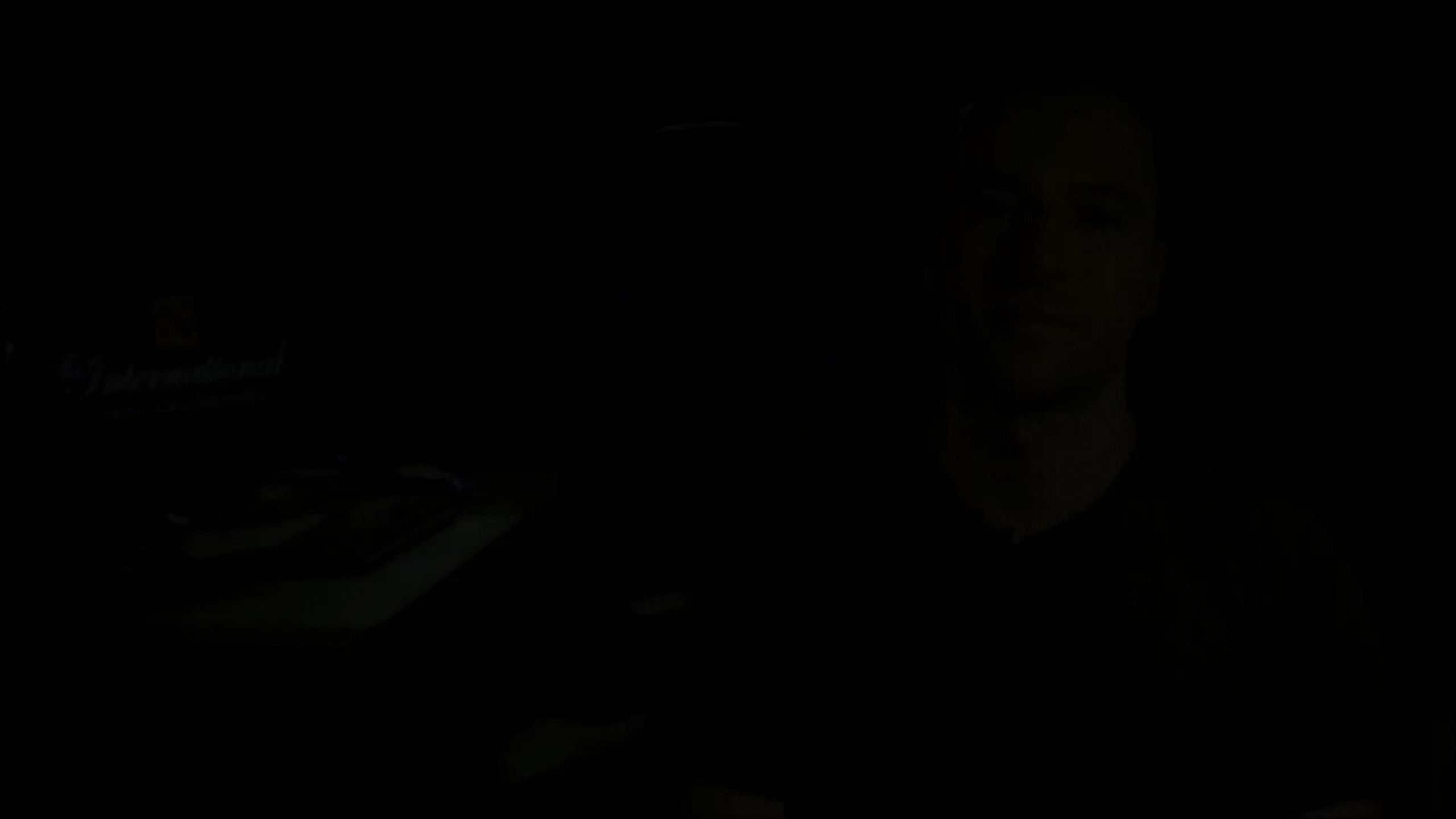
KNIGHTEC

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Welcome

18.00	Grab some food and drink
18.20	Region Manager Carl Falk – Introduction
18.30	AI/ML seminar part 1 - Stefan Danielsson
19.15	Pause and “fika”
19.30	AI/ML seminar part 2 – Jonas Dürango
20.15	Finish

KNIGHTEC



“Game AI has always been used as a measure of progress in AI, throughout it’s history”.

Approaching Machine Learning

with Artificial Intelligence, Gadgets & Games

KNIGHTEC

Stefan Danielsson

Experience

Software Developer, Knightec

MDH - Computer Science

Artificial Intelligence

- **Studies**
- **Thesis EEG signal classification and feature selection**

Hobbies

Programming

Robots

2D games

3D printing

CNC milling and turning

KNIGHTEC

How familiar are you with AI?

a) I have never heard of it. What is this dark thing you speak of?

b) I am not sure what it is, but I have my guesses

c) I have read a lot about AI in magazines, articles and on the web

d) I have studied AI to some extent, and I have some practical experience

e) I incorporate AI in my work, every day

f) I probably know more about AI, then anyone else in this room

KNIGHTEC

Artificial Intelligence history in 60 seconds

Thinking machines

In 1950, Alan Turing published a paper, in which he speculated about the possibility of creating machines that can think. This paper lay the foundation for how intelligent behavior could be distinguishable from that of a human.

Birth of Game AI

In 1951, the first renowned checkers program and chess program, was written..

Mimicking problem solving

In 1955, a program called “Logic Theorist” was developed. It was the first program deliberately engineered to mimic the problem solving skills of a human mathematician. Among other things, it proved mathematical theorems, and even developed more elegant proofs for some theorems.

