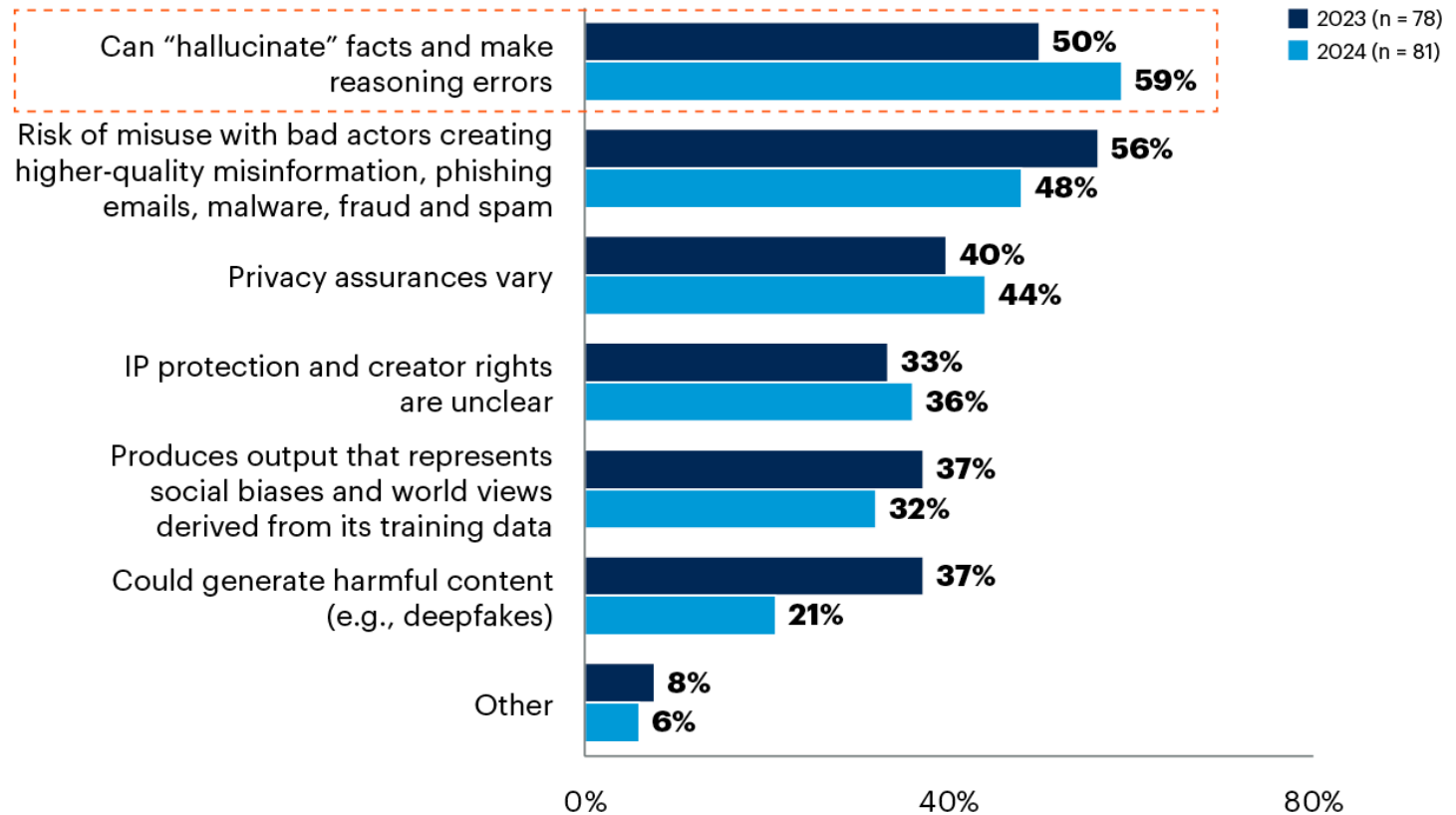
Gartner, Key Findings From the 2024 Gartner CIO Generative AI Survey, August 2024.

# You are in good company

Gartner, Key Findings From the 2024 Gartner CIO Generative AI Survey, August 2024.

## Potential Risks of GenAI

Multiple responses allowed



n varies by year; CIOs; excluding, "Not sure"

Q: Finally, which of these potential risks of generative AI are you most concerned about in the context of your enterprise?

Source: 2023 Gartner CIO Generative AI Survey; 2024 Gartner CIO Generative AI Survey; Gartner's Research Circle members and external participants

814380\_C

Hallucinations are not a bug. It's **creativity**.



# Automated Reasoning checks



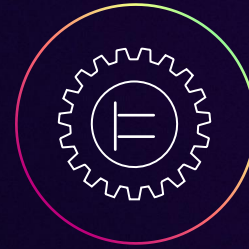


# We have three objectives



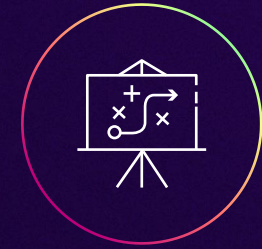
## Accurate

Identifies and suggests corrections for inaccurate factual claims on supported knowledge



## Sound

When it says something is incorrect – it is; if we cannot make a claim one way or another, we'll tell you



## Transparent

We can explain exactly why we believe a claim is accurate or not

# When it works best

- ✓ HR policies
- ✓ Laws and regulation
- ✓ Operational workflows

# It's not suited for

- ✗ Marketing messaging best practices
- ✗ Probabilistic calculations (what are the chances of?)
- ✗ Qualitative descriptions

# Amazon Bedrock Guardrails



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context

# Amazon Bedrock Guardrails



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context

# Amazon Bedrock Guardrails



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context

# Amazon Bedrock Guardrails



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context

# Amazon Bedrock Guardrails



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context

# Amazon Bedrock Guardrails



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context



# Even more complete



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context



Identify, correct, and explain factual claims in responses based on ground truth formal logic

# Even more complete



Evaluate prompts and model responses for agents, knowledge bases, FMs in Amazon Bedrock, and self-managed or third-party FMs



Configure thresholds to filter harmful content, jailbreaks, and prompt injection attacks



Define and disallow denied topics with short natural language descriptions



Remove personally identifiable information (PII) and sensitive information in generative AI applications



Filter hallucinations by detecting groundedness and relevance of model responses based on context



Identify, correct, and explain factual claims in responses based on ground truth formal logic

AIM393

# Introduction to Automated Reasoning checks in Amazon Bedrock Guardrails

**Stefano Buliani**

(he/him)

Product Manager

Amazon Web Services

**Byron Cook**

(he/him)

VP, Distinguished Scientist

Amazon Web Services



# What *is* automated reasoning?

a.k.a. *symbolic AI*

What *is* automated reasoning?

DARTMOUTH COLLEGE  
Department of Mathematics & Astronomy  
HANOVER · NEW HAMPSHIRE

March,  
1956

Mr. Ray Solomonoff  
Technical Research Group  
17 Union Square West  
New York, New York

Dear Ray:

You are one of the people  
we should like to invite to the  
"Summer Research Project on Artificial  
Intelligence."

Terms: \$1,200 - \$900 of  
which will probably count as a fel-  
lowship and be tax free, plus  
traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H.  
(a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy } for  
A. Minsky } all  
John Hollnagel } 2 months  
R. Solomonoff }  
Julian Bigelow }

Shannon } some  
Rochester } of these  
Selfridge } for part  
McCulloch } of time.  
Newell }  
Simon }  
McCarthy }  
Et al }

Dartmouth, 1956: The term  
*Artificial Intelligence* is coined

DARTMOUTH COLLEGE  
Department of Mathematics & Astronomy  
HANOVER · NEW HAMPSHIRE

March,  
1956

Mr. Ray Solomonoff  
Technical Research Group  
17 Union Square West  
New York, New York

Dear Ray:

You are one of the people  
we should like to invite to the  
"Summer Research Project on Artificial Intelligence."

Terms: \$1,200 - \$900 of  
which will probably count as a fel-  
lowship and be tax free, plus  
traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H.  
(a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy } for  
M. Minsky } all  
John O'Halloran } 2 months  
R. Solomonoff }  
Julian Bigelow }

Shannon } some  
Rochester } of these  
Selfridge } for part  
McCulloch } of time.  
Newell }  
Simon }  
McCarthy }  
Et al }



Oliver  
Selfridge

Nathaniel  
Rochester

Marvin  
Minsky

John  
McCarthy

Ray  
Solomonoff

Peter  
Milner

Claude  
Shannon

DARTMOUTH COLLEGE  
Department of Mathematics & Astronomy  
HANOVER · NEW HAMPSHIRE

March,  
1956

Mr. Ray Solomonoff  
Technical Research Group  
17 Union Square West  
New York, New York

Dear Ray:

You are one of the people  
we should like to invite to the  
"Summer Research Project on Artificial  
Intelligence."

Terms: \$1,200 - \$900 of  
which will probably count as a fel-  
lowship and be tax free, plus  
traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H.  
(a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy } for  
A. Minsky } all  
John O'Halloran } 2 months  
R. Solomonoff }  
Julian Bigelow }

Shannon } some  
Rochester } of these  
Selfridge } for part  
McCulloch } of time.  
Newell }  
Simon }  
McCarthy }  
Et al }

# AI

Connectionist AI

Symbolic AI

Probabilistic AI



DARTMOUTH COLLEGE  
Department of Mathematics & Astronomy  
HANOVER · NEW HAMPSHIRE

March,  
1956

Mr. Ray Solomonoff  
Technical Research Group  
17 Union Square West  
New York, New York

Dear Ray:

You are one of the people we should like to invite to the "Summer Research Project on Artificial Intelligence."

Terms: \$1,200 - \$900 of which will probably count as a fellowship and be tax free, plus traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H.  
(a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy } for  
A. Minsky } all  
John O'Halloran } 2 months  
R. Solomonoff }  
Julian Bigelow }

Shannon } some  
Rochester } of these  
Selfridge } for part  
McCulloch } of time.  
Newell }  
Simon }  
McCarthy }  
Et al }

# AI

Cognitive

Logic

Connectionist AI

Symbolic AI

Probabilistic AI

Bayesian

DARTMOUTH COLLEGE  
Department of Mathematics & Astronomy  
HANOVER · NEW HAMPSHIRE

March,  
1956

Mr. Ray Solomonoff  
Technical Research Group  
17 Union Square West  
New York, New York

Dear Ray:

You are one of the people we should like to invite to the "Summer Research Project on Artificial Intelligence."

Terms: \$1,200 - \$900 of which will probably count as a fellowship and be tax free, plus traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H.  
(a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy } for all  
A. Minsky } 2 months  
John O'Halloran  
R. Solomonoff  
Julian Bigelow

Shannon } some of these  
Rochester } for part  
Selfridge } of time.  
McCulloch  
Newell  
Simon  
McCarthy  
Et al

# AI

Connectionist AI

Symbolic AI

Probabilistic AI

"Machine Learning"

"Automated Reasoning"

DARTMOUTH COLLEGE  
Department of Mathematics & Astronomy  
HANOVER · NEW HAMPSHIRE

March,  
1956

Mr. Ray Solomonoff  
Technical Research Group  
17 Union Square West  
New York, New York

Dear Ray:

You are one of the people we should like to invite to the "Summer Research Project on Artificial Intelligence."

Terms: \$1,200 - \$900 of which will probably count as a fellowship and be tax free, plus traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H.  
(a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy } for all 2 months  
A. Newell }  
John G. Holland }  
R. Solomonoff }  
Julian Bigelow }

Shannon } some of these for part of time.  
Rochester }  
Selfridge }  
McCulloch }  
Newell }  
Simon }  
McCarthy }  
Et al }

# AI

Connectionist AI

Symbolic AI

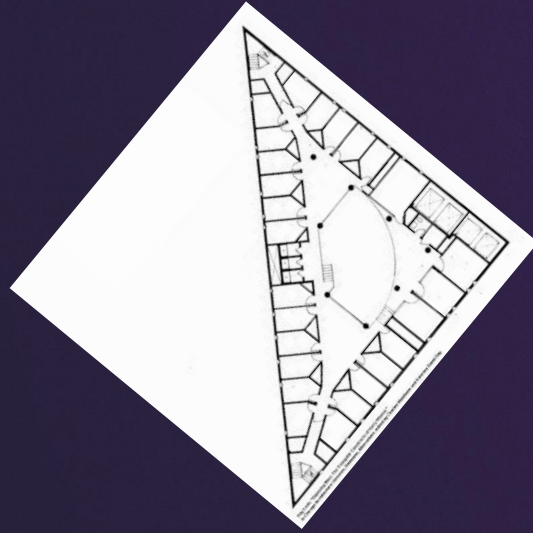
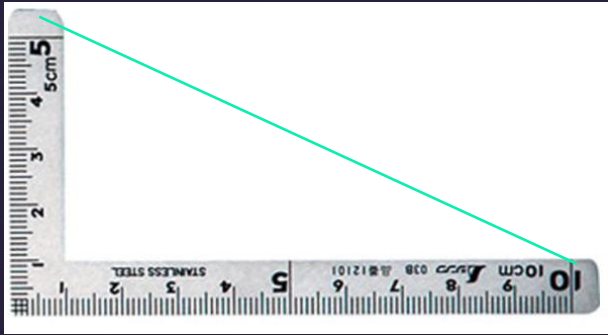
Probabilistic AI

"Machine Learning"

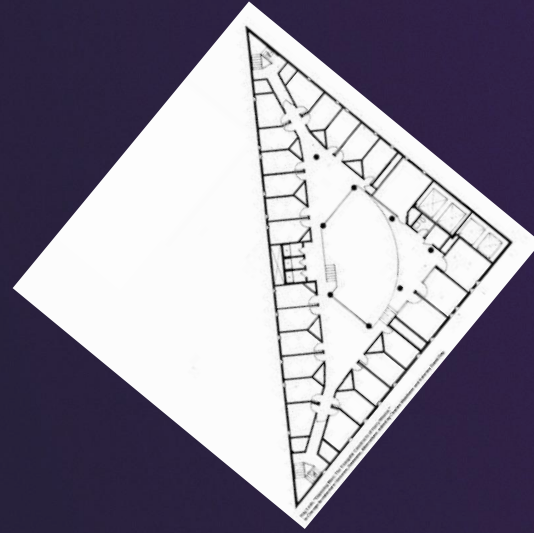
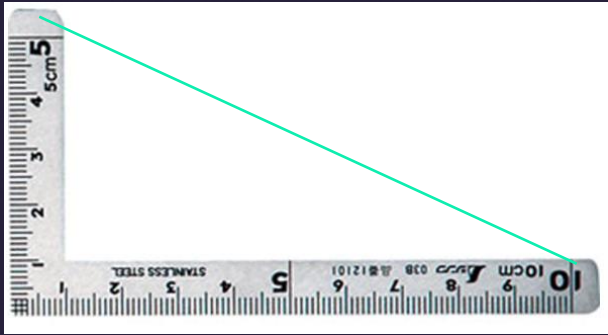
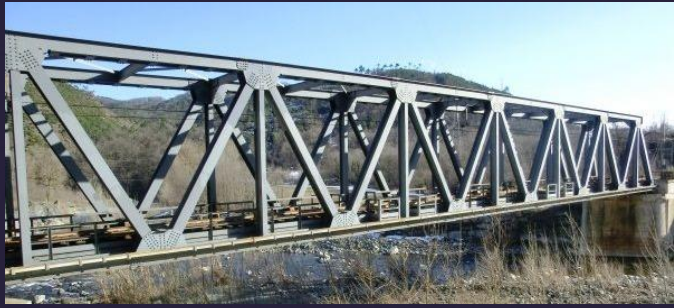
"Automated Reasoning"

Today's lingo

Today's lingo

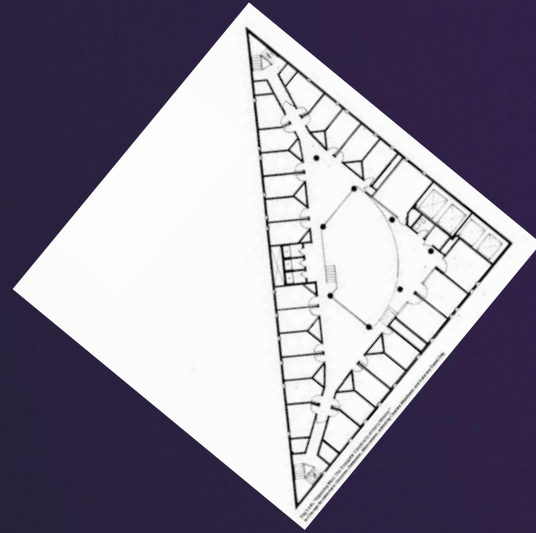


Machine learning takes some data and tries to find a **model** that hopefully will fit the remaining expected data

