Introduction to AI

Amanda, Noah, Jason, Zac,
Clayton, Caden, Alice

Table of Contents

01

AI Overview

What is AI?

- Artificial Intelligence (AI) is a field of science concerned with building machines that can reason, learn, and act in such a way that would normally require human intelligence or that involves data whose scale exceeds what humans can analyze.
- AI enables computers and machines to simulate human learning, comprehension, problem solving, decision making, and creativity.

Benefits of AI:

- Automation - factory robots, inspecting products for defects, reduce human error
- Extremely fast, infinitely available, pattern recognition

Different Types of AI

Currently Exists:

Reactive Machines

A limited AI that reacts to different stimuli based on preprogrammed rules. Cannot learn new data, predicts based on given data - ex: Netflix Recommendation Engine

Limited Memory

Uses Deep Learning algorithms to monitor specific situations over time. Data is not saved to memory, but AI is programmed from actions of past and present data - ex: Self-driving cars

Does Not Yet Exist:

Theory of Mind

Potentially can understand the world and how other entities have thoughts and emotions. Understand intentions and predict behavior.

Self Aware

A step above theory of mind, gaining a sense of self and understanding emotions, their state of being, and predicting others' feelings.

Amanda Board - Penguine

Training Models of AI

Supervised Learning

”Inputs and outputs” - Simplest form of ML, consists of using labeled data sets to train algorithms to further classify data or predict outcomes accurately.

Unsupervised Learning

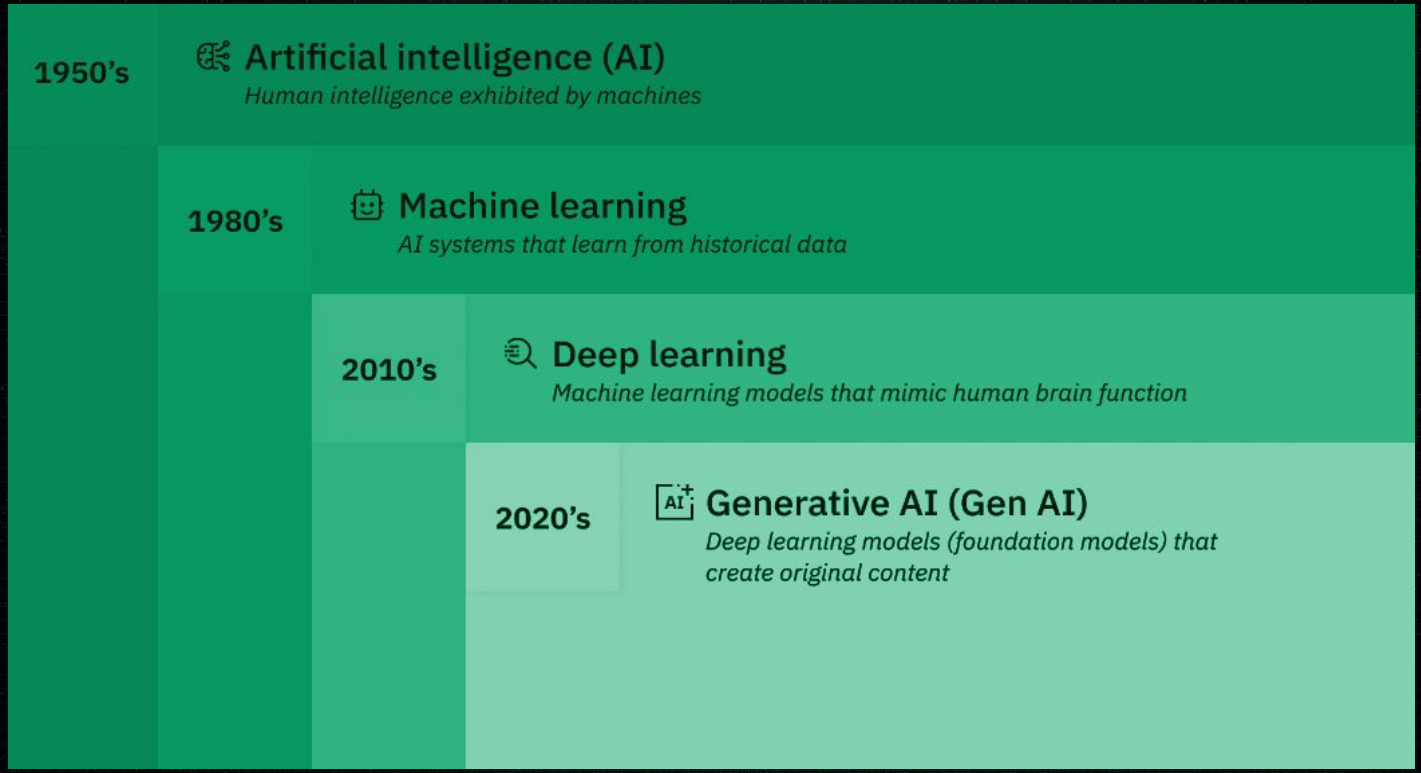
”Without human intervention” - Automate data extraction from large, unlabeled and unstructured data sets to then make their own predictions about what the data represents.