KNIGHTEC

Industrial robots

In 1961, the company Unimation deployed their industrial robot “Unimate”, on an automobile assembly line, for GM.

Expert systems

In 1965, the first “expert systems” started to take root. The systems could answer questions, or solve problems about a specific domain of knowledge, using logical rules that were derived from the knowledge of experts.

Bayesian methods

In the 1960s, major breakthroughs were made for probabilistic inference, in the field of Statistics and Machine learning.

KNIGHTEC

First autonomous vehicle

In 1979, the “Stanford Cart” becomes the first computer-controlled, autonomous vehicle, after it successfully traverses a chair-filled room.

Backgammon AI

In 1979, a program called “BKG” defeats the reigning backgammon world champion.

Common sense knowledge

In middle of the 1980s, the “Cyc” project was introduced. It was the first attempt to solve the common sense knowledge problem, by creating a massive database, that would contain all the ordinary facts, that an average person knows.

Neural networks

In 1986, the concept of backpropagation for neural networks is introduced, and the concepts of neural networks becomes widely accepted and used.

KNIGHTEC

Intelligent agents

In the 1990s, IA became popular and widely used. IA is an autonomous entity, which observes through sensors, and acts upon an environment using actuators, and directs its activity towards achieving goals.

Checkers AI

In 1994 - Checkers world champion lost to a checkers program called “Chinook”.

Vector machines

In 1995, Vector machines started to become widely used.

“No Hands Across America“

In 1995, a semi-autonomous car drove coast-to-coast across the United States with computer-controlled steering.

Othello AI

In 1997, a program called “Logistello” defeated the world champion.

Deep Blue versus Kasparov

In 1997, the computer “Deep Blue” wins over the reigning chess champion Garry Kasparov.

KNIGHTEC

Google Translate	Google launched their human language translator in 2006.
Atari 2600	In 2013, DeepMind successfully could learn to master just about any Atari 2600 games, and beat the wold champions in some of them.
AlphaGo	In 2016, Google's AlphaGo program becomes the first program to beat an professional human Go player.
Star Craft	In 2016, DeepMind and Blizzard initiated an AI collaboration towards StarCraft, as a possible future challenge, since it requires a high level of strategic thinking and handling imperfect information. Blizzard develops a Machine Learning API that gives researchers and developers hooks into the game.
OpenAI & Dota II	During “The International” in 2017, OpenAI let a machine-learned bot play 1v1 demonstration games, against professional Dota 2 players. They all got wrecked!
Chess master in 4 hs	This Christmas, after being programmed with only the rules of chess (no strategies), in just four hours, Google’s “AlphaZero” had mastered the game to the extent it was able to best the highest-rated chess-playing program “Stockfish”.

KNIGHTEC

2010… boom!

IoT

More people and things than ever, are connected.

Availability

Everyone can afford it. Anyone can use it.

Big data & VLDB

The ability to store, capture, analyze, search, secure, share, visualize, query, update, and process large amounts of data.

Cloud computing

Combined computational power, in a dynamic fashion. Accessible, from anywhere!

End-node computing

Decrease in space complexity. Increase in data relevance.

Hardware performance

Hardware acceleration, transistor count, GPU computing, and more…

Deep Learning is kicking into gear!

KNIGHTEC

What is Artificial Intelligence?

KNIGHTEC

Weak-, or applied AI

An intellect that follows rules, that humans gave it, in order to achieve some goal. This kind of intellect can be perceived, as “smart”, but in fact, this intellect is not smart in any way, since it does not in any way resembles the intellect that of a human brain. It lacks the fundamental features that is associated with human intelligence. For example, the ability to reason, associate, relate, generalize, or to be creative. The intellect merely autonomously follows the rules given to it.

Strong-, or generalized AI

An intellect that focuses on mimicking the behavior of a human brain, rather then achieving optimal solutions to a problem. Focus lies on the ability to reason, associate, relate, generalize, or to be creative, in the same fashion that of a human brain. In other words it could be argued that in a “Strong AI”, the reasoning itself is more important then the actual solution.

Superintelligence

A sentient and self-aware intellect, that goes beyond and overcomes the limitations of our known physical existence, and that by some definitions also becomes independent of it. A system that is more intelligent than every person on earth, that has ever lived, combined!

KNIGHTEC

Learning through examples

From basic weak AI, to ML and learning.

Illustrated and visualized through robots, games and applications. Focus on the behavior, and the logic behind the algorithms.

- **Autonomous robots**
- **Basic swarm AI**
- **Optimization and search trees**
- **Cumulative rewarding**
- **Complexity and problem construction**
- **Evolutionary computation and Genetic Algorithms**
- **Artificial Neural Networks, and supervised learning**

Examples will be available in a GIT-repository for you to try out after the event!