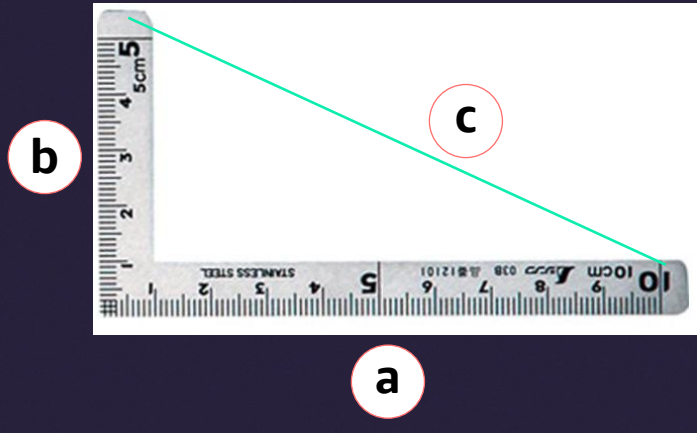


Machine learning takes some data and tries to find a **model** that hopefully will fit the remaining expected data

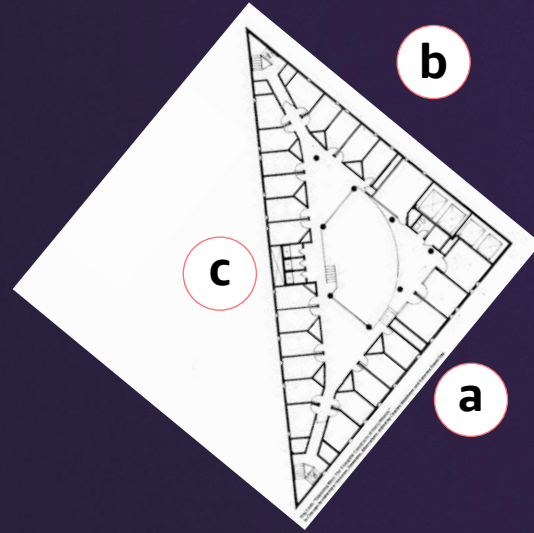
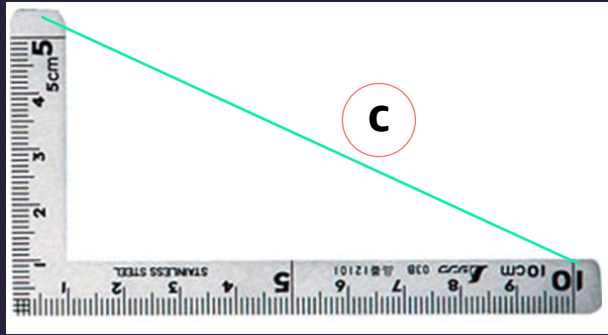
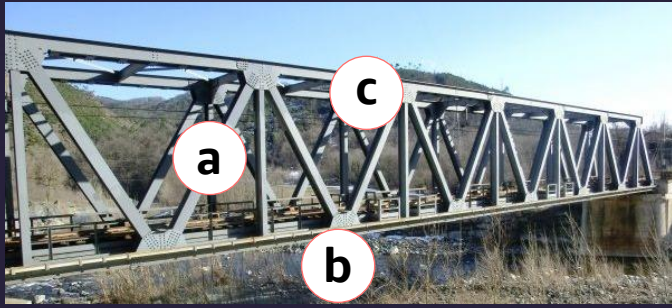
$$a^2 + b^2 = c^2$$



Machine learning takes some data and tries to find a **model** that hopefully will fit the remaining expected data

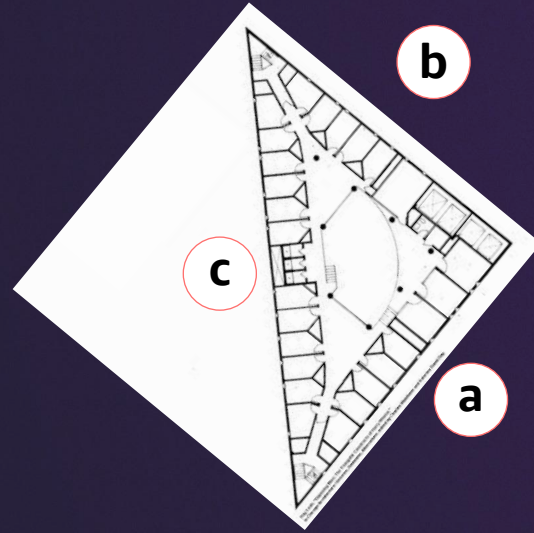
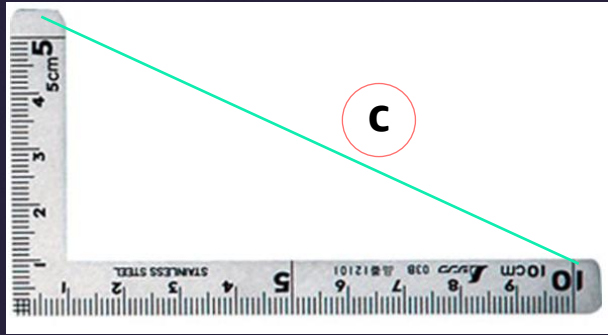
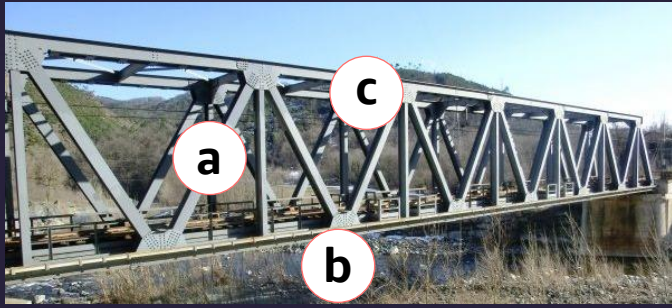


$$a^2 + b^2 = c^2$$



Machine learning takes some data and tries to find a **model** that hopefully will fit the remaining expected data

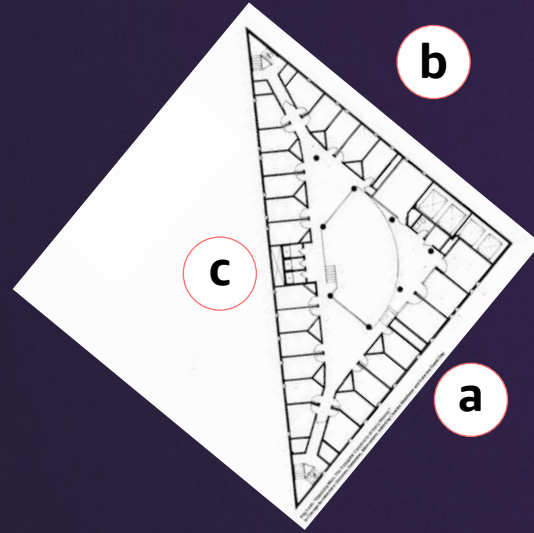
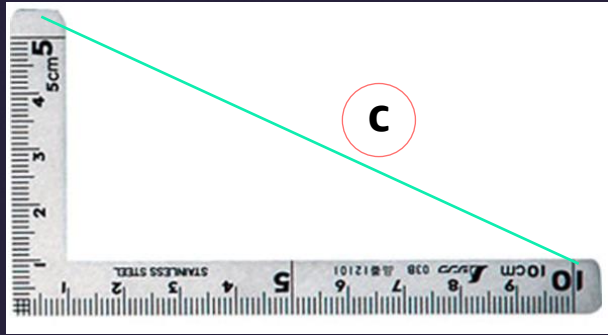
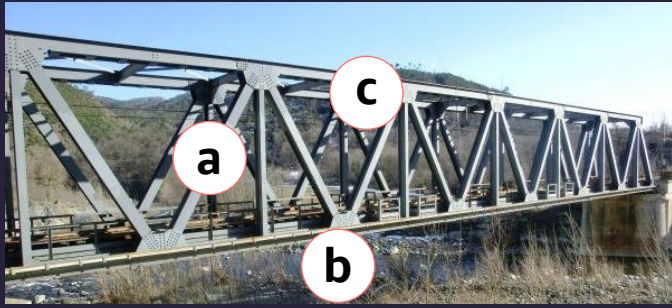
$$a^2 + b^2 = c^2$$



Machine learning takes some data and tries to find a **model** that hopefully will fit the remaining expected data

$$a^2 + b^2 = c^2$$



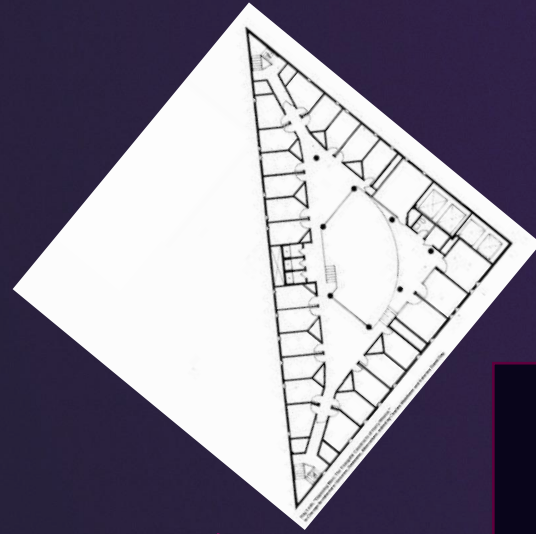
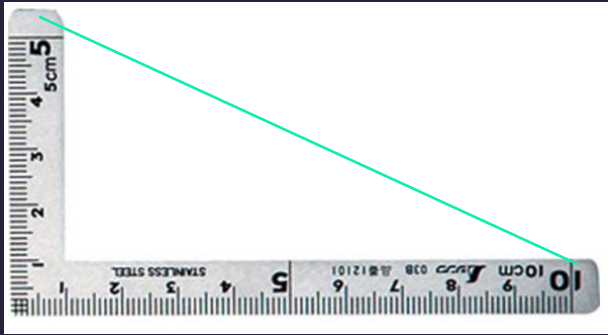
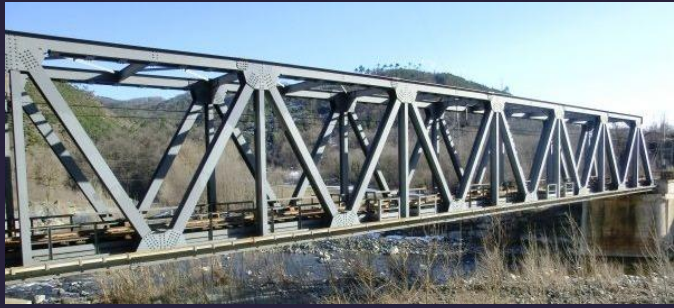


Machine learning takes some data and tries to find a **model** that hopefully will fit the remaining expected data

$$a^2 + b^2 = c^2$$

$$a^2 + b^2 = c^{2.1}$$

Sometimes inaccurate

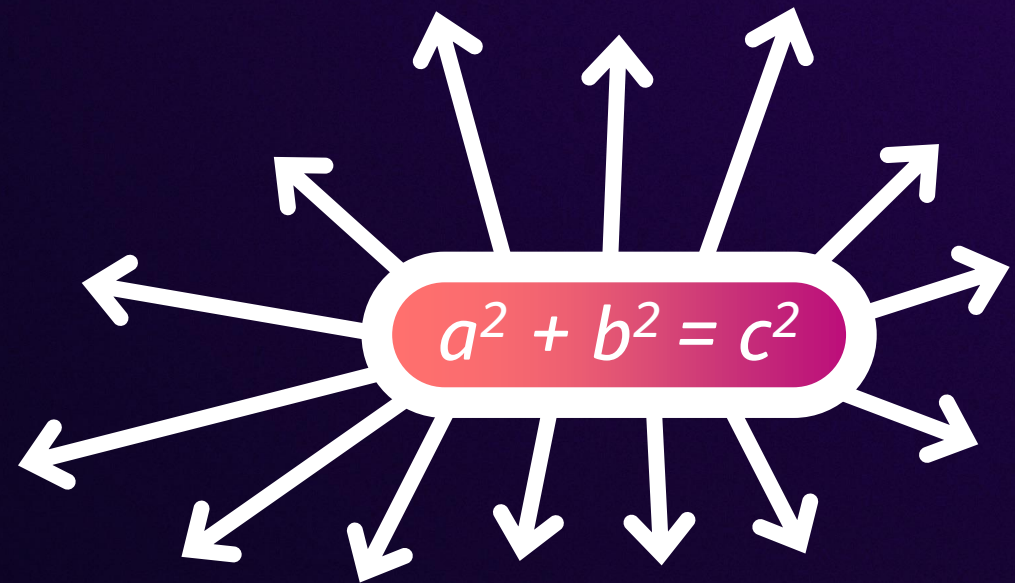


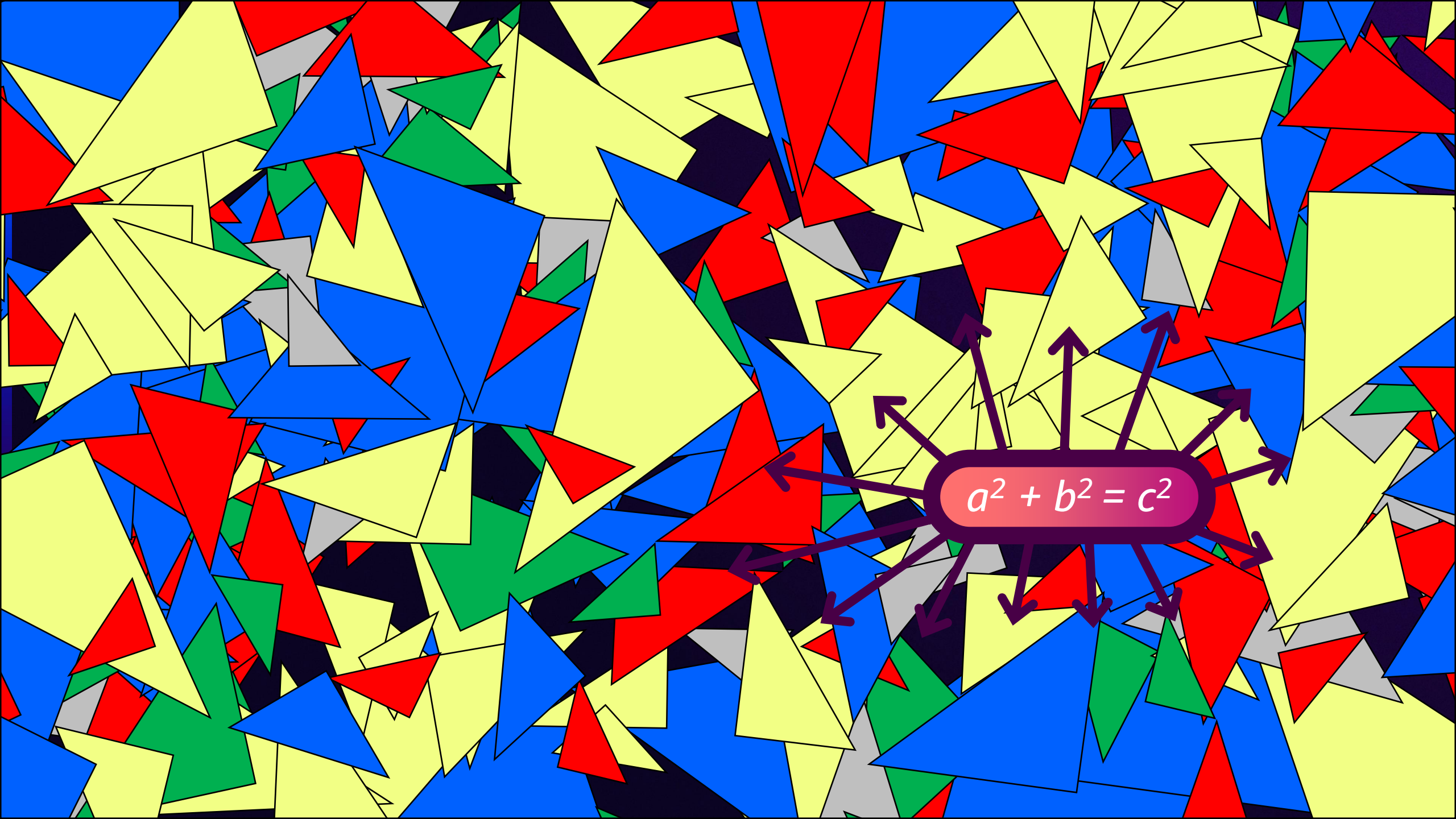
Inference/Testing

Training

$$a^2 + b^2 = c^2$$

# Infinite set





$$a^2 + b^2 = c^2$$

*Reasoning* is the act of talking *accurately* about *all possible* data a model can produce



$a^2 + b^2 = c^2$

*Reasoning* is the act of talking *accurately* about *all possible* data a model can produce

Even when infinite or intractably large


$$a^2 + b^2 = c^2$$

DARTMOUTH COLLEGE  
Department of Mathematics & Astronomy  
HANOVER · NEW HAMPSHIRE

March,  
1956

Mr. Ray Solomonoff  
Technical Research Group  
17 Union Square West  
New York, New York

Dear Ray:

You are one of the people we should like to invite to the "Summer Research Project on Artificial Intelligence."