Reinforcement Learning

“Learn by doing” - performs a task through trial and error until desired response. Once desired response is given, agent receives positive reinforcement. If task is done poorly then would receive negative reinforcement.

Amanda Board – Penguine

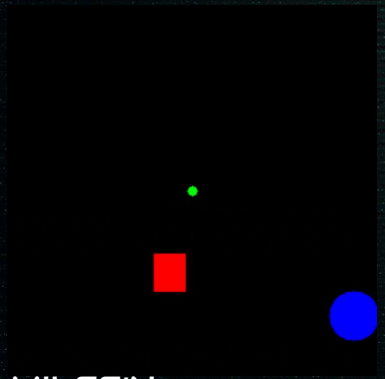
Evolution of AI



Machine Learning vs. Deep Learning

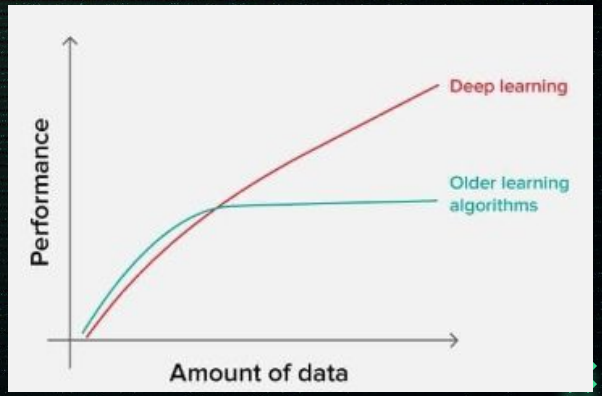
Machine Learning

- Umbrella Term
- Make Optimized Predictions
- Training Data
- Reinforcement training
- Supervised vs. Unsupervised