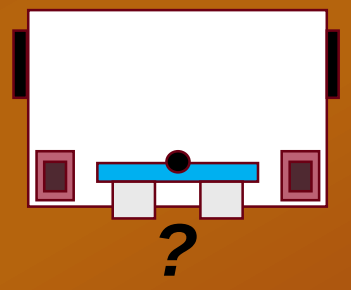
KNIGHTEC

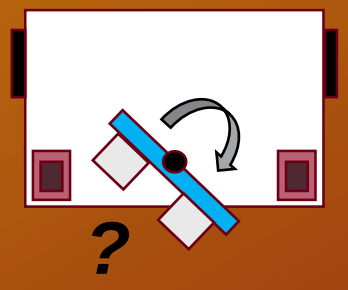
Autonomous robots

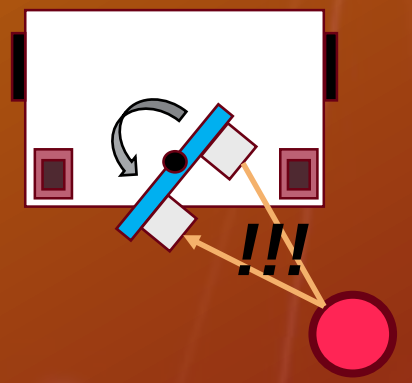
The 1985s robotic paradigm

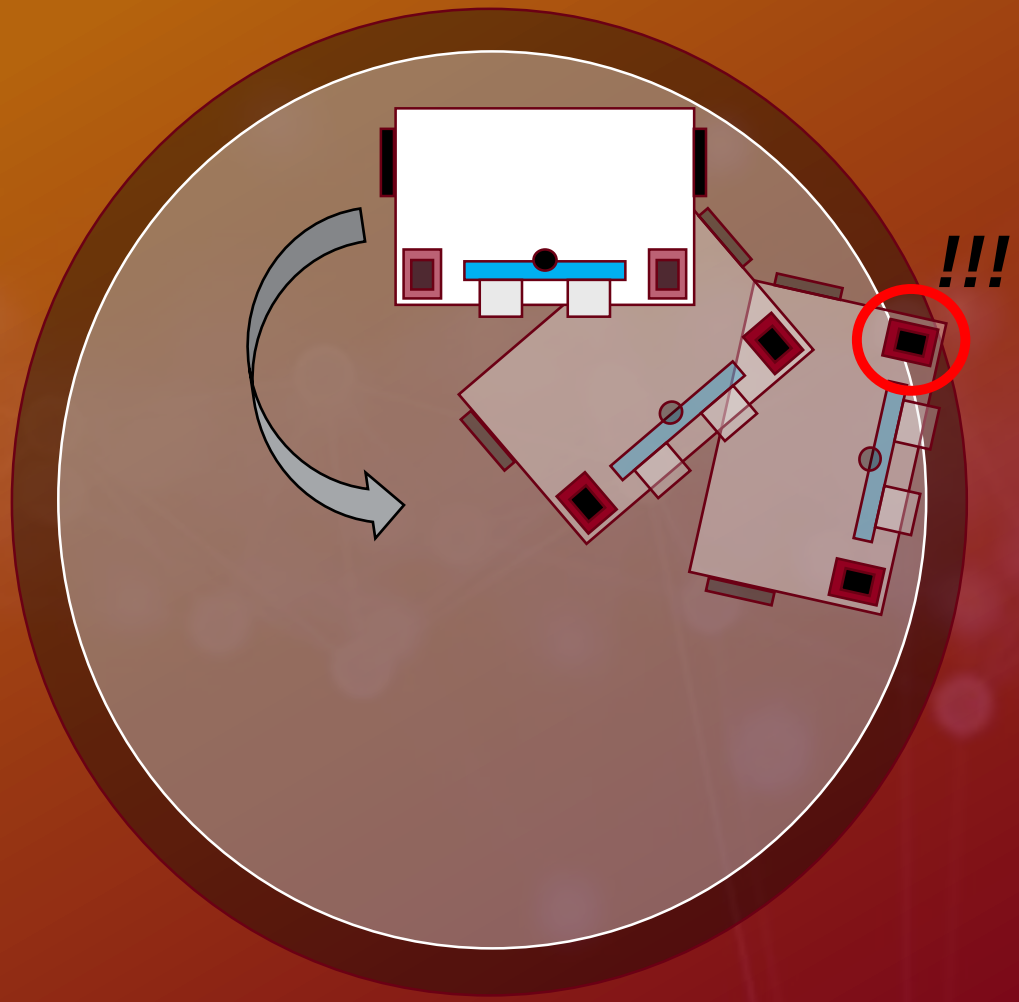
- Sense – Gather sensor data from the environment
- Plan – Interpret the data, and make an *independent* decision based on it
- Act – Perform a suitable action, based on the decision

KNIGHTEC



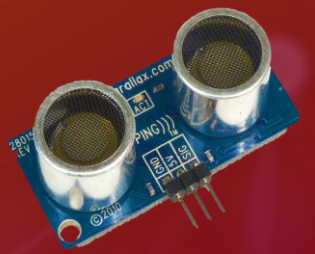






Sonar sensor

Sensor data	Left	Front	Right
Raw time	-	-	8 ms
Servo Angle	-75°	0°	+75°
PWM	255	255	35
Distance	> 100 mm	> 100 mm	17.5 mm
Hit	False	False	True



Infrared reflectance sensor

Sensor data	Left	Right
PWM	255	35
Hit	False	True

KNIGHTEC

We can approximate parallelism with interrupts and timers.

```
// Main loop

Interrupt sensors();
Interrupt servo();

while(true)
{
      switch(decision)
      {
            case decision1:
            case decision2 :
            case decision3 :
            …
               default:
      }
}
```