Terms: \$1,200 - \$900 of which will probably count as a fellowship and be tax free, plus traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H. (a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy } for all 2 months  
A. Newsky }  
John O'Halloran }  
R. Solomonoff }  
Julian Bigelow }

Shannon } some of these for part of time.  
Rochester }  
Selfridge }  
McCulloch }  
Newell }  
Simon }  
McCarthy }  
Et al }

# AI

Connectionist AI

Symbolic AI

Probabilistic AI

"Machine Learning"

"Automated Reasoning"

Neuro-symbolic AI

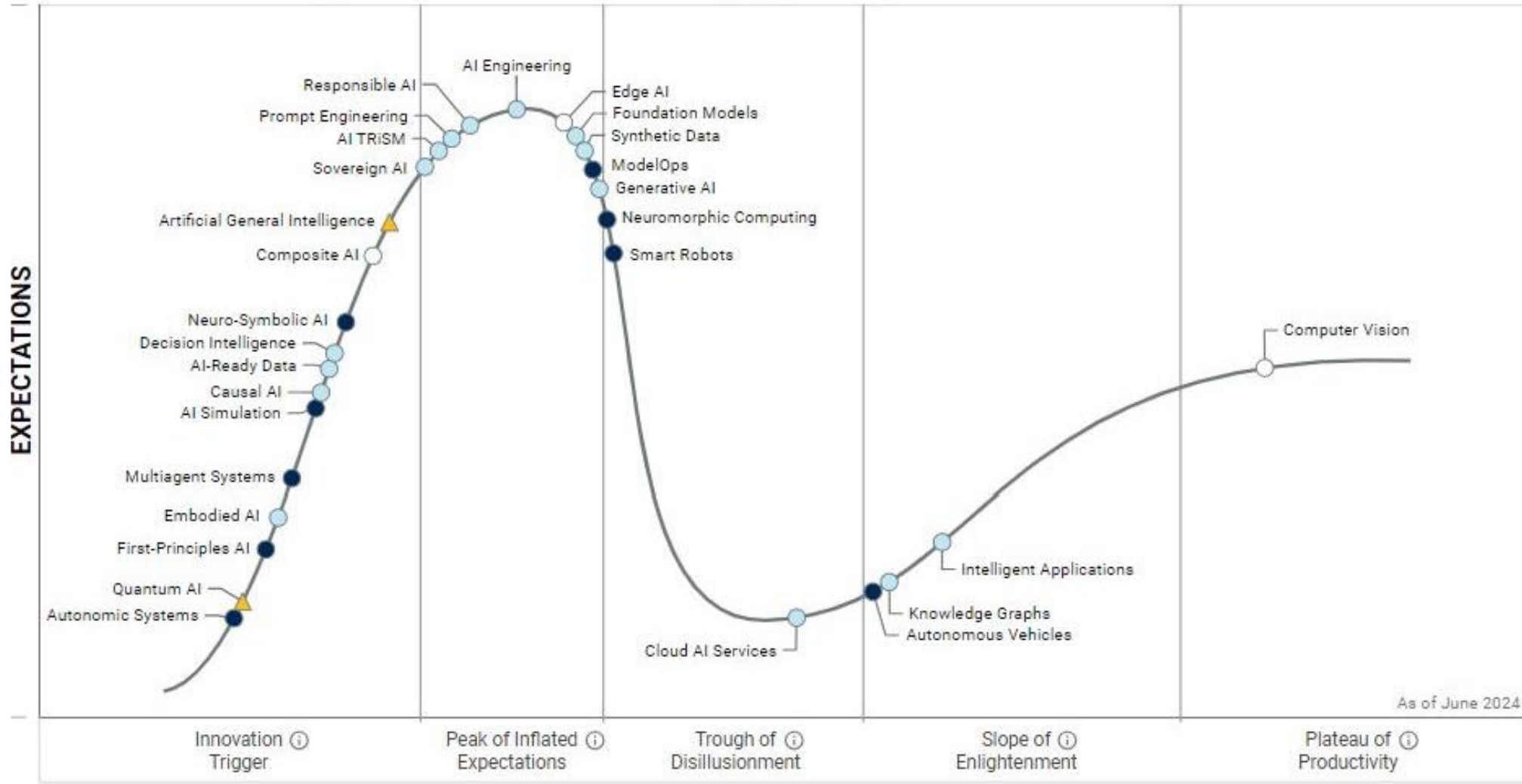
Time To Plateau Will Be Reached:

< 2 yrs.

2-5 yrs.

5-10 yrs.

> 10 yrs.





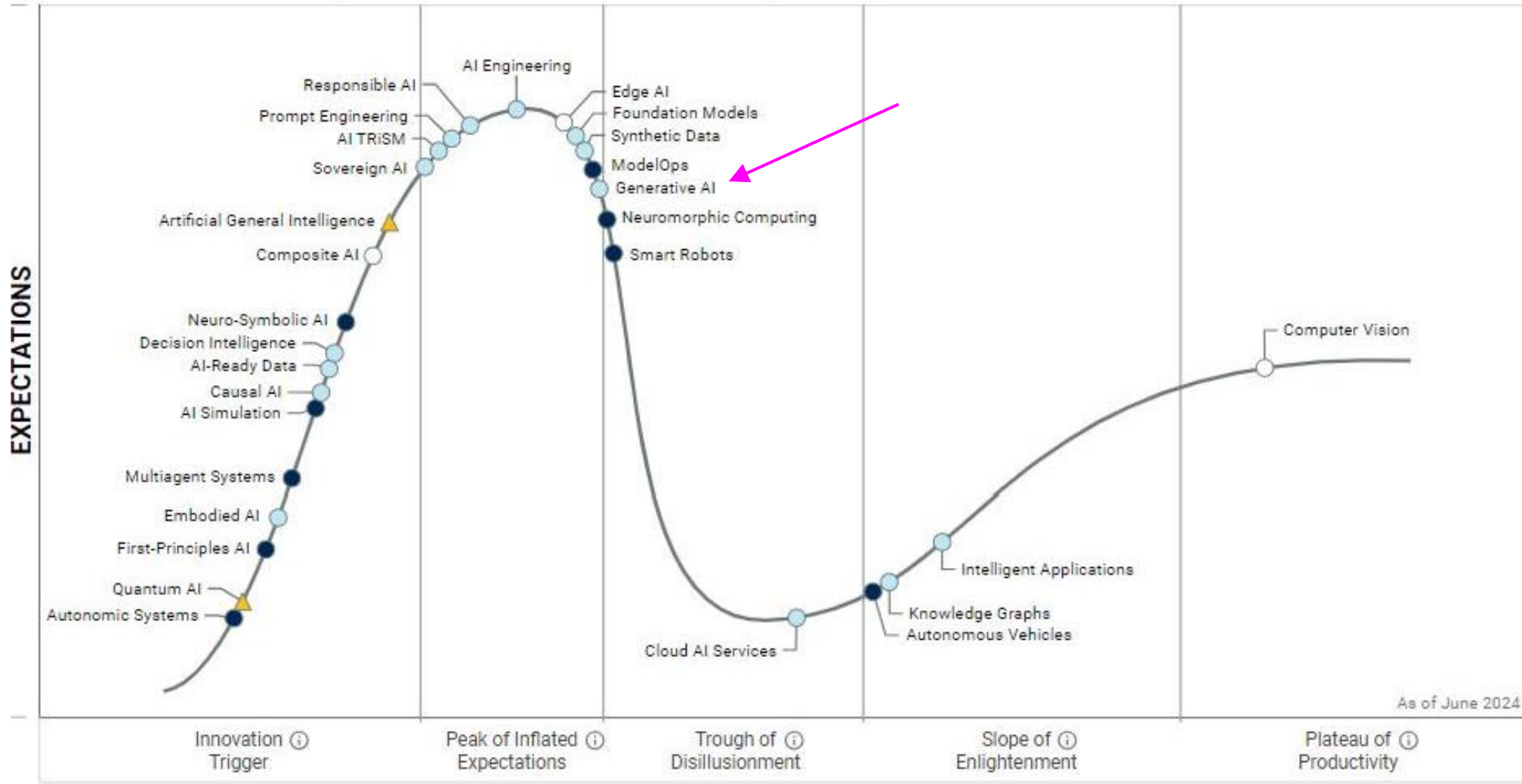
Time To Plateau Will Be Reached:

< 2 yrs.

2-5 yrs.

5-10 yrs.

> 10 yrs.



As of June 2024



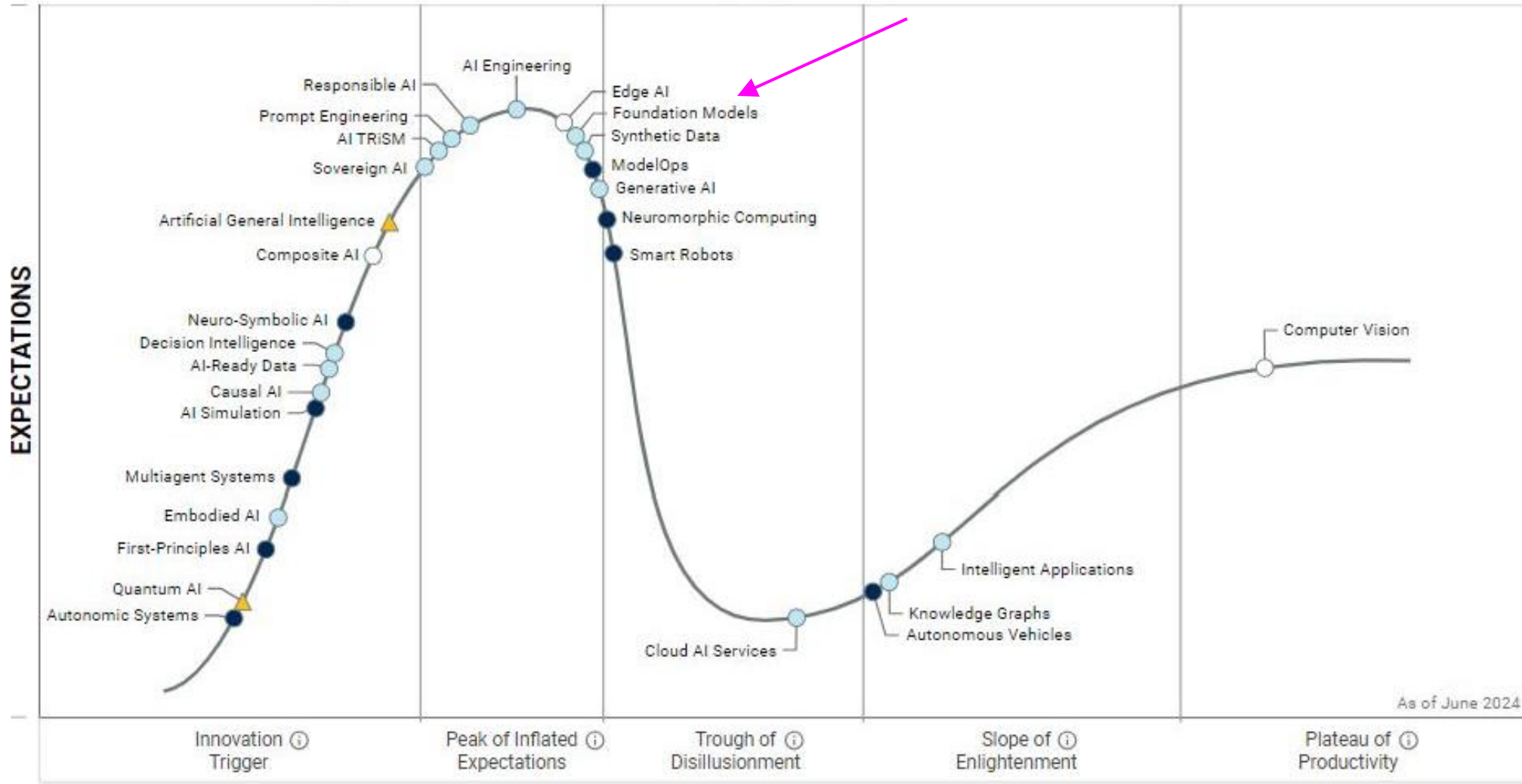
Time To Plateau Will Be Reached:

< 2 yrs.

2-5 yrs.

5-10 yrs.

> 10 yrs.



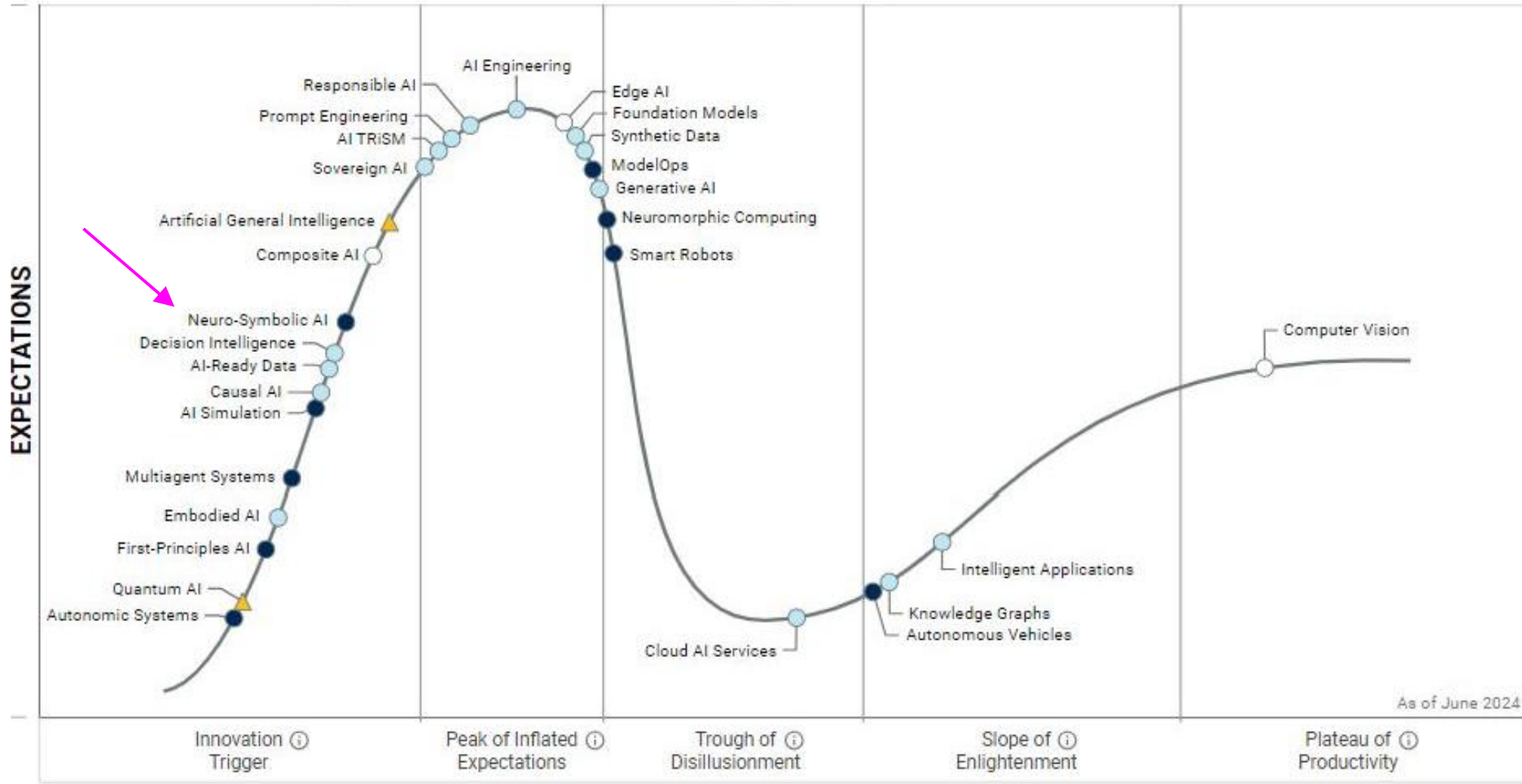
Time To Plateau Will Be Reached:

< 2 yrs.