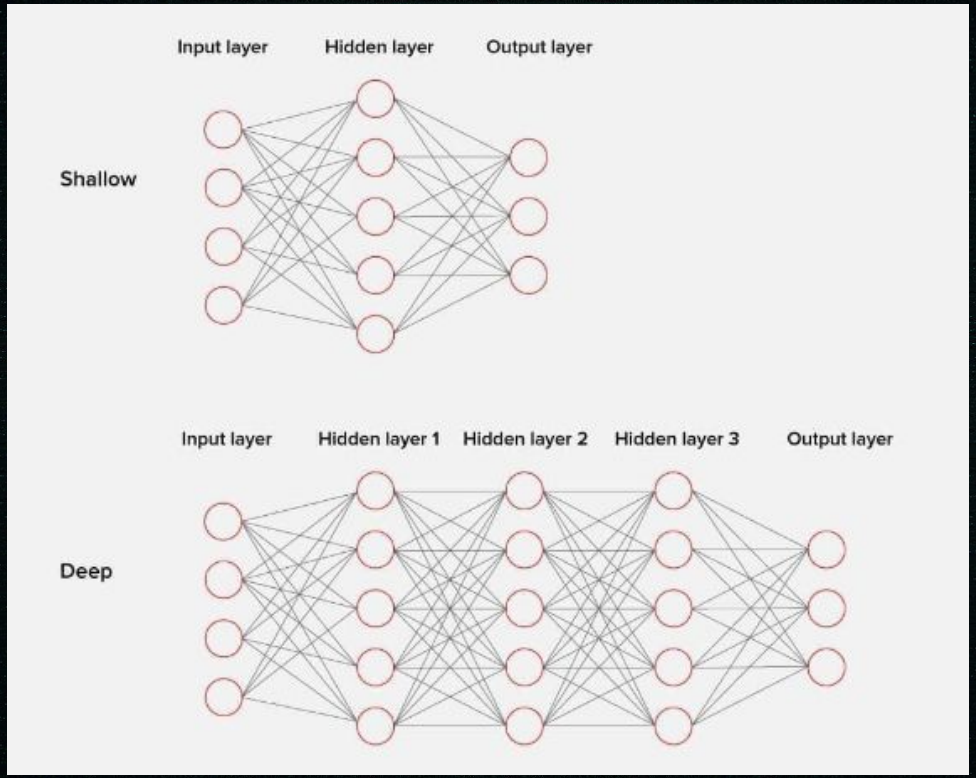


Deep Learning

- Subset of ML
- Large Data Sets
- Scaleable
- Neural Networks
- Black Box process



Neural Network Diagram



History of AI

Inception

Alan Turing's describes his Turing Machine in 1935.

1st Created

Christopher Strachey creates "Checkers" program in 1951.

Major Improvement

IPL, LISP, and PROLOG languages designed with AI in mind (1960-1973).

Major Improvement

Multi-layered Neural Networks massively increased accuracy. (1980s)

Major Improvement

Natural Language processing with IBM Watson, Hanson's Sophia, and assistants Siri and Alexa. (2011-2016)

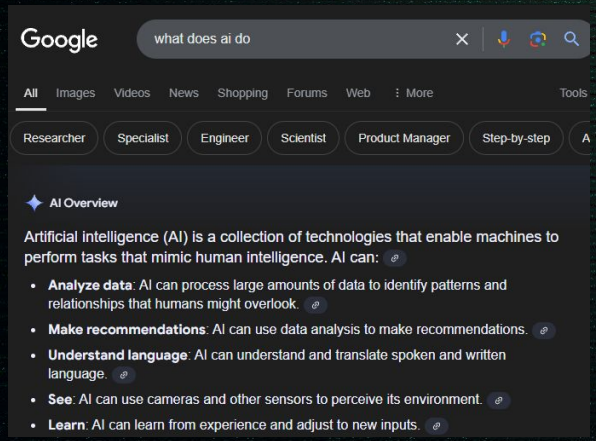
Modern Day

OpenAI releases ChatGPT and DALL-E for generative text and images. (2020s)

Modern Uses

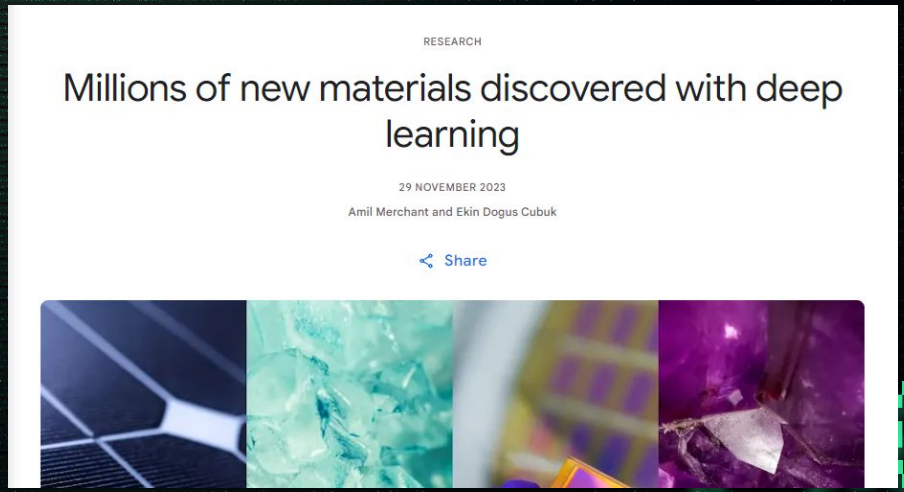
General Public

- Digital Assistants
- Spelling and grammar checks
- Advertising
- Online Shopping recommendations
- Video Games
- Search Engines