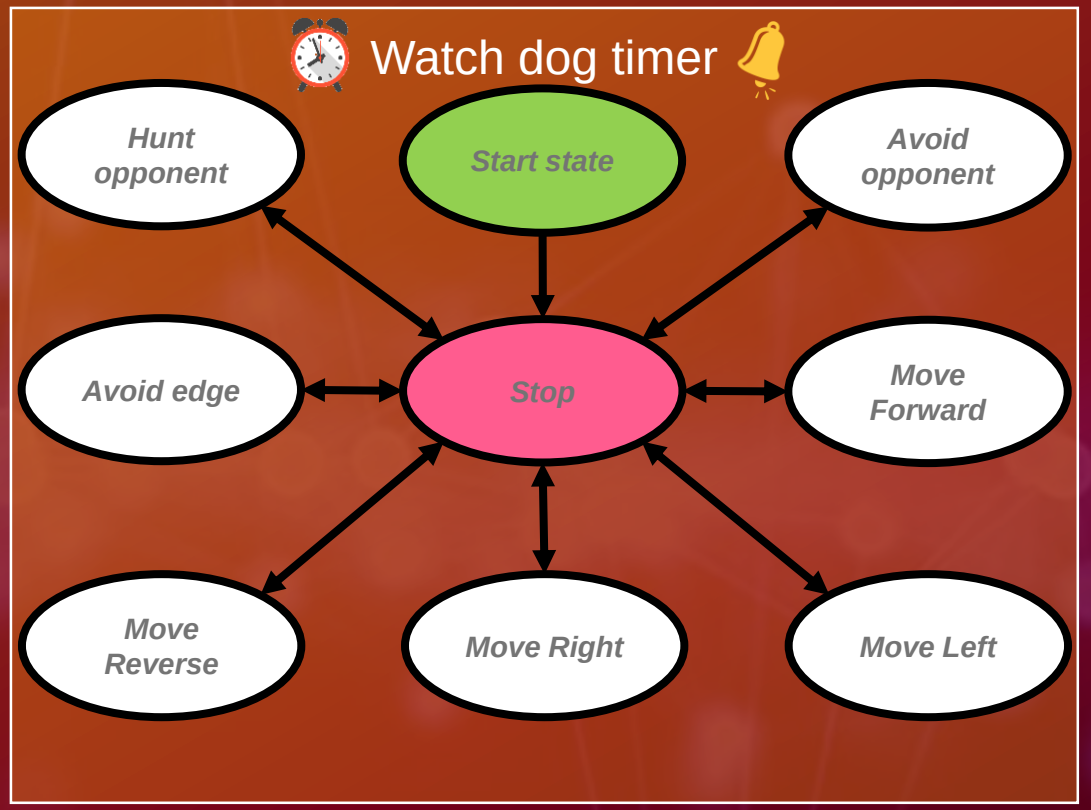
```
// Sensor(s) interrupt

distance = readPing();
lineHit = readReflectance();
```

```
// Servo interrupt

while (1)
{
      sweepServo(direction, step);
}
```