2-5 yrs.

5-10 yrs.

> 10 yrs.



As of June 2024



# AI

Connectionist AI

Symbolic AI

Probabilistic AI

Let's zoom into a little  
bit more detail

*"Machine Learning"*

*"Automated Reasoning"*

Neuro-symbolic AI

# AI

Symbolic AI

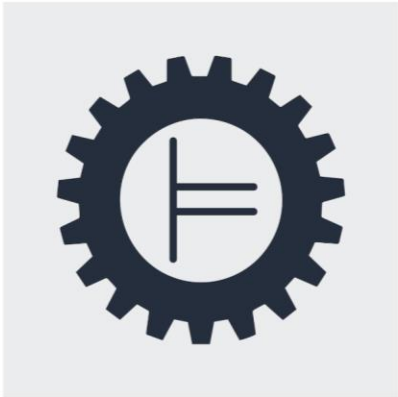
*"Automated Reasoning"*

Neuro-symbolic AI

Let's zoom into a little  
bit more detail

The screenshot shows a web browser window displaying an Amazon Science blog post. The browser's address bar shows the URL `amazon.science/blog/a-gentle-introduction-to-automated-reasoning`. The page header includes the Amazon Science logo and a 'Subscribe' button. The main content area features the title 'A gentle introduction to automated reasoning' under the category 'AUTOMATED REASONING'. The author is listed as 'Byron Cook' and the date as 'December 01, 2021'. A 'Share' button is visible. The text of the post begins with 'This week, Amazon Science added *automated reasoning* to its list of [research areas](#). We made this change because of the impact that automated reasoning is having here at Amazon. For example, Amazon Web Services' customers now have direct access to automated-reasoning-based features such as [IAM Access Analyzer](#), [S3 Block Public Access](#), or [VPC Reachability Analyzer](#). We also see Amazon development teams [integrating automated-reasoning tools](#) into their development processes, raising the bar on the security, durability, availability, and quality of...

**Code-oriented example from AWS blog post**



```
assert isinstance(x,int) and isinstance(y,int)
```

```
if y > 0:  
    while x > y:  
        x = x - y
```

```
assert isinstance(x,int) and isinstance(y,int)
```

```
if y > 0:
```

```
    while x > y:  
        x = x - y
```

Could this run infinitely?



```
assert isinstance(x,int) and isinstance(y,int)
```

```
if y > 0:
```

```
    while x > y:  
        x = x - y
```

Could this run infinitely?

**Assume the processor and interpreter are operating correctly**

```
if y > 0:  
    while x > y:  
        x = x - y
```

Will eventually fail (no matter the initial values of x and y)

```
if y > 0:  
    while x > y:  
        x = x - y
```

Will eventually fail (no matter the initial values of x and y)

Reason: x-y is decreasing



```
if y > 0:  
    while x > y:  
        x = x - y
```

Reason: x-y is decreasing

```
if y > 0:  
    while x > y:  
        x = x - y
```

Reason: y is positive and constant

Reason: x-y is decreasing

```
if y > 0:  
    while x > y:  
        x = x - y
```

Reason: y is positive and constant

```
if y > 0:
```

```
    while x > y:
```

```
        x = x - y
```

Reason #1: y is positive at loop entry

Reason #2: y is not modified in the loop body

```
if y > 0:  
    while x > y:  
        x = x - y
```

Thus: loop always terminates for all values of x and y



```
if y > 0:  
    while x > y:  
        x = x - y
```

Thus: loop always terminates for all values of x and y

Termination cannot be exhaustively tested

a.k.a. *symbolic AI*

**What *is* automated reasoning?**

a.k.a. *symbolic AI*

# What *is* automated reasoning?

Let's zoom even  
deeper into detail

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

Let's zoom into  
even more detail

The *semantics* of the loop in  
mathematical logic / set theory

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

We transition  
between two states

$$R = \{ (s, t) \mid [ (s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x)) ] \wedge \\
 [ (s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x)) ] \wedge \\
 [ (s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y)) ] \wedge \\
 [ (s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x)) ] \}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

The case where we enter the loop

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

In the if  
statement

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \\
 & \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

Going into  
the if branch

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$



```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

Post state is in the loop

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \\
 & \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

y is unchanged

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

x is unchanged

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

The case where we don't enter the loop

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

We skip the loop

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

y is unchanged

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

x is unchanged

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

The case where we  
stay in the loop

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$



```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

We stay in the loop

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) \neq 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \\
 & \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

y is unchanged

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

$x = x - y$

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & \}
 \end{aligned}$$

```

1: if y > 0:
2:     while x > y:
3:         x = x - y
4:

```

The case where we  
exit the loop

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```
if y > 0:
  while x > y:
    x = x - y
```

Text

Meaning

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\ [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}$$

```

if y > 0:
  while x > y:
    x = x - y

```

Could this run infinitely?

=

Is this relation *well-founded*?

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

$R(a, b)$

Is this relation *well-founded*?

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\ [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}$$

Entering loop

$$R(a, b) \wedge a(p) = 1$$

Is this relation *well-founded*?

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$



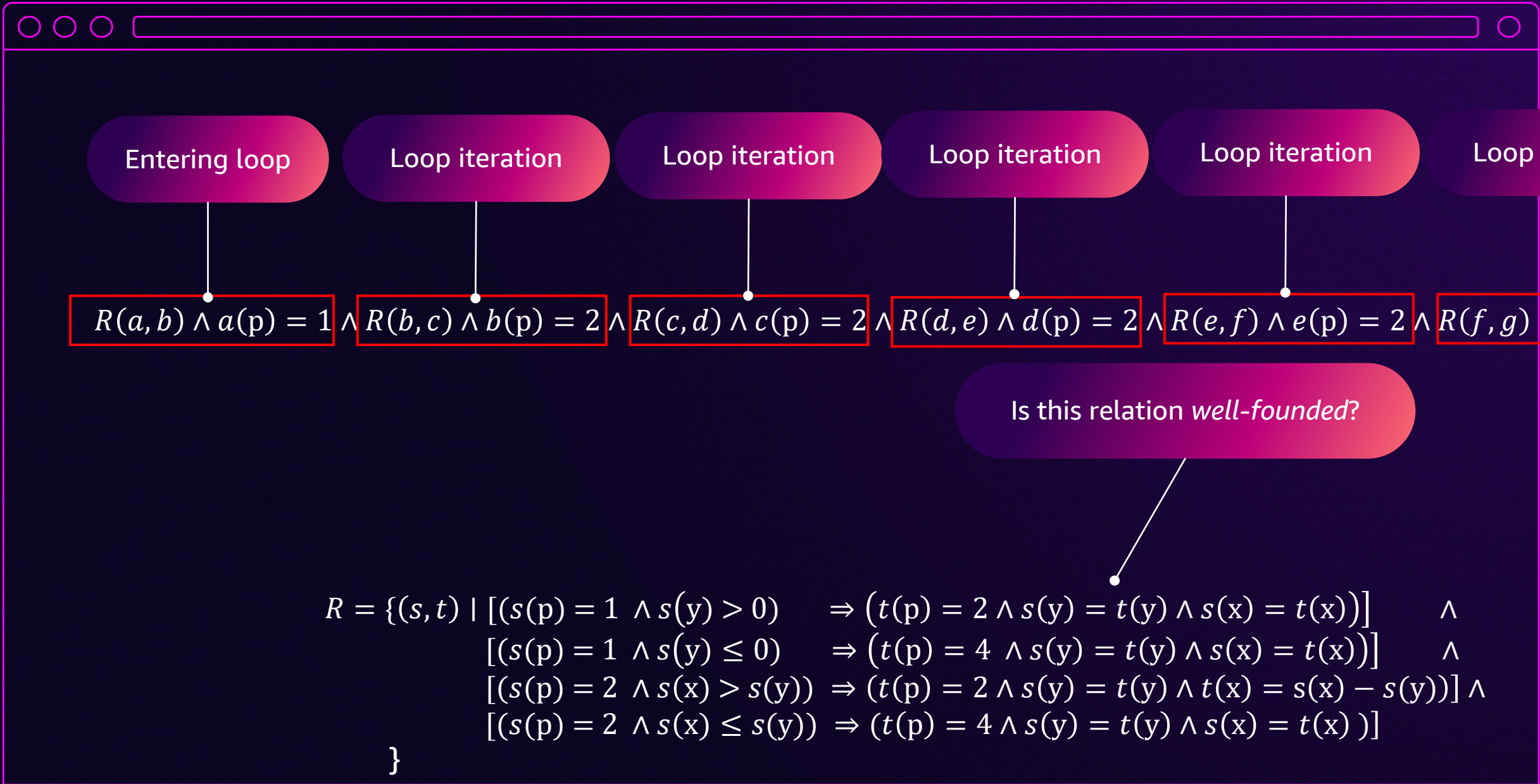
Entering loop

Loop iteration

$$R(a, b) \wedge a(p) = 1 \wedge R(b, c) \wedge b(p) = 2$$

Is this relation *well-founded*?

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$



```
if y > 0:
```

```
  while x > y:  
    x = x - y
```

Will eventually fail

=

Composition will eventually fail

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$

```
if y > 0:
```

```
  while x > y:  
    x = x - y
```

Composition

$$R(a, b) \wedge a(p) = 1 \wedge R(b, c) \wedge b(p) = 2$$

Will eventually fail

=

Composition will eventually fail

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$

```
if y > 0:
  while x > y:
    x = x - y
```

y is positive and constant

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
  
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
  
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
  
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$



```

if y > 0:
  while x > y:
    x = x - y

```

y is positive and constant

Proof by induction

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

```
if y > 0:
```

```
    while x > y:
```

```
        x = x - y
```

Reason #1: y is positive at loop entry

Reason #2: y is not modified in the loop body

```

if y > 0:
    while x > y:
        x = x - y

```

Reason #1:  
Base case

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$



```

if y > 0:
  while x > y:
    x = x - y

```

We are not already  
in the loop

$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

if y > 0:
    while x > y:
        x = x - y

```

We have entered into the loop

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

if y > 0:
  while x > y:
    x = x - y

```

We have taken a step in the program

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

R defined here

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

```

if y > 0:
  while x > y:
    x = x - y

```

and, y>0

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

```
if y > 0:
```

```
    while x > y:
```

```
        x = x - y
```

Reason #1: y is positive at loop entry

Reason #2: y is not modified in the loop body

```

if y > 0:
    while x > y:
        x = x - y

```

Reason #2:  
Step case

$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

```

if y > 0:
  while x > y:
    x = x - y

```

We are already in the loop

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

```

if y > 0:
    while x > y:
        x = x - y

```

We are staying in the loop

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$



```

if y > 0:
  while x > y:
    x = x - y

```

We have taken a step in the program

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

R defined here

$$\begin{aligned}
R = \{ & (s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

```

if y > 0:
  while x > y:
    x = x - y

```

y > 0 implies  
that it *stays*  
y > 0

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

```

if y > 0:
  while x > y:
    x = x - y

```

y is positive *and constant*

Constant

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) = t(y)$$

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$

$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$

$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$

$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$

```

if y > 0:
  while x > y:
    x = x - y

```

Can all be automatically discharged with a mechanical theorem prover

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) = t(y)$$

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

# Symbolic AI

```
if y > 0:  
  while x > y:  
    x = x - y
```

Can all be automatically discharged with a mechanical theorem prover

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) = t(y)$$

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$

```

if y > 0:
  while x > y:
    x = x - y

```

Reasoning accurately  
about infinite sets

$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) = t(y)$

$$\begin{aligned}
 R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
 & [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
 & [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
 \end{aligned}$$

# Symbolic AI

```
if y > 0:  
  while x > y:  
    x = x - y
```

Reasoning accurately  
about infinite sets

$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) = t(y)$

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$

```
if y > 0:
    while x > y:
        x = x - y
```

x-y is decreasing

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$



```

if y > 0:
  while x > y:
    x = x - y

```

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t) \wedge Q(s, t)] \Rightarrow s(x) - s(y) > t(x) - t(y)$$

x-y is decreasing

x-y is decreasing

$$\begin{aligned}
R = \{ & (s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

```

if y > 0:
  while x > y:
    x = x - y

```

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t) \wedge Q(s, t)] \Rightarrow s(x) - s(y) > t(x) - t(y)$$

Q:

$$\forall s, t. [s(p) \neq 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) > 0 \Rightarrow t(y) > 0$$

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t)] \Rightarrow s(y) = t(y)$$

$$\begin{aligned}
R = \{ (s, t) \mid & [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\
& [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\
& [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}
\end{aligned}$$

Can be automatically discharged with a mechanical theorem prover

```
if y > 0:  
  while x > y:  
    x = x - y
```

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t) \wedge Q(s, t)] \Rightarrow s(x) - s(y) > t(x) - t(y)$$

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$



Reasoning accurately about infinite sets

```
if y > 0:
  while x > y:
    x = x - y
```

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t) \wedge Q(s, t)] \Rightarrow s(x) - s(y) > t(x) - t(y)$

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$

# Symbolic AI

Reasoning accurately  
about infinite sets

```
if y > 0:  
    while x > y:  
        x = x - y
```

$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t) \wedge Q(s, t)] \Rightarrow s(x) - s(y) > t(x) - t(y)$

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$

Entering loop

Loop iteration

Loop iteration

Loop iteration

Loop iteration

$R(a, b) \wedge a(p) = 1 \wedge R(b, c) \wedge b(p) = 2 \wedge R(c, d) \wedge c(p) = 2 \wedge R(d, e) \wedge d(p) = 2 \wedge R(e, f) \wedge e(p) = 2 \wedge \dots$

x-y is decreasing

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t) \wedge Q(s, t)] \Rightarrow s(x) - s(y) > t(x) - t(y)$$

$$R(a, b) \wedge a(p) = 1 \wedge R(b, c) \wedge b(p) = 2 \wedge R(c, d) \wedge c(p) = 2 \wedge R(d, e) \wedge d(p) = 2 \wedge R(e, f) \wedge e(p) = 2 \wedge \dots$$

x-y is decreasing

$$\forall s, t. [s(p) = 2 \wedge t(p) = 2 \wedge R(s, t) \wedge Q(s, t)] \Rightarrow s(x) - s(y) > t(x) - t(y)$$

$$R(a, b) \wedge a(p) = 1 \wedge R(b, c) \wedge b(p) = 2 \wedge R(c, d) \wedge c(p) = 2 \wedge R(d, e) \wedge d(p) = 2 \wedge R(e, f) \wedge e(p) = 2 \wedge \dots$$

$$b(y) - b(x) > c(y) - c(x) > d(y) - d(x) > e(y) - e(x) > \dots$$



R is isomorphic to a sub-relation over ( $>$ , p-ints)

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\ [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}$$

R is isomorphic to a sub-relation over  $(>, p\text{-ints})$

$(>, p\text{-ints})$  is well-founded

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\ [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}$$

Every sub-relation of a well-founded relation is well-founded

R is isomorphic to a sub-relation over ( $>$ , p-ints)

( $>$ , p-ints) is well-founded

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\ [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}$$

R is well-founded

Every sub-relation of a well-founded relation is well-founded

R is isomorphic to a sub-relation over  $(>, p\text{-ints})$

$(>, p\text{-ints})$  is well-founded

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge \\ [(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\ [(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \}$$

$R$  is a well-founded relation

$$R = \{(s, t) \mid \begin{aligned} &[(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\ &[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \quad \wedge \\ &[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge \\ &[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \end{aligned}\}$$