Specialized Use

- Medicinal Design (ARCH)
- Inorganic Crystal Discovery (GNoME)
- Protein Structure (AlphaFold)
- Planned Vera C. Rubin Observatory



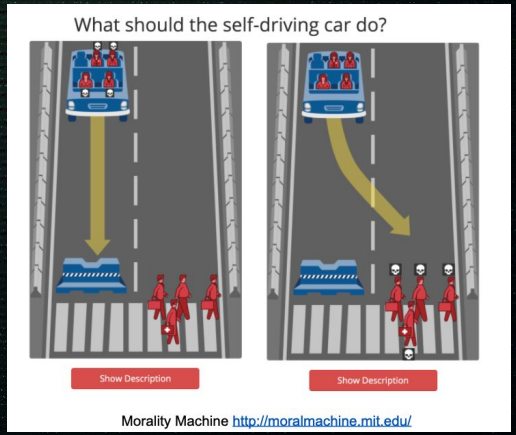
Ethics in AI

Unethical

- Deepfakes
- Generative AI as Actual Work
- Incorrect Information
- Biases in Training Data

Dilemmas

- Lethal Autonomous Weapons
- Copyright Uncertainty
- AI & Jobs
- Self Driving Cars



Example Ethical Dilemma

You run a popular YouTube channel where you review movies and TV shows. One day, you come across an online deepfake tool that allows you to create realistic videos of actors saying anything you'd like. Thinking it would be a fun way to preview an upcoming movie, you create a deepfake video of a beloved actor giving a glowing, exclusive interview about how excited they are for the movie.

In the deepfake, the actor praises the film, even though you haven't seen it yet. You publish the video, clearly labeling it as a "leaked interview," and it goes viral. Viewers are thrilled by the actor's excitement and start eagerly anticipating the movie based on the fake comments. However, a week later, the actual movie releases, and it turns out to be a disappointment. Fans feel misled by your video, thinking the real actor gave a false endorsement.

What should you do next?

A. Take the video down and apologize

B. Edit the video description to clarify that it's a deepfake

C. Leave the video online as is

D. Create more deepfake content

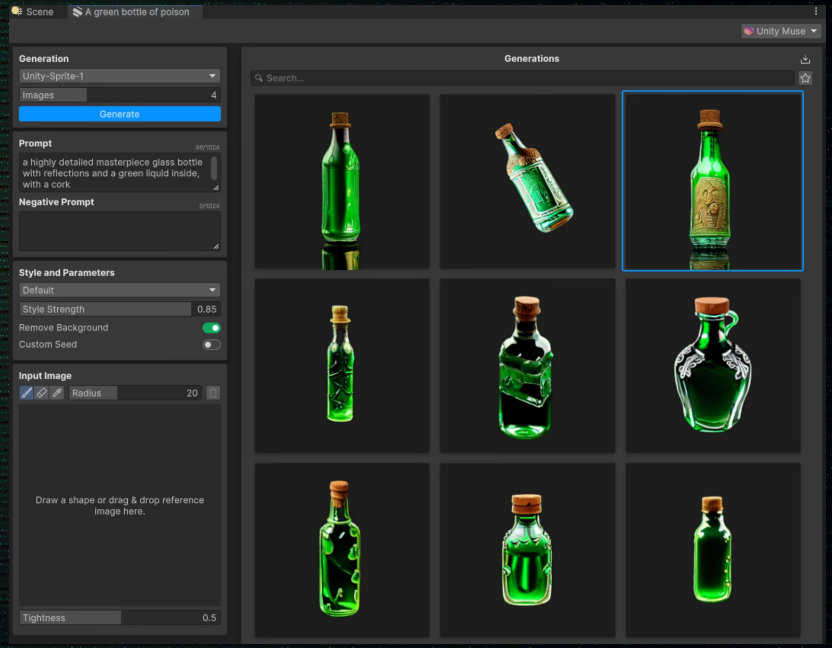
02

Types of AI in Unity

Unity Muse

Built-in generative AI

- Textures
- 2D Sprites
- Animations
- Behaviors
- Problem Solving



Unity Sentis

Package to implement other AI models into Unity

Example use cases:

- Object Identification
- Handwriting Detection
- AI Enemy Difficulties
- AI NPC Dialogue

Pathfinding

Built-in pathfinding - NavMesh

- Easy to use
- Very simple
- Can’t do some things

A* Pathfinding Package

- Tilemap
- Node based
- Much more complex

03
Demo Modes

Game Demos

- A short presentation to show off selected features of a game
- They can involve interactive or visual representations of the game
- Can be playable or non-playable

Playable Demos

- What?
  - A playable version of the game
  - Usually a small/limited version of the game
  - Requires enough development to be completed
- Why?
  - Most engaging / interactive form of a demo
  - More information about the game can be collected
    - Bugs
    - Server Issues

Non-Playable Demos

- What?
  - A video or some other visual presentation of the game
  - Shows off features and/or narrative elements
- Why?
  - Can be done early or late in development
  - Helpful marketing tool
  - Can be used to show progress in development

04

Static/Dynamic Binding Intro

Static Binding

What is static binding?

- Computer determining everything at compile-time
- Deciding what types will be used, methods, etc
- It is also called early binding

Where does static binding shine? I.E What are the best uses for it?

- It improves performance because everything is assigned before running your program.
- Easier to debug (don't have to wait until the code is running to see errors)
- If your program will know the type of your object or the methods it contains

How does static binding differ from dynamic binding?

- Dynamic binding is called late binding
- Bound at run-time, types are not assigned beforehand
- Slower
- Keyword "virtual" for methods
- Allows for more flexibility

Example of Static Binding

Where could you potentially replace static binding with dynamic binding?
Enemy: have a general attack method for an enemy class, allow different subclasses of enemies to override that Attack method with their own unique attacks. Can use keywords “virtual” and “override” for your method

Basic example for static binding: here all enemies, regardless of subclass, would be using the same attack method when it’s called. The WaterEnemy could not have its own, unique Attack Method.

```
using UnityEngine;
2 references
public class Enemy : MonoBehaviour{

    0 references
    int basicspeed= 12f;
    1 reference
    public static void Attack(){
        Debug.Log("Enemy begins attacking");
    }
}
0 references
public class WaterEnemy : Enemy{
    0 references
    int waterspeed = 10f;
}
```

How to use Dynamic Binding

- Two classes required for Dynamic binding: A superclass and subclass
  - EX: Grandpa and Father.

```
public partial class Grandpa  {
    public virtual void DoDynamic() { GD.Print($"Dynamic Grandpa"); }
    public void DoStatic() { GD.Print($"Static Grandpa"); }
}
public partial class Father : Grandpa {
    public override void DoDynamic() { GD.Print($"Dynamic Father"); }
    public void DoStatic() { GD.Print($"Static Father"); }
}
```

Static Type

Dynamic Type

```
Grandpa func1 = new Father();
```

- To define a dynamically bound function, you must use the keyword “Virtual” this can be paired with the keyword “Override” to define when and what you would like bound dynamically.
- Dynamic binding is performed at run time, making it slower than static binding

05

Dynamic Binding Examples

```
using Godot;
using System;

public partial class Sprite2d : Sprite2D {
    public override void _Process(double delta) {
        SuperClass func1 = new SubClass(); // Dynamic binding
        SubClass func2 = new SubClass(); // Static Binding
            if (Input.IsActionJustPressed("move")) {
                func1.DoDynamic();
                func1.DoStatic();
                func2.DoDynamic();
                func2.DoStatic();
            }
    }
}
public partial class SuperClass : Sprite2D {
    public virtual void DoDynamic() { GD.Print($"SuperClass dynamic function"); }
    public void DoStatic() { GD.Print($"SuperClass static function"); }
}
public partial class SubClass : SuperClass {
    public override void DoDynamic() { GD.Print($"SubClass dynamic Function"); }
    public void DoStatic() { GD.Print($"SubClass static function"); }
}
```

What Will the outputs be?