A simple robot can be thought of as a state machine



KNIGHTEC

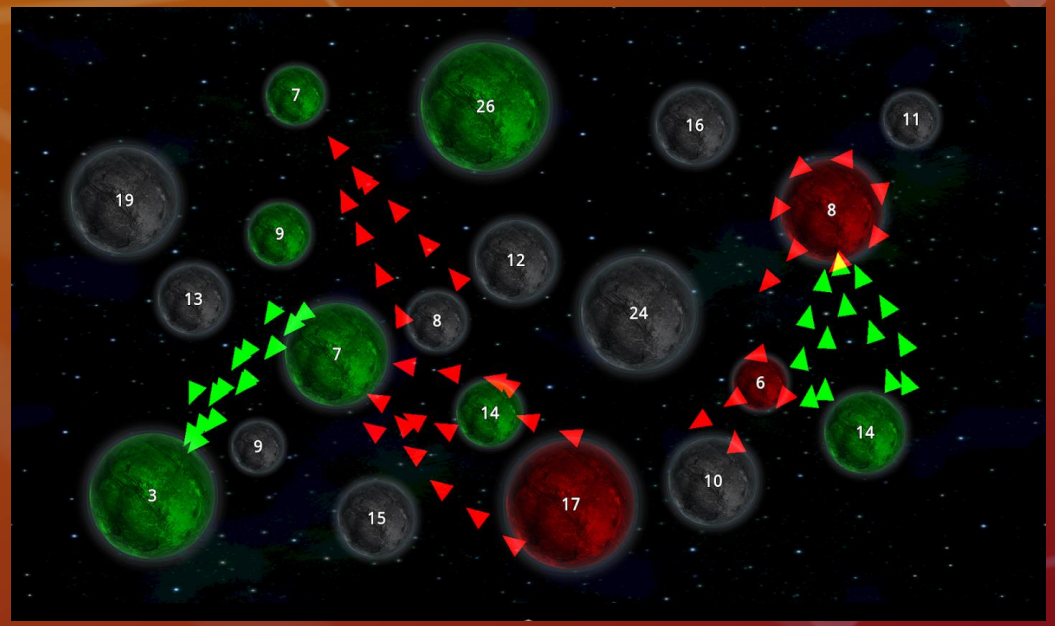
Demo

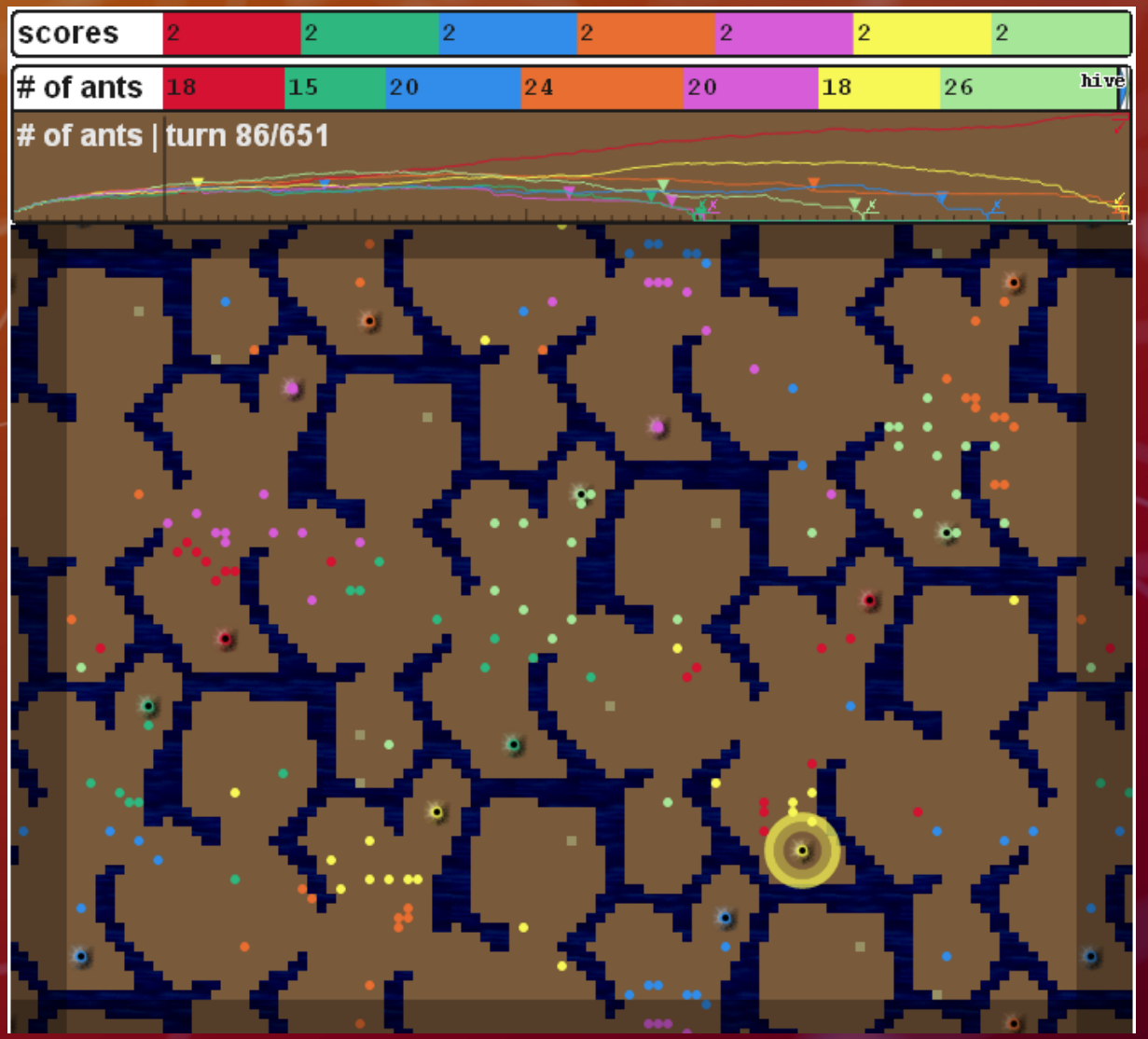
KNIGHTEC

Basic swarm AI

Google AI challenge – AI vs AI

- **Ant wars**
- **Planet wars**
- **Tron**





KNIGHTEC

The Ant wars Engine

Food = MORE ants!

Imperfect information – You only know what you can see. You have to explore.

Each turn your bot will be given information for all squares that are visible to your ants.

A bot can issue up to one order for each ant (move one square), during one turn (UP, DOWN, LEFT, RIGHT).

KNIGHTEC

The Ant wars Engine

The game then goes through 5 phases, in the following order:

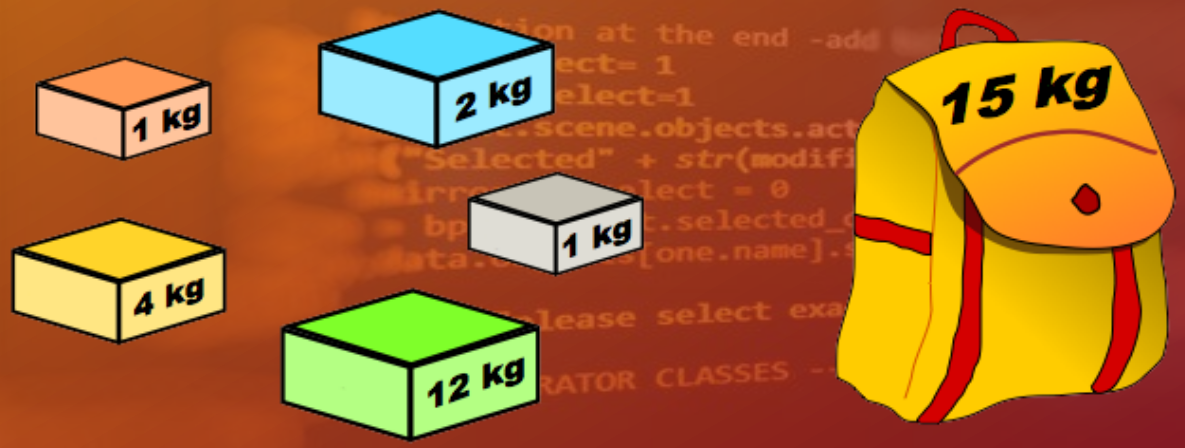
(1) **move** all ants (ants that collide in the same square are killed)
(2) **attack** enemy ants if within range
(3) **raze** ant hills with enemy ants positioned on them
(4) **spawn** more ants at hills, and food on map
(5) **gather** food next to ants

KNIGHTEC

Demo

KNIGHTEC

Construction and problem representation



”The knapsack problem”

We have one knapsack, and some items.