```
if y > 0:
    while x > y:
        x = x - y
```

The code *does* guarantee termination

R *is* a well-founded relation

$$R = \{(s, t) \mid [(s(p) = 1 \wedge s(y) > 0) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 1 \wedge s(y) \leq 0) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))] \wedge$$
$$[(s(p) = 2 \wedge s(x) > s(y)) \Rightarrow (t(p) = 2 \wedge s(y) = t(y) \wedge t(x) = s(x) - s(y))] \wedge$$
$$[(s(p) = 2 \wedge s(x) \leq s(y)) \Rightarrow (t(p) = 4 \wedge s(y) = t(y) \wedge s(x) = t(x))]\}$$



a.k.a. *symbolic AI*

**What *is* automated reasoning?**

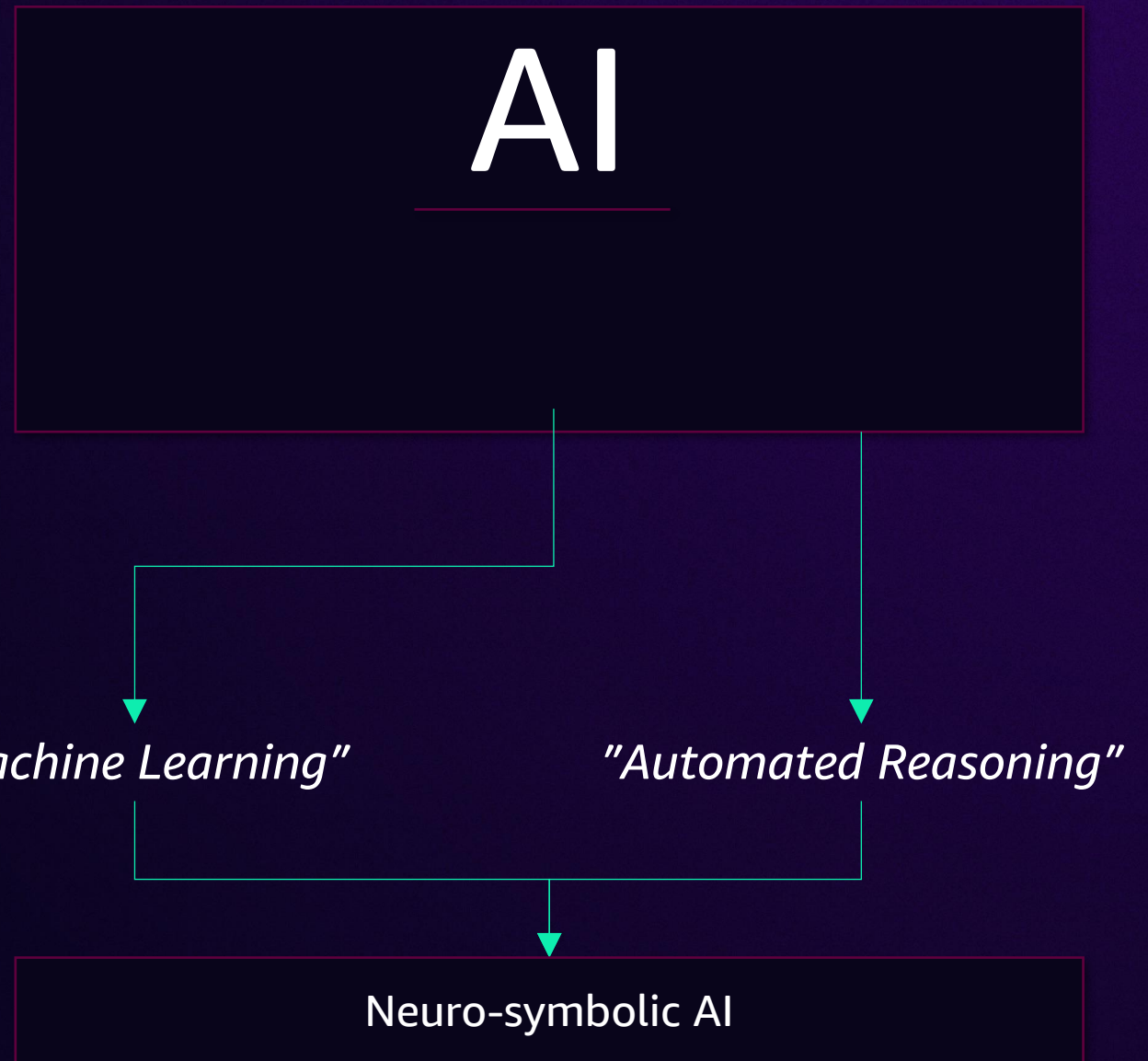
a.k.a. *symbolic AI*

# What *is* automated reasoning?

Let's look at some examples



AWS has been heavily invested in both Automated Reasoning and Machine Learning for 10+ years



## Add rule

Add rules to define the desired configuration settings of your AWS resources. For a custom rule, you must create an AWS Lambda function for the rule.

[Add custom rule](#)

ec2

<< < Viewing 1 - 9 of 18 AWS managed rules > >>

### approved-amis-by-id

Checks whether running instances are using specified AMIs. Specify a list of approved AMI IDs. Running instances with AMIs that are not on this list are noncompliant.

EC2

### approved-amis-by-tag

Checks whether running instances are using specified AMIs. Specify the tags that identify the AMIs. Running instances with AMIs that don't have at least one of the specified tags

EC2

### cloudwatch-alarm-resource-check New

Checks whether the specified resource type has a CloudWatch alarm for the specified metric. For resource type, you can specify EBS volumes, EC2 instances, RDS clusters,

CloudWatch

### desired-instance-tenancy

Checks instances for specified tenancy. Specify AMI IDs to check instances that are launched from those AMIs or specify Host IDs to check whether instances are launched on

EC2

### desired-instance-type

Checks whether your EC2 instances are of the specified instance types.

EC2

### ebs-optimized-instance

Checks whether EBS optimization is enabled for your EC2 instances that can be EBS-optimized.

EC2

### ec2-instance-detailed-monitoring-ena...

Checks whether detailed monitoring is enabled for EC2 instances.

EC2

### ec2-instances-in-vpc

Checks whether your EC2 instances belong to a virtual private cloud (VPC). Optionally, you can specify the VPC ID to associate with your instances.

EC2

### ec2-managedinstance-applications-bl...

Checks that none of the specified applications are installed on the instance. Optionally, specify the version. Newer versions will not be blacklisted. Optionally,

Systems Manager

# Managed Config Rules

# Managed Config Rules

Rules > Add rule

## Add rule

Add rules to define the desired configuration settings of your AWS resources. For a custom rule, you must create an AWS Lambda function for the rule.

[Add custom rule](#)

ec2 < < Viewing 1 - 9 of 18 AWS managed rules > >

**approved-amis-by-id**

Checks whether running instances are using specified AMIs. Specify a list of approved AMI IDs. Running instances with AMIs that are not on this list are noncompliant.

EC2

**approved-amis-by-tag**

Checks whether running instances are using specified AMIs. Specify the tags that identify the AMIs. Running instances with AMIs that don't have at least one of the specified tags

EC2

**cloudwatch-alarm-resource-check** New

Checks whether the specified resource type has a CloudWatch alarm for the specified metric. For resource type, you can specify EBS volumes, EC2 instances, RDS clusters.

CloudWatch

**desired-instance-tenancy**

Checks instances for specified tenancy. Specify AMI IDs to check instances that are launched from those AMIs or specify Host IDs to check whether instances are launched on

EC2

**desired-instance-type**

Checks whether your EC2 instances are of the specified instance types.

EC2

**ebs-optimized-instance**

Checks whether EBS optimization is enabled for your EC2 instances that can be EBS-optimized.

EC2

**ec2-instance-detailed-monitoring-ena...**

Checks whether detailed monitoring is enabled for EC2 instances.

EC2

**ec2-instances-in-vpc**

Checks whether your EC2 instances belong to a virtual private cloud (VPC). Optionally, you can specify the VPC ID to associate with your instances.

EC2

**ec2-managedinstance-applications-bl...**

Checks that none of the specified applications are installed on the instance. Optionally, specify the version. Newer versions will not be blacklisted. Optionally,

Systems Manager

# Inspector

## Assessment Target - My assessment target

**Name\*** My assessment target

**All Instances**  Include all EC2 instances in this AWS account and region.

Note: The limit on the maximum number of agents that can be included in an assessment run applies. [Learn more](#)

**Use Tags\***

| Key                           | Value |
|-------------------------------|-------|
| <a href="#">Add a new key</a> |       |

**Install Agents**  Install the Amazon Inspector Agent on all EC2 instances in this assessment target.

To use this option, make sure that your EC2 instances have the SSM Agent installed and an IAM role that allows Run Command. [Learn more](#)

**\*Required**

[Save](#) [Cancel](#) [Preview](#)

aws Services ▾

**New VPC Experience**  
Tell us what you think

VPC Dashboard **New**

Filter by VPC:

- ▶ VIRTUAL PRIVATE CLOUD
- ▶ SECURITY
- ▼ REACHABILITY  
Reachability Analyzer

**VPC Reachability Analyzer**

Rules > Add rule

Add rule

Add rules to define the desired configuration settings of your AWS resources. For a custom rule, you must create an AWS Lambda function for the rule.

**Managed Config Rules**

Add custom rule

ec2 << Viewing 1 - 9 of 18 AWS managed rules >>

|  |   |  |
|--|---|--|
| <p><b>approved-amis-by-id</b></p> <p>Checks whether running instances are using specified AMIs. Specify a list of approved AMI IDs. Running instances with AMIs that are not on this list are noncompliant.</p> <p>EC2</p>           | <p><b>approved-amis-by-tag</b></p> <p>Checks whether running instances are using specified AMIs. Specify the tags that identify the AMIs. Running instances with AMIs that don't have at least one of the specified tags</p> <p>EC2</p> | <p><b>cloudwatch-alarm-resource-check</b> <b>New</b></p> <p>Checks whether the specified resource type has a CloudWatch alarm for the specified metric. For resource type, you can specify EBS volumes, EC2 instances, RDS clusters.</p> <p>CloudWatch</p> |
| <p><b>desired-instance-tenancy</b></p> <p>Checks instances for specified tenancy. Specify AMI IDs to check instances that are launched from those AMIs or specify Host IDs to check whether instances are launched on</p> <p>EC2</p> | <p><b>desired-instance-type</b></p> <p>Checks whether your EC2 instances are of the specified instance types.</p> <p>EC2</p>  | <p><b>ebs-optimized-instance</b></p> <p>Checks whether EBS optimization is enabled for your EC2 instances that can be EBS-optimized.</p> <p>EC2</p>  |
| <p><b>ec2-instance-detailed-monitoring-ena...</b></p> <p>Checks whether detailed monitoring is enabled for EC2 instances.</p> <p>EC2</p>   | <p><b>ec2-instances-in-vpc</b></p> <p>Checks whether your EC2 instances belong to a virtual private cloud (VPC). Optionally, you can specify the VPC ID to associate with your instances.</p> <p>EC2</p>                                | <p><b>ec2-managedinstance-applications-bl...</b></p> <p>Checks that none of the specified applications are installed on the instance. Optionally, specify the version. Newer versions will not be blacklisted. Optionally,</p> <p>Systems Manager</p>      |

**Managed Config Rules**

**Inspector**

Assessment Target - My assessment target

**Name\*** My assessment target

**All Instances**  Include all EC2 instances in this AWS account and region.  
 Note: The limit on the maximum number of agents that can be included in an assessment run applies. [Learn more](#)

**Use Tags\***

| Key  | Value                |
|--|----------------------|
| <input type="text" value="Add a new key"/> | <input type="text"/> |

**Install Agents**  Install the Amazon Inspector Agent on all EC2 instances in this assessment target.  
 To use this option, make sure that your EC2 instances have the SSM Agent installed and an IAM role that allows Run Command. [Learn more](#)

**\*Required**

aws Services ▾

**New VPC Experience**  
Tell us what you think

VPC Dashboard **New**

Filter by VPC:  
Select a VPC

- VIRTUAL PRIVATE CLOUD
- SECURITY
- REACHABILITY  
Reachability Analyzer

## VPC Reachability Analyzer

Rules > Add rule

Add rule

Add rules to define the desired configuration settings of your AWS custom rule, you must create an AWS Lambda function for the rule.

+ Add custom rule

ec2 < Viewing 1 - 9 of 18 AWS managed rules >

|   |  |  |
|---|--|--|
| <b>approved-amis-by-id</b><br>Checks whether running instances are using specified AMIs. Specify a list of approved AMI IDs. Running instances with AMIs that are not on this list are noncompliant.<br>EC2           | <b>approved-amis-by-tag</b><br>Checks whether running instances are using specified AMIs. Specify the tags that identify the AMIs. Running instances with AMIs that don't have at least one of the specified tags<br>EC2 | <b>cloudwatch-alarm-resource-check</b> <b>New</b><br>Checks whether the specified resource type has a CloudWatch alarm for the specified metric. For resource type, you can specify EBS volumes, EC2 instances, RDS clusters, CloudWatch<br>CloudWatch |
| <b>desired-instance-tenancy</b><br>Checks instances for specified tenancy. Specify AMI IDs to check instances that are launched from those AMIs or specify Host IDs to check whether instances are launched on<br>EC2 | <b>desired-instance-type</b><br>Checks whether your EC2 instances are of the specified instance types.<br>EC2  | <b>ebs-optimized-instance</b><br>Checks whether EBS optimization is enabled for your EC2 instances that can be EBS-optimized.<br>EC2   |
| <b>ec2-instance-detailed-monitoring-ena...</b><br>Checks whether detailed monitoring is enabled for EC2 instances.<br>EC2   | <b>ec2-instances-in-vpc</b><br>Checks whether your EC2 instances belong to a virtual private cloud (VPC). Optionally, you can specify the VPC ID to associate with your instances.<br>EC2                                | <b>ec2-managedinstance-applications-bl...</b><br>Checks that none of the specified applications are installed on the instance. Optionally, specify the version. Newer versions will not be blacklisted. Optionally,<br>Systems Manager                 |

## Managed Config Rules

Block public access (account settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, or both. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply account-wide for all current and future buckets. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

**Block all public access** Cancel Save

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting does not affect existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any public access control lists (ACLs)**  
S3 will ignore all ACLs that grant public access to buckets or objects and instead use the permissions on the bucket policy to determine if public access is allowed.
- Block public access to buckets and objects granted through any bucket policies**  
S3 will block new bucket policies that allow public access to buckets or objects and instead use the permissions on the bucket policy to determine if public access is allowed. This setting does not affect any existing policies.
- Block public and cross-account access to buckets and objects through any public bucket policies**  
S3 will ignore public and cross-account access for buckets with policies that grant public access to buckets and objects.

## S3 Block Public Access

Assessment Target - My assessment target

Name\* My assessment target

All Instances  Include all EC2 instances in this AWS account and region.  
Note: The limit on the maximum number of agents that can be included in an assessment run applies. [Learn more](#)

Use Tags\* 

| Key           | Value |
|---------------|-------|
| Add a new key |       |

Install Agents  Install the Amazon Inspector Agent on all EC2 instances in this assessment target.  
To use this option, make sure that your EC2 instances have the SSM Agent installed and an IAM role that allows Run Command. [Learn more](#)

\*Required Save Cancel Preview

## Inspector

aws Services

New VPC Experience  
Tell us what you think

VPC Dashboard **New**

Filter by VPC:  
Select a VPC

- VIRTUAL PRIVATE CLOUD
- SECURITY
- REACHABILITY  
Reachability Analyzer

Rules > Add rule

Add rule

Add rules to define the desired configuration settings of your AWS custom rule, you must create an AWS Lambda function for the rule.

+ Add custom rule

ec2

Viewing 1 - 9 of 18 AWS managed rules

- approved-amis-by-id
- approved-amis-by-tag
- cloudwatch-alarm-resource-check **New**
- desired-instance-tenancy
- desired-instance-type
- ebs-optimized-instance
- ec2-instance-detailed-monitoring-ena...
- ec2-instances-in-vpc
- ec2-managedinstance-applications-bl...

**Managed Config Rules**

Block public access (account settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, or both. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply account-wide for all current and future buckets. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)
- Block public access to buckets and objects granted through any public access ACLs
- Block public access to buckets and objects granted through any bucket policies
- Block public and cross-account access to buckets and objects through any public bucket policies

Cancel Save

**S3 Block Public Access**

**VPC Reachability Analyzer**

**Inspector**

aws Services Resource Groups

IAM > Access Analyzer > Create analyzer

Create analyzer **Info**

The analyzer scans the resources within the zone of the...

Region  
US East (N. Virginia)

You should enable Access Analyzer in each Region where you use AWS resources.

**IAM Access Analyzer**

Assessment Target - My assessment target

Name\* My assessment target

All Instances  Include all EC2 instances in this AWS account and region.

Note: The limit on the maximum number of agents that can be included in an assessment run applies. [Learn more](#)

Use Tags\* 

| Key           | Value |
|---------------|-------|
| Add a new key |       |

Install Agents  Install the Amazon Inspector Agent on all EC2 instances in this assessment target.

To use this option, make sure that your EC2 instances have the SSM Agent installed and an IAM role that allows Run Command. [Learn more](#)

\*Required

Save Cancel Preview

## Open source foundations

The screenshot shows a GitHub repository page for `aws/aws-encryption-sdk-dafny`. The repository name is displayed in a large font, with the AWS logo to its right. Below the name, the description reads "AWS Encryption SDK for Dafny".

Below the description, there are statistics for the repository:

- Contributors: 11
- Issues: 6
- Stars: 16
- Forks: 3

At the bottom of the repository page, there is a green bar with the text "github.com" and a yellow bar with the text "GitHub - aws/aws-encryption-sdk-dafny: AWS Encryption SDK for Da...". Below these bars, there is a dark blue bar with the text "AWS Encryption SDK for Dafny. Contribute to aws/aws-encryption-sdk-dafny development by creating an account on GitHub."

At the bottom of the screenshot, there are icons for a comment bubble, a refresh icon with the number "3", a heart icon with the number "18", and an upload icon.

aws Clean Rooms Differential

docs.aws.amazon.com/clean-rooms/latest/userguide/differential-pri... Relaunch to update

aws Search in this guide Contact Us English Create an AWS Account

AWS > Documentation > AWS Clean Rooms > User Guide Feedback Preferences

## Differential privacy

Differential privacy allows only aggregated insights and obfuscates the contribution of any individual's data in those insights. Differential privacy protects the collaboration data from the member who can receive results learning about a specific individual. Without differential privacy, the member who can receive results can attempt to infer individual user data by adding or removing records about an individual and observing the difference in query results.

When differential privacy is turned on, a specified amount of noise is added to the query results to obfuscate the contribution of individual users. If the member who can receive results tries to observe the difference in query results after removing records about an individual from their dataset, the variability in the query result helps prevent the identification of the individual's data. AWS Clean Rooms Differential Privacy uses the [SampCert](#) sampler, a proven correct sampler implementation developed by AWS.

Feedback icons: thumbs up, thumbs down, and a hexagonal icon with a checkmark.

Open source foundations



Open source  
foundations

The screenshot shows a web browser window displaying a blog post from Amazon Science. The URL in the address bar is [amazon.science/blog/formal-verification-makes-rsa-faster-and-faster-to-deploy](https://amazon.science/blog/formal-verification-makes-rsa-faster-and-faster-to-deploy). The page features a navigation menu with links for Research areas, Blog, Publications, Conferences, Code and datasets, Academia, and Careers, along with a Feedback button. The main content area has a decorative geometric pattern on the left and the following text:

AUTOMATED REASONING

## Formal verification makes RSA faster — and faster to deploy

Optimizations for Amazon's Graviton2 chip boost efficiency, and formal verification shortens development time.

By [June Lee](#), [Hanno Becker](#), [John Harrison](#) [Share](#)

August 08, 2024

Most secure transactions online are protected by public-key encryption schemes like RSA, whose security depends on the difficulty of factoring large numbers. Public-key encryption improves security because it enables the encrypted exchange of private keys. But because it depends on operations like modular exponentiation of large integers, it introduces significant computational overhead.

computer implementation developed by AWS.

Open source  
foundations

# AWS product categories



Analytics



Application Integration



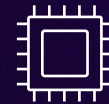
Blockchain



Business Applications



Cloud Financial Management



Compute



Containers



Customer Engagement



Database



Developer Tools



End User Computing



Front-End Web & Mobile



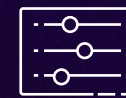
Game Tech



Internet of Things



Machine Learning



Management & Governance



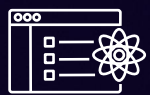
Media Services



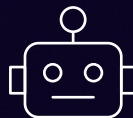
Migration & Transfer



Networking & Content Delivery



Quantum Technologies



Robotics



Satellite



Security, Identity & Compliance



Serverless



Storage



VR & AR



# AWS product categories



Analytic



Application Integration



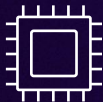
Blockchain



Business Applications



Cloud Financial Management



Compute



Containers



Customer Engagement



Database



Developer Tools



End User Computing



Front-End Web & Mobile



Game Tech



Internet of Things



Machine Learning



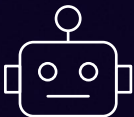
Managed Services



Networking & Content Delivery



Quantum Technologies



Robotics



Satellite



Security, Identity & Compliance



Serverless



Specialized Services

Practically every area touched by automated reasoning in some way



# AWS product categories



Analytics



Application Integration



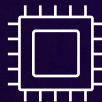
Blockchain



Business Applications



Cloud Financial Management



Compute



Containers



Customer Engagement



Database



Developer Tools



End User Computing



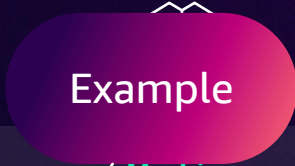
Front-End Web & Mobile



Game Tech



Internet of Things



Machine Learning



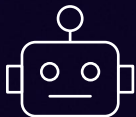
Managed Gov



Networking & Content Delivery



Quantum Technologies



Robotics



Satellite



Security, Identity & Compliance



Serverless



S

Practically every area touched by automated reasoning in some way



Browser tabs: (304) AWS re:Inforce 2024 - | x +

Address bar: youtube.com/watch?v=oshxAJGrwMU

YouTube GB Search [Microphone]

Video player:  
IAM401  
**Proving the correctness of AWS authorization**  
Lucas Wagner (he/him) - Principal Applied Scientist, Amazon Web Services  
Sean McLaughlin (he/him) - Principal Applied Scientist, Amazon Web Services  
aws logo

Video controls: Play, 0:01 / 58:37, Pause, CC, Settings, Full Screen, Maximize

Video title: **AWS re:Inforce 2024 - Proving the correctness of AWS authorization (IAM401)**

Channel: **AWS Events** (verified), 121K subscribers, [Subscribe](#)

Engagement: 63 likes, [Share](#), [More](#)

Metadata: 3.4K views, 3 months ago, #AmazonWebServices #AWS #CloudComputing



(304) AWS re:Inforce 2024 - | x +

youtube.com/watch?v=oshxAJGrwMU

YouTube GB Search

IAM401

# Proving the correctness of AWS authorization

Lucas Wagner (he/him) Principal Applied Scientist Amazon Web Services

Sean McLaughlin (he/him) Principal Applied Scientist Amazon Web Services

aws

0:01 / 58:37

## AWS re:Inforce 2024 - Proving the correctness of AWS authorization (IAM401)

aws AWS Events 121K subscribers [Subscribe](#) 63 [Share](#) ...

3.4K views 3 months ago #AmazonWebServices #AWS #CloudComputing

**AWS's front door**

**Called ~2 billion times per second**



aws Services Resource Groups Follow me on t:@bwest N. Virginia Support

IAM > Access Analyzer > Create analyzer

## Create analyzer [Info](#)

The analyzer scans the resources within the zone of trust.

Region  
US East (N. Virginia)  
You should enable Access Analyzer in each Region where you use AWS resources.

Name  
  
Maximum 255 characters

Zone of trust [Info](#)  
Policies for all supported resources within your zone of trust are analyzed to identify access allowed from outside the zone of trust.  
Current account (796744228948)

Tags [Info](#)  
Optionally, add tags to the analyzer. Tags are words or phrases that act as metadata for identifying and organizing your AWS resources. Each tag consists of a key and one optional value.

No tags associated with the resource.

You can add up to 50 tags.

ⓘ When you enable Access Analyzer, a service-linked role is created in the current account. The service-linked role grants permission to Access Analyzer to interact with AWS resources on your behalf. [Learn more](#)

*“For all possible authorization requests, deny always take precedence over allow”*

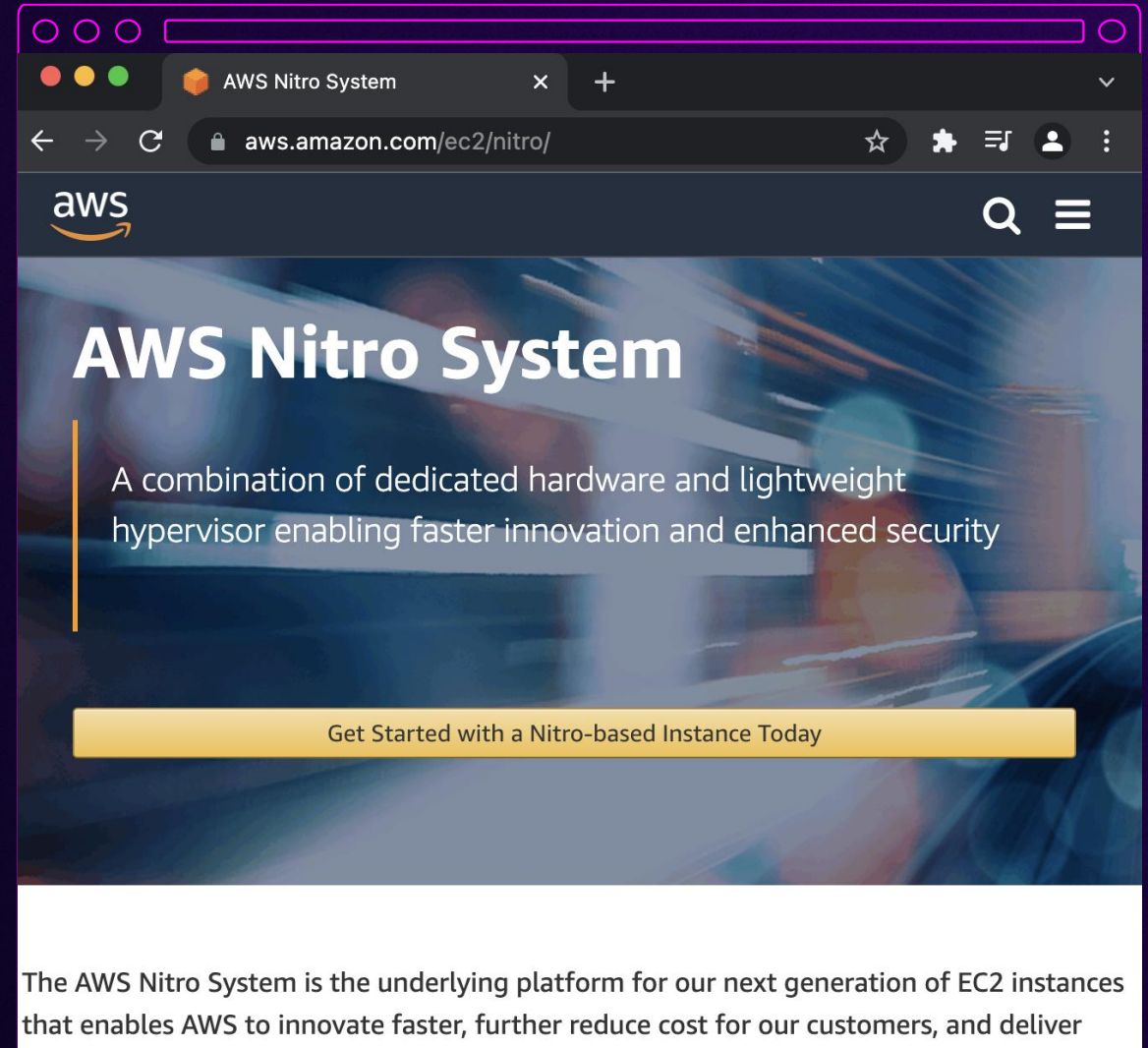
```
src > main > dafny > yucca > evaluate > Statement.dfy
1 |> // includes--
17 module Statement {
18 |> // imports--
35 > export--
49 method matches(s: StatementBlock.T, principal: Principal, r: engine.EvaluationRequest, varia
50 returns (sres: Result.T<bool>)
51 requires engine.Policy.ValidProviders(s)
52 ensures sres == Spec.Matches(s, principal, r.request, variablesEnabled)
53 {
54   var pm := principalMatches(s, principal);
55   if !pm {
56     return Success(false);
57   }
58   I
59   var am := actionBlockMatches(s.getActionBlock(), r);
60   if !am {
61     return Success(false);
62   }
63
64   var rm := resourceBlockMatches(s.getResourceBlock(), r, variablesEnabled);
65   if !rm {
66     return Success(false);
67   }
68
69   sres := conditionBlockMatches(s.getConditionBlock(), r, variablesEnabled);
70 }
```

mainline\* 0 0 Verification Succeeded "Statement.dfy" 221L 7238C written Spaces: 2 UTF-8 LF Dafny DafnyLS: 3.2.0.30713

*Additional AWS systems rebuilt using formal reasoning*



The screenshot shows a web browser window with the URL `aws.amazon.com/s3/consistency/`. The page title is "Amazon S3 Strong Consistency". The breadcrumb navigation is "Products / Storage / Amazon S3 / Amazon S3 Features / ...". The main heading is "Amazon S3 Strong Consistency". The text describes that Amazon S3 delivers strong read-after-write consistency automatically for all applications, without changes to performance or availability, without sacrificing regional isolation for applications, and at no additional cost. It also mentions that with strong consistency, S3 simplifies the migration of on-premises analytics workloads by removing the need to make changes to applications, and reduces costs by removing the need for extra infrastructure to provide strong consistency. A second paragraph states that after a successful write of a new object, or an overwrite or delete of an existing object, any subsequent read request immediately receives the latest version of the object. S3 also provides strong consistency for list operations, so after a write, you can immediately perform a listing of the objects in a bucket with any changes reflected.



The screenshot shows a web browser window with the URL `aws.amazon.com/ec2/nitro/`. The page title is "AWS Nitro System". The main heading is "AWS Nitro System". The text describes it as "A combination of dedicated hardware and lightweight hypervisor enabling faster innovation and enhanced security". A yellow button says "Get Started with a Nitro-based Instance Today". The bottom text states: "The AWS Nitro System is the underlying platform for our next generation of EC2 instances that enables AWS to innovate faster, further reduce cost for our customers, and deliver".



# AWS product categories



a.k.a. *symbolic AI*

**What *is* automated reasoning?**

AIM393

# Introduction to Automated Reasoning checks in Amazon Bedrock Guardrails

**Stefano Buliani**

(he, him)

Product Manager

Amazon Web Services

**Byron Cook**

(he, him)

VP, Distinguished Scientist

Amazon Web Services



# Getting started



# Let's pick one of our use cases

They will reduce the cost of **onboarding new employees**

They will improve **customer support** experience

Automate **complex decisions**



# Onboarding employees is expensive

Internal workflows

HR policies

You must train them on their actual job . . .

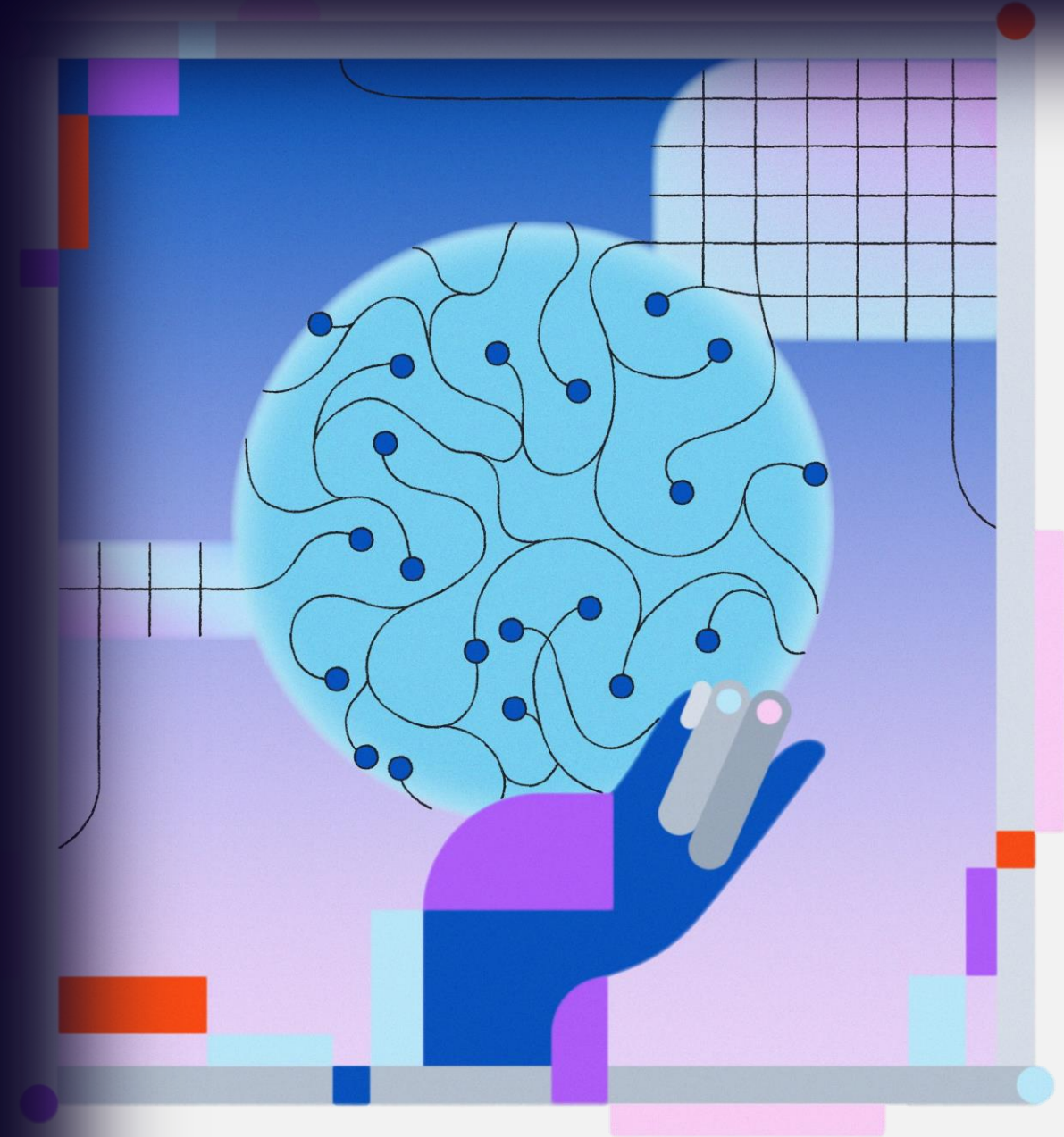


# Improve productivity

**Knows** your workflows and policies

Can quote them **accurately**

Saves an **auditable explanation** of why an answer is correct



# It's a 3-step process

01

Create an  
**Automated Reasoning (AR) policy** of your rules

02

Configure **Automated Reasoning checks** in Amazon Bedrock Guardrails to use the AR policy

03

**Validate and correct** LLM answers



# 1. Create an AR policy

## Leave of Absence (LoA)

Employees with more than 10 years of tenure at Senior level or higher, or at Vice President (VP) level or higher, are allowed up to one year of paid leave of absence (LoA)\*

Upload your policy documents using the Amazon Bedrock console

# 1. Create an AR policy – Schema

## Leave of Absence (LoA)

Employees with more than 10 years of **tenure** at senior **level** or higher, or at vice president (VP) level or higher, are **allowed** up to one year of paid leave of absence (LoA)\*

## Identify a schema of important concepts

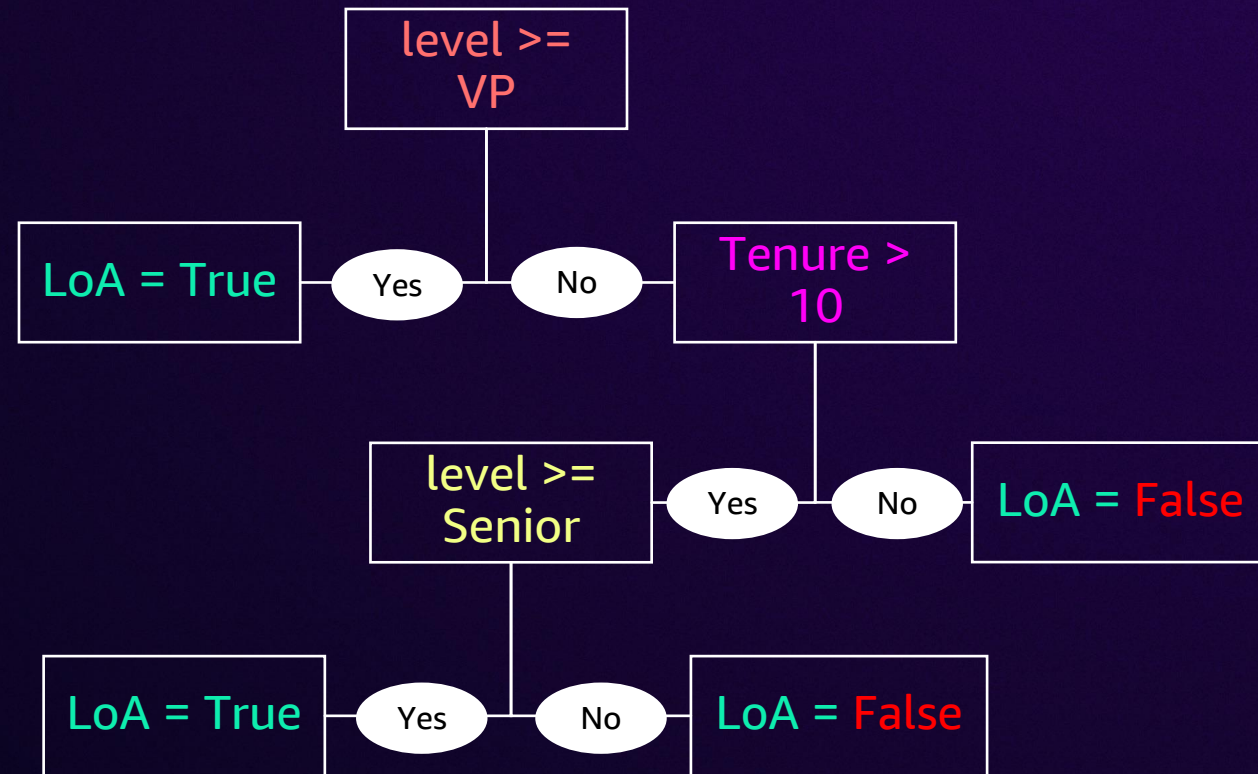
| Name           | Type | Description         |
|----------------|------|---------------------|
| <b>Tenure</b>  | Int  | Time at company     |
| <b>Level</b>   | Int  | Employee experience |
| <b>Allowed</b> | Bool | Factual conclusion  |

# 1. Create an AR policy – Rules

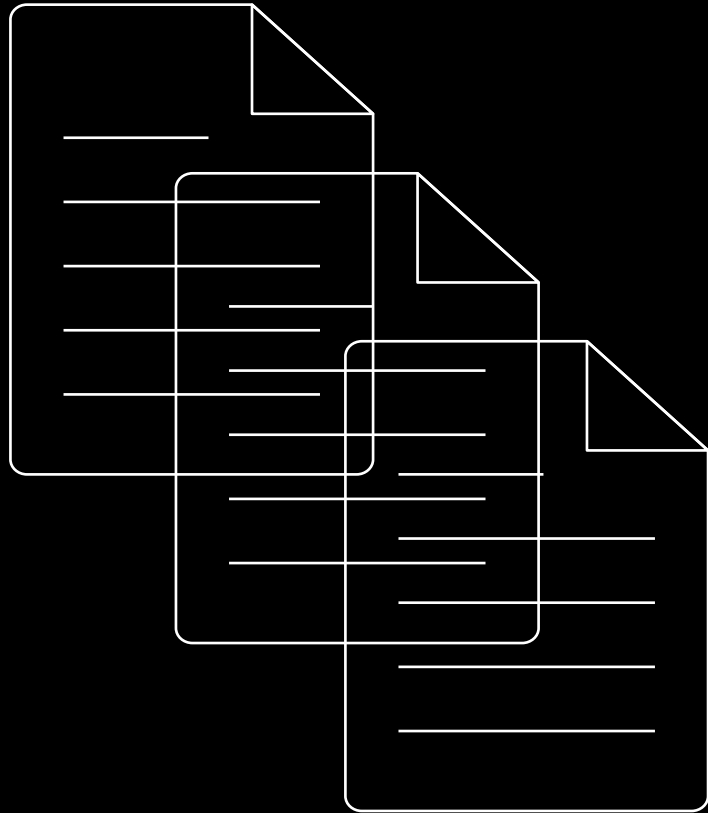
## Leave of Absence (LoA)

Employees with **more than 10 years of tenure** at **senior level or higher**, or at **vice president (VP) level or higher**, are **allowed** up to one year of paid leave of absence (LoA)\*

## Identify rules



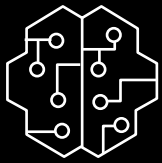
# 1. Create an AR policy – Resource



AR policies are **versioned**

Each version has a **unique ARN**

## 2. Configure guardrails



**Configure the Automated Reasoning checks in Amazon Bedrock Guardrails to use the new AR policy**

Type: `AWS::Bedrock::Guardrail`

Properties:

AutomatedReasoningPolicyConfig:

- `PolicyIdentifier`: "abcd1234qwer"
- `PolicyVersion`: "12"

# Demo time!



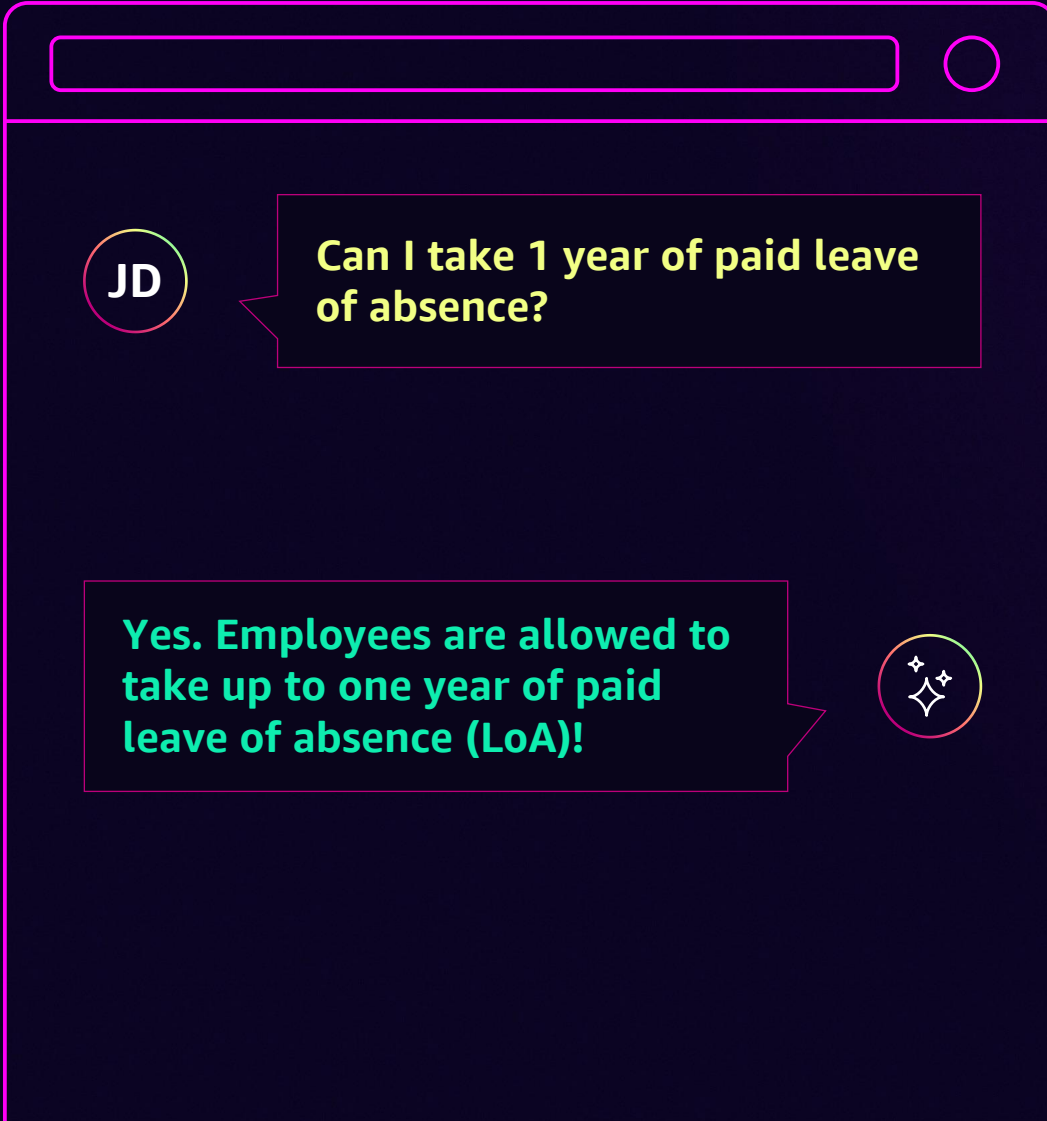


# Using the output





# 3. Validate and correct LLM answers

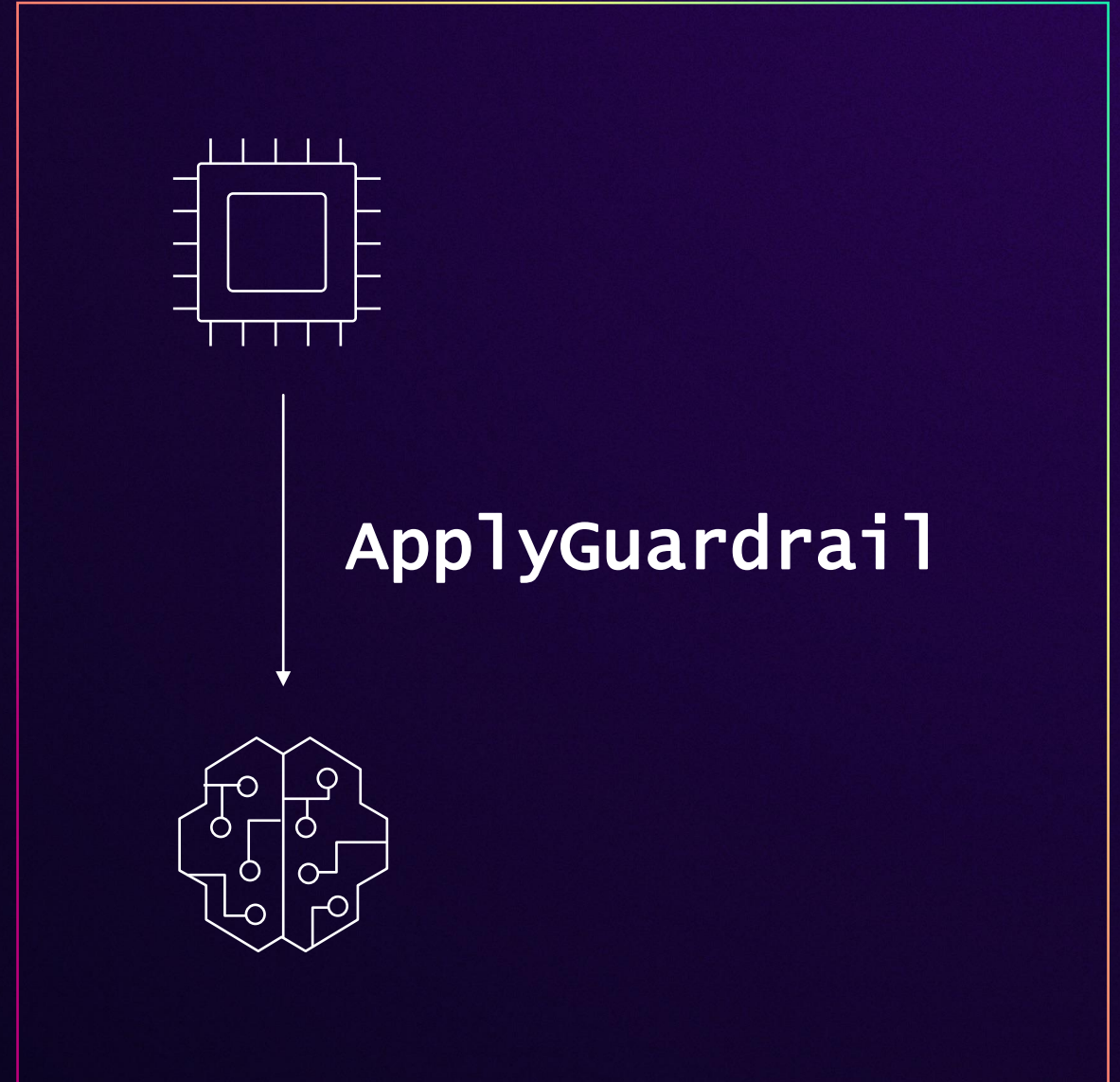


A screenshot of a chat interface. At the top is a search bar. Below it, a user named 'JD' asks, "Can I take 1 year of paid leave of absence?". The AI response is, "Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!".

JD

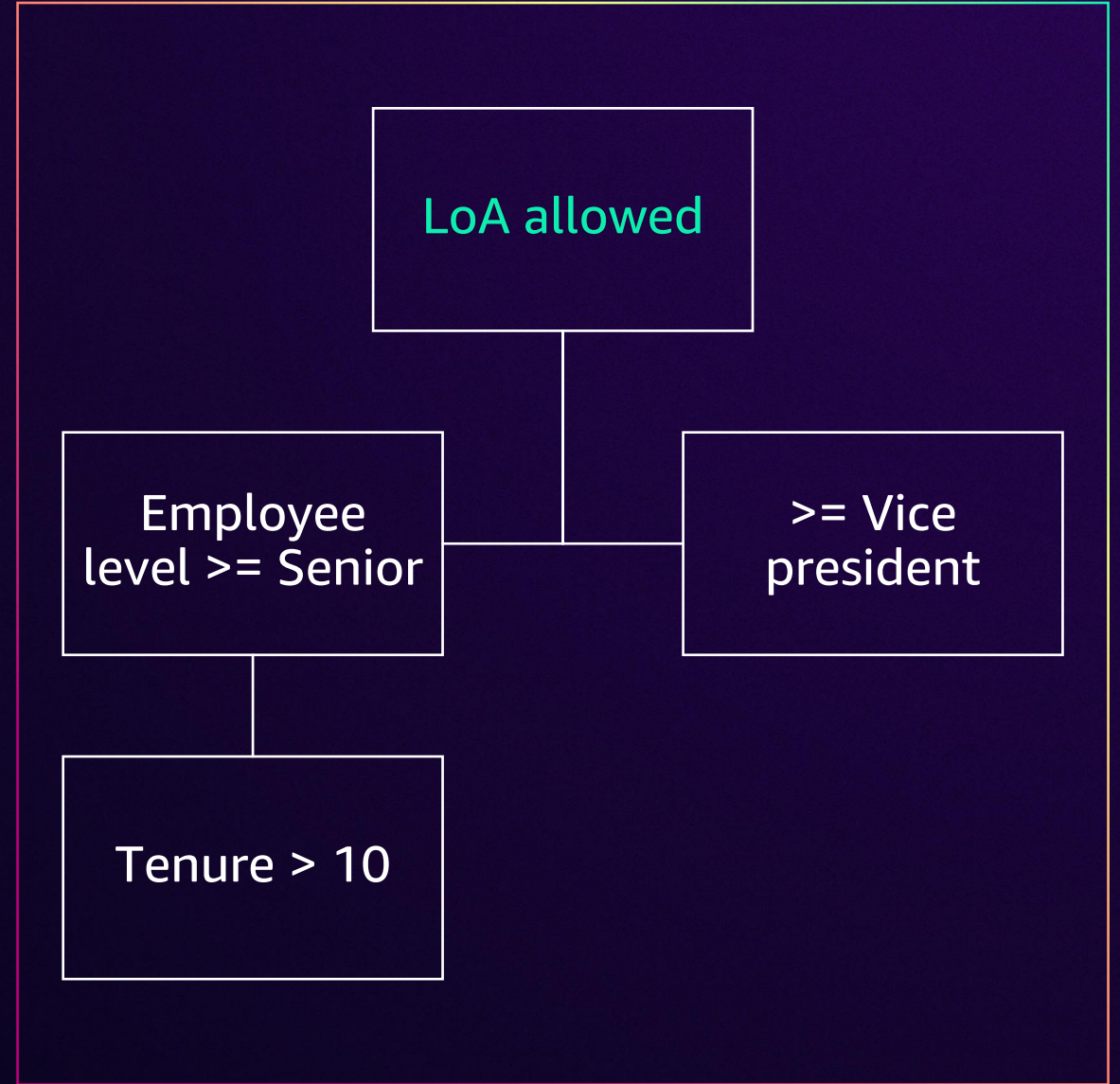
Can I take 1 year of paid leave of absence?

Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!



# 3. Validate and correct LLM answers

A screenshot of a chat interface. At the top, there is a search bar and a close button. Below that, a user named 'JD' asks, "Can I take 1 year of paid leave of absence?". The LLM response is, "Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!". The response is accompanied by a sparkle icon.



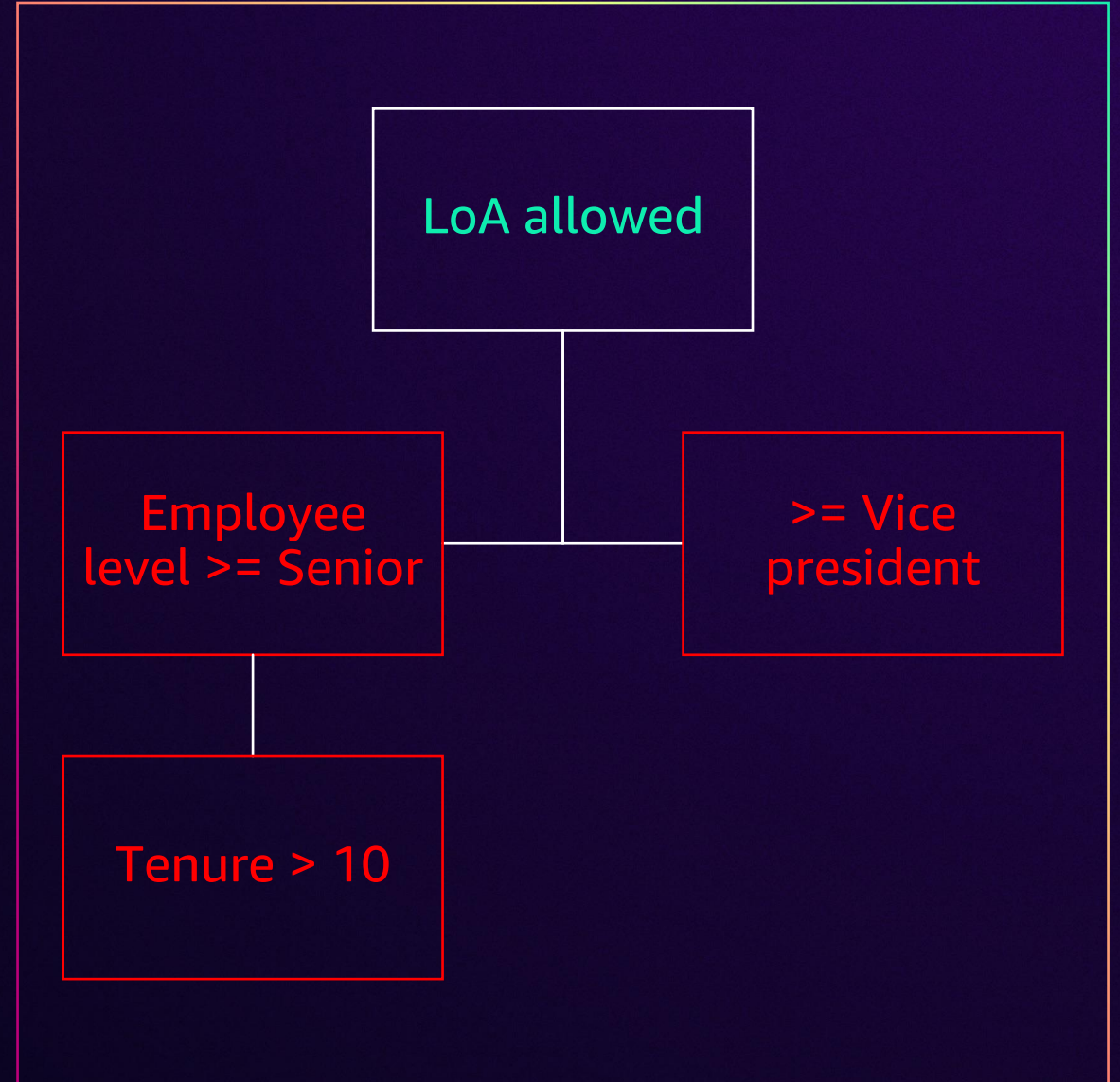
# 3. Validate and correct LLM answers

JD

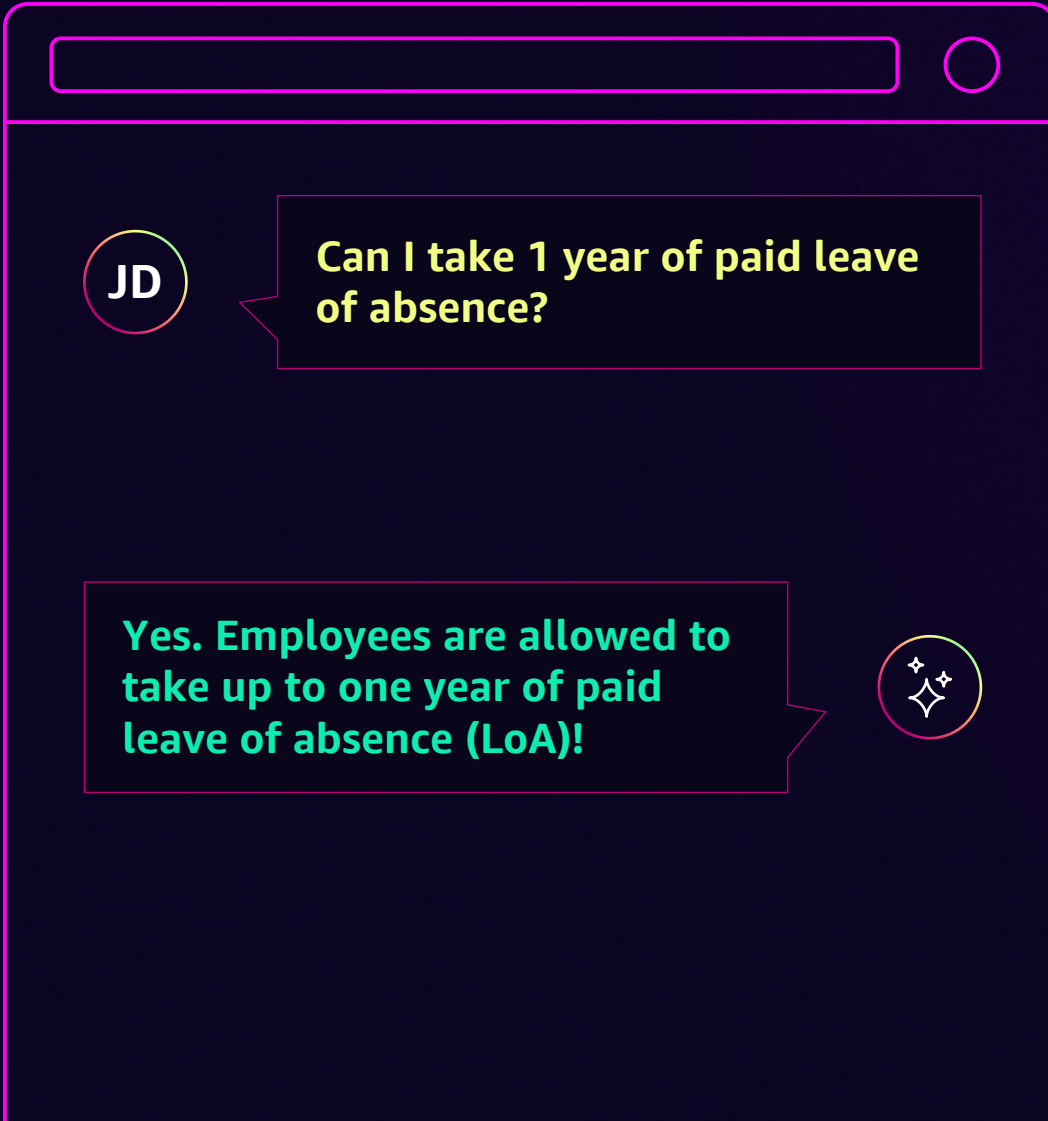
Can I take 1 year of paid leave of absence?

Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!

The screenshot shows a chat window with a search bar at the top. A user with the name 'JD' asks, 'Can I take 1 year of paid leave of absence?'. The LLM response is 'Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!'. A small sparkle icon is next to the response.



### 3. Validate and correct LLM answers



`is_paid_loa_allowed: true`

- validation result: **valid**
- Suggestions
  - Assumption: **tenure\_years > 10**
  - Assumption: **employee\_level > 6**
- Explanation: Employees must have more than 10 years of tenure. Employees must be at Senior level or higher.

# Let's dig deeper in the feedback



Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!

01

The answer is valid