Outputs

```
using Godot;
using System;

public partial class Sprite2d : Sprite2D {
    public override void _Process(double delta) {
        SuperClass func1 = new SubClass(); // Dynamic binding
        SubClass func2 = new SubClass(); // Static Binding
            if (Input.IsActionJustPressed("move")) {
                func1.DoDynamic();
                func1.DoStatic();
                func2.DoDynamic();
                func2.DoStatic();
            }
    }
}
public partial class SuperClass : Sprite2D {
    public virtual void DoDynamic() { GD.Print($"SuperClass dynamic function"); }
    public void DoStatic() { GD.Print($"SuperClass static function"); }
}
public partial class SubClass : SuperClass {
    public override void DoDynamic() { GD.Print($"SubClass dynamic Function"); }
    public void DoStatic() { GD.Print($"SubClass static function"); }
}
```

```
SubClass dynamic Function
SuperClass static function
SubClass dynamic Function
SubClass static function
```

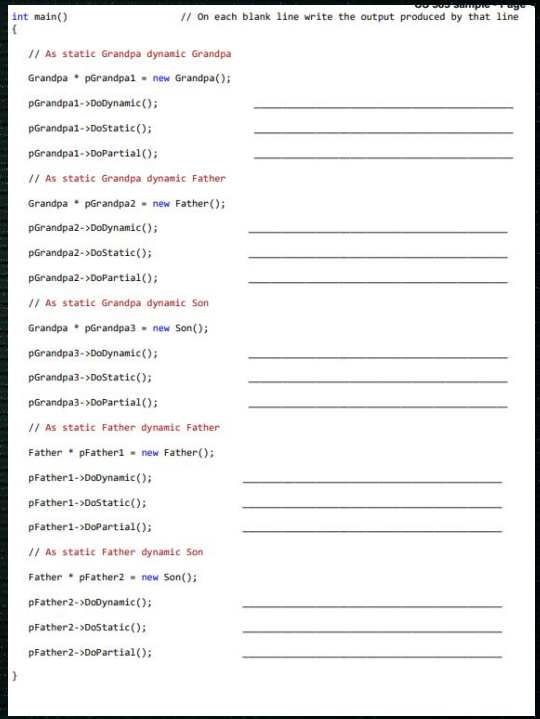
Grandpa Father son explanation

```cpp
#include <iostream>
using namespace std;

class Grandpa
{
   public:
     virtual void DoDynamic() {cout << "      Dynamic Grandpa" << endl;}
     void DoStatic(){cout << "      Static Grandpa" << endl;}
     void DoPartial() {cout << "      Partial Grandpa" << endl;}

};

class Father: public Grandpa
{
   public:
     void DoDynamic(){ cout << "      Dynamic Father" << endl; }
     void DoStatic(){cout << "      Static Father" << endl;}
     virtual void DoPartial() {cout << "      Partial Father" << endl;}
};

class Son: public Father
{
   public:
     void DoDynamic(){ cout << "      Dynamic Son" << endl; }
     void DoStatic(){cout << "      Static Son" << endl;}
     void DoPartial() {cout << "      Partial Son" << endl;}
};
```

```cpp
int main()                    // On each blank line write the output produced by that line
{
   // As static Grandpa dynamic Grandpa
   Grandpa * pGrandpa1 = new Grandpa();
   pGrandpa1->DoDynamic();  _____
   pGrandpa1->DoStatic();   _____
   pGrandpa1->DoPartial();  _____
   // As static Grandpa dynamic Father
   Grandpa * pGrandpa2 = new Father();
   pGrandpa2->DoDynamic();  _____
   pGrandpa2->DoStatic();   _____
   pGrandpa2->DoPartial();  _____
   // As static Grandpa dynamic Son
   Grandpa * pGrandpa3 = new Son();
   pGrandpa3->DoDynamic();  _____
   pGrandpa3->DoStatic();   _____
   pGrandpa3->DoPartial();  _____
   // As static Father dynamic Father
   Father * pFather1 = new Father();
   pFather1->DoDynamic();   _____
   pFather1->DoStatic();    _____
   pFather1->DoPartial();   _____
   // As static Father dynamic Son
   Father * pFather2 = new Son();
   pFather2->DoDynamic();   _____
   pFather2->DoStatic();    _____
   pFather2->DoPartial();   _____
}
```