Items have:

- *Weight*
- *Value*

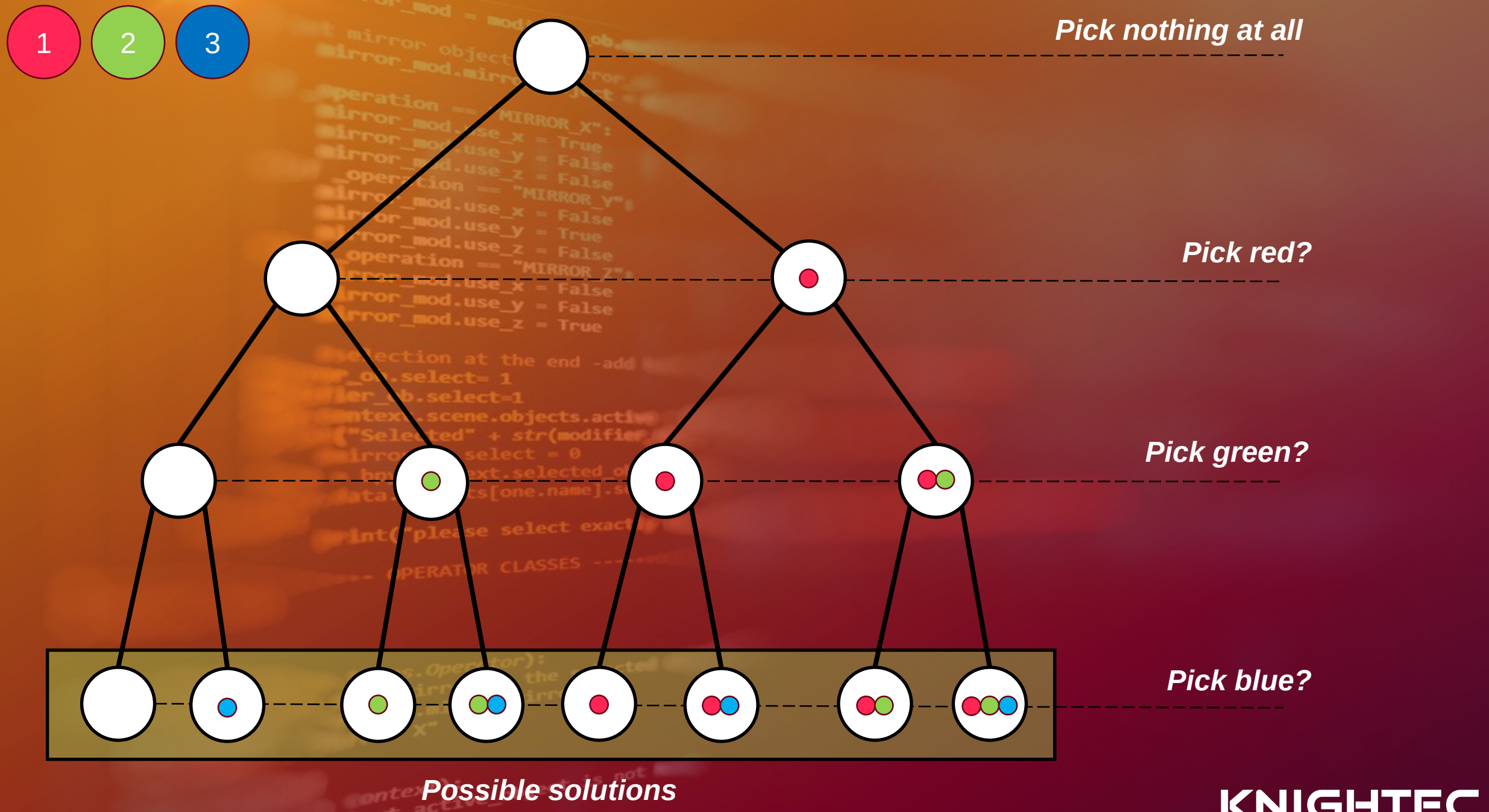
Knapsack has:

- *Weight limit*

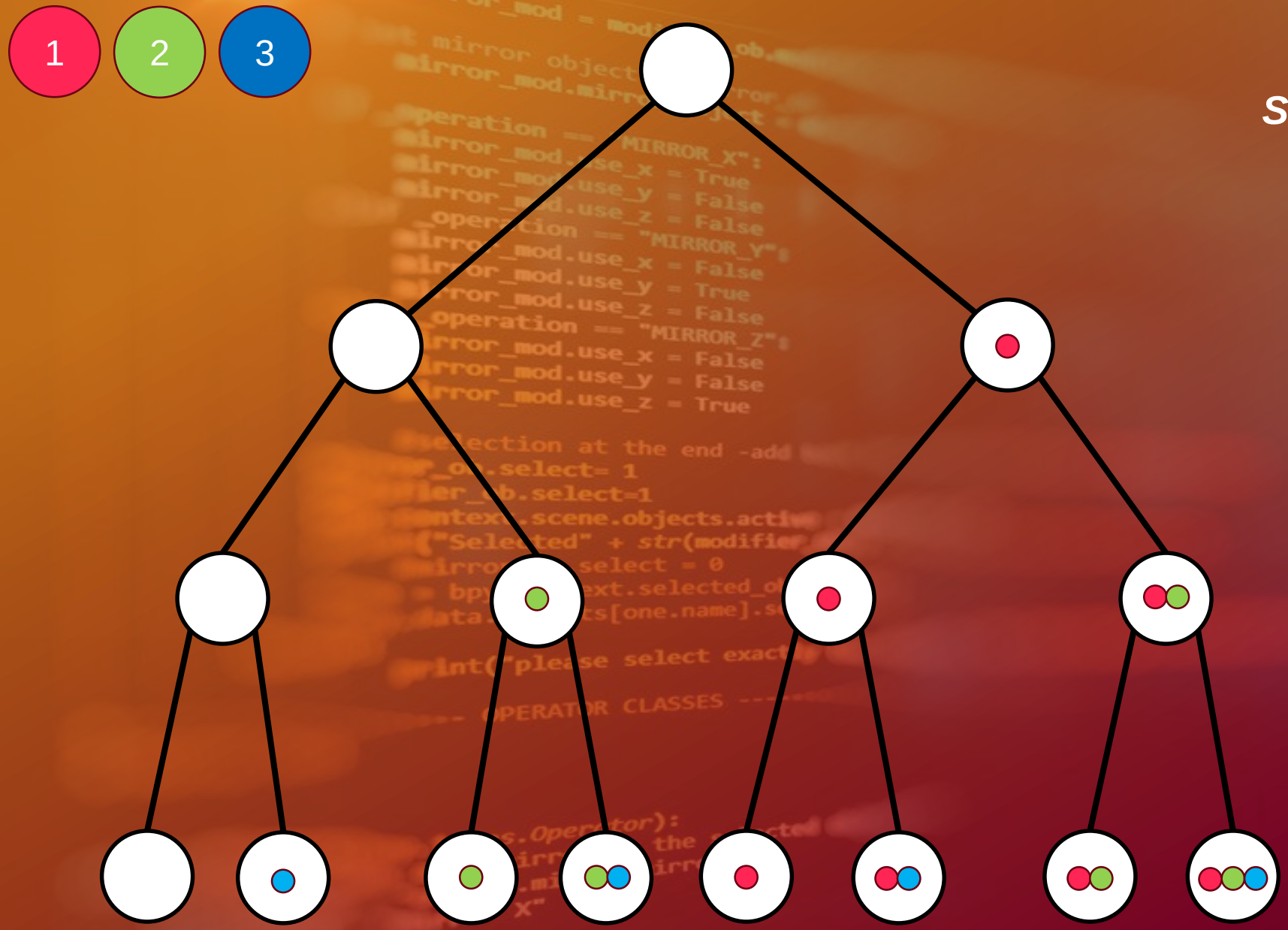
The problem:

What items do we put in the knapsack, for maximum total value?