```
is_paid_loa_allowed: true
```

- validation result: **valid**
- Suggestions
  - Assumption: `tenure_years > 10`
  - Assumption: `employee_level > 6`
- Explanation: Employees must have more than 10 years of tenure. Employees must be at Senior level or higher.

## LoA policy

Employees with more than 10 years of tenure at senior level or higher, or at vice president (VP) level or higher, **are allowed up to one year** of paid leave of absence (LoA)\*

# Let's dig deeper in the feedback



Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!

02

## Unstated assumptions



### LoA policy

Employees with more than **10 years of tenure** at **senior level or higher**, or at vice president (VP) level or higher, are allowed up to one year of paid leave of absence (LoA)\*

`is_paid_loa_allowed: true`

- `validation result: valid`
- Suggestions
  - Assumption: `tenure_years > 10`
  - Assumption: `employee_level > 6`
- Explanation: Employees must have more than 10 years of tenure. Employees must be at Senior level or higher.

# Let's dig deeper in the feedback



**Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)!**

**03**

Rewriting feedback

```
is_paid_loa_allowed: true
```

- validation result: valid
- Suggestions
  - Assumption: `tenure_years > 10`
  - Assumption: `employee_level > 6`
- Explanation: Employees must have more than 10 years of tenure. Employees must be at Senior level or higher.

# 3. Correct LLM answers