Grandpa, Father, Son Answers

1. Dynamic Grandpa
2. Static Grandpa
3. Partial Grandpa
4. Dynamic Father
5. Static Grandpa
6. Partial Grandpa
7. Dynamic Son
8. Static Grandpa
9. Partial Grandpa
10. Dynamic Father
11. Static Father
12. Partial Father
13. Dynamic Son
14. Static Father
15. Partial Son



```
int main()                          // On each blank line write the output produced by that line
{
  // As static Grandpa dynamic Grandpa
  Grandpa * pGrandpa1 = new Grandpa();
  pGrandpa1->DoDynamic();           _____
  pGrandpa1->DoStatic();            _____
  pGrandpa1->DoPartial();           _____
  // As static Grandpa dynamic Father
  Grandpa * pGrandpa2 = new Father();
  pGrandpa2->DoDynamic();           _____
  pGrandpa2->DoStatic();            _____
  pGrandpa2->DoPartial();           _____
  // As static Grandpa dynamic Son
  Grandpa * pGrandpa3 = new Son();
  pGrandpa3->DoDynamic();           _____
  pGrandpa3->DoStatic();            _____
  pGrandpa3->DoPartial();           _____
  // As static Father dynamic Father
  Father * pFather1 = new Father();
  pFather1->DoDynamic();            _____
  pFather1->DoStatic();             _____
  pFather1->DoPartial();            _____
  // As static Father dynamic Son
  Father * pFather2 = new Son();
  pFather2->DoDynamic();            _____
  pFather2->DoStatic();             _____
  pFather2->DoPartial();            _____
}
```

Unity Demo

Zachary Squires - Teal Team 6

What Do I Need For the Oral Exam?

Wow I’m so glad you asked, here is a video

DynamicBindingShadowNew.mp4

06

DEADLINES

Timeline

10/24: TL4 Weekly Status Report

10/29: TL6 Presentation

10/31: TL5 Weekly Status Report and TL5 Deliverables due

11/07 TL6 Weekly Status Report

11/10-15 ORAL EXAM WEEK

Be prepared with all team lead deliverables!

Thanks

Do you have any questions?

Sources

https://cloud.google.com/learn/what-is-artificial-intelligence
https://www.coursera.org/articles/types-of-ai
https://www.ibm.com/topics/artificial-intelligence
https://www.ibm.com/think/topics/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks
https://www.ibm.com/think/topics/supervised-vs-unsupervised-learning
https://www.tableau.com/data-insights/ai/examples
https://www.ajmc.com/view/accelerating-drug-discovery-with-ai-for-more-effective-treatments
https://deepmind.google/discover/blog/alphafold-using-ai-for-scientific-discovery/
https://deepmind.google/discover/blog/millions-of-new-materials-discovered-with-deep-learning/
https://www.popsci.com/science/ai-astronomy/
https://www.geeksforgeeks.org/early-and-late-binding-in-c-sharp/

Sources

https://research.aimultiple.com/ai-ethics/
https://www.britannica.com/science/history-of-artificial-intelligence/
https://www.dataversity.net/a-brief-history-of-neural-networks/
https://www.coursera.org/articles/history-of-ai
https://blog.metaphysic.ai/deepfakes/
https://www.moralmachine.net/
https://itstamart.github.io/blog/rl-for-navigation/
https://unity.com/products/muse
https://unity.com/products/sentis
https://arongranberg.com/astar/