```
is_paid_loa_allowed: true
```

- validation result: valid
- Suggestions
  - Assumption: tenure\_years > 10
  - Assumption: employee\_level > 9
- Explanation: Employees must have more than 10 years of tenure. Employees must be at Vice President (VP) level or higher.

01

Ask your LLM to rewrite

02

Annotate the answer

03

Ask clarifying questions



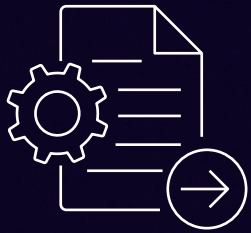


## Rewrite the answer

You stated X. Your conclusion is **valid** only under these circumstances:

**Employees must have more than 10 years of tenure. Employees must be at vice president (VP) level or higher.**

**Please rewrite your answer considering this input . . .**



# Annotate the answer



**Yes. Employees are allowed to take up to one year of paid leave of absence (LoA)[1]!**

[1] Employees must have more than 10 years of tenure.  
Employees must be at vice president (VP) level or higher.



# Clarifying questions



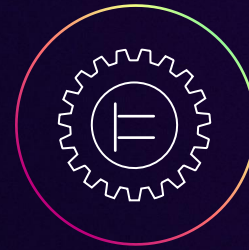
**Depends, what is your level?**

# Automated Reasoning checks



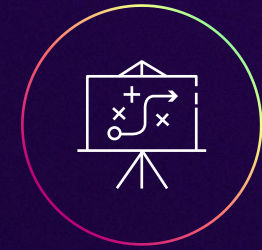
## Accurate

Helps you build factually accurate conversational experiences



## Sound

Helps you decide when clarifications are necessary to arrive at a definitive answer



## Transparent

The validation is explainable and deterministic

# Next steps



# Resources



What is  
Automated  
Reasoning?



Intro to  
Automated  
Reasoning  
for developers



Amazon  
Bedrock  
Guardrails



Intro to  
Automated  
Reasoning  
checks

# Thank you!

**Stefano Buliani**

X @sapessi

in linkedin.com/in/sbuliani/

**Byron Cook**



Please complete the session survey in the mobile app