KNIGHTEC


1
2
3
Pick nothing at all
Pick red?
Pick green?
Pick blue?
Possible solutions

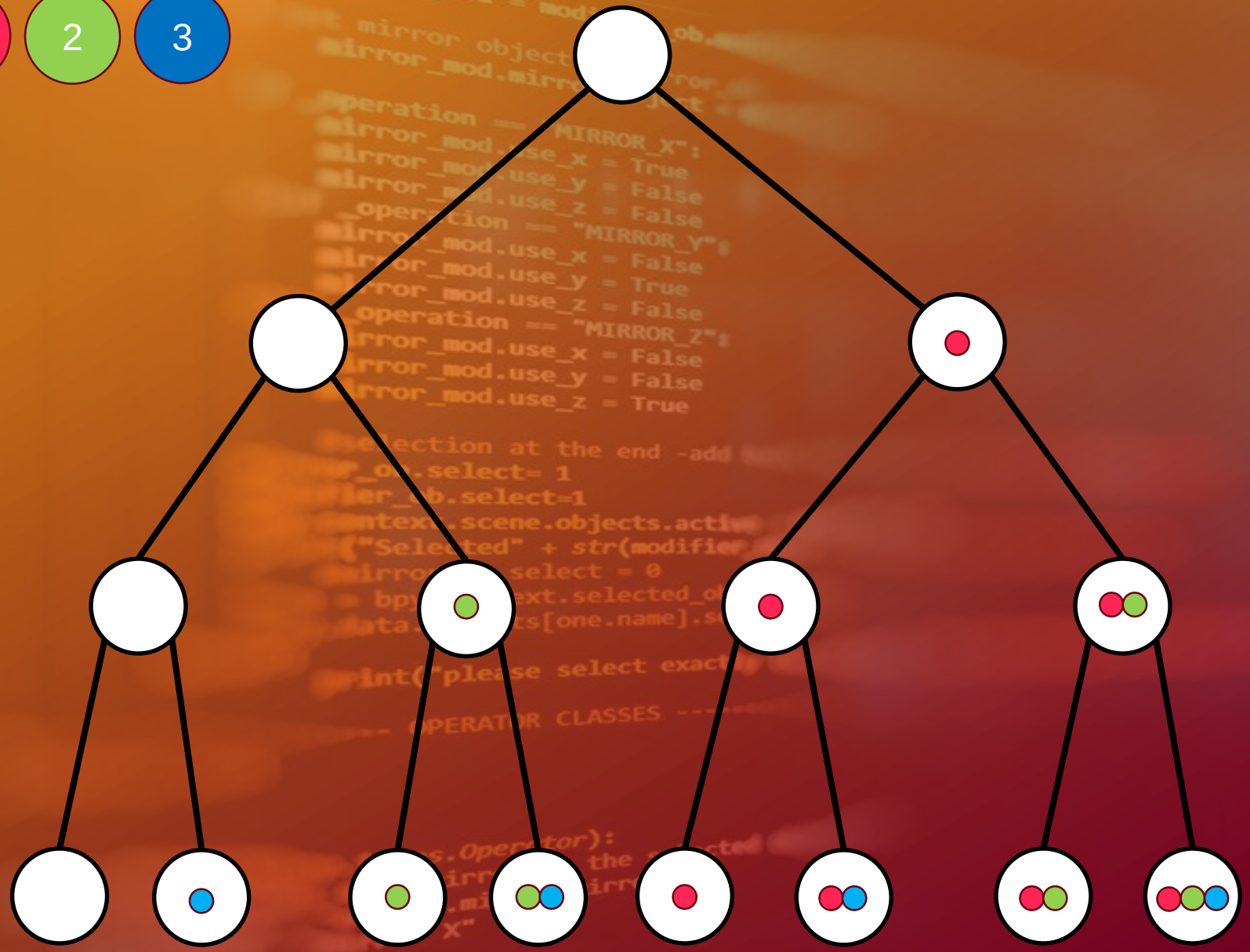
KNIGHTEC



$$Solutions = n^m = 2^3 = 8$$

KNIGHTEC

1 2 3



Tree traversal

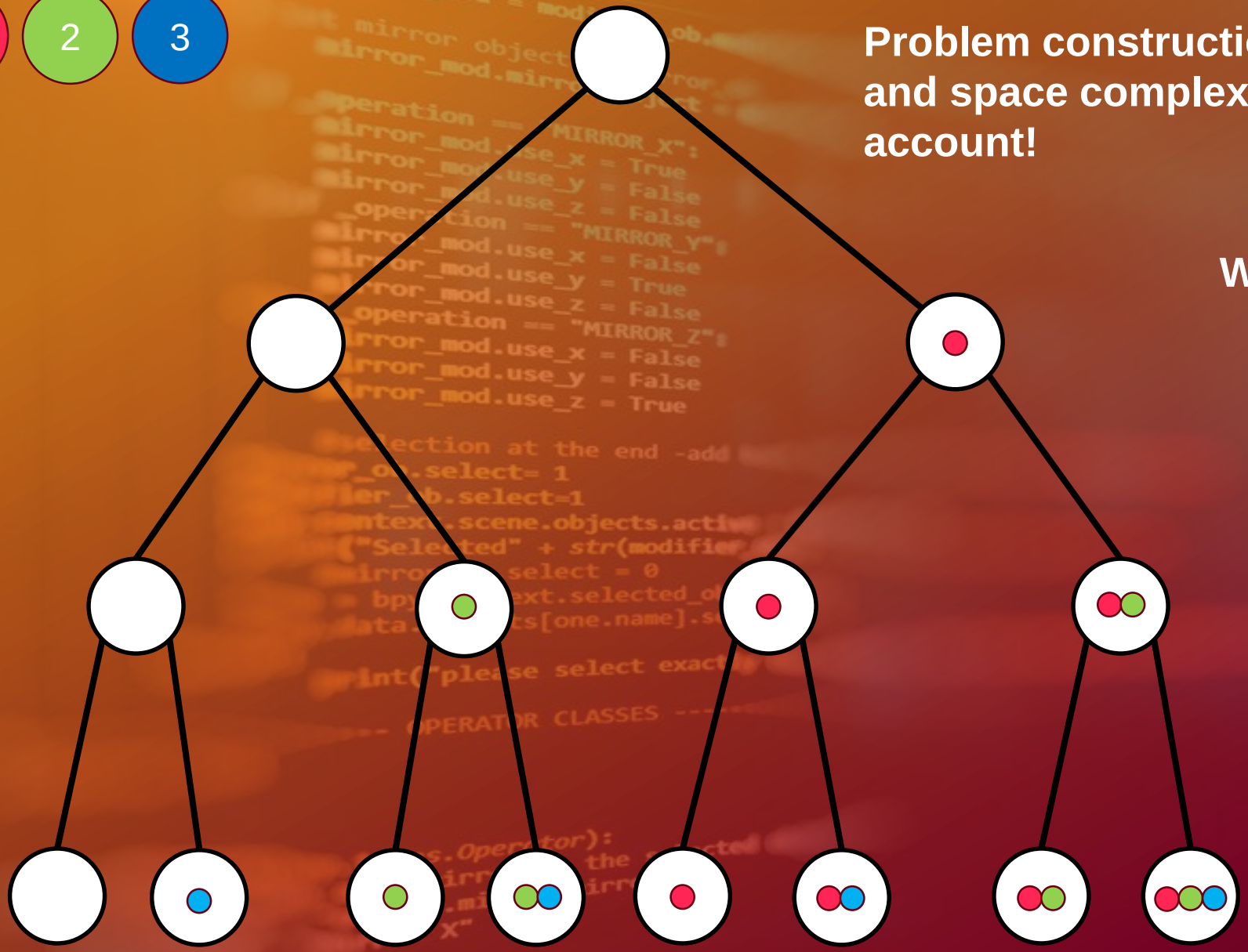
- *Depth First Search*
- *Breadth First Search*
- *Morris in-order*
- *…*

KNIGHTEC

Demo

KNIGHTEC

1 2 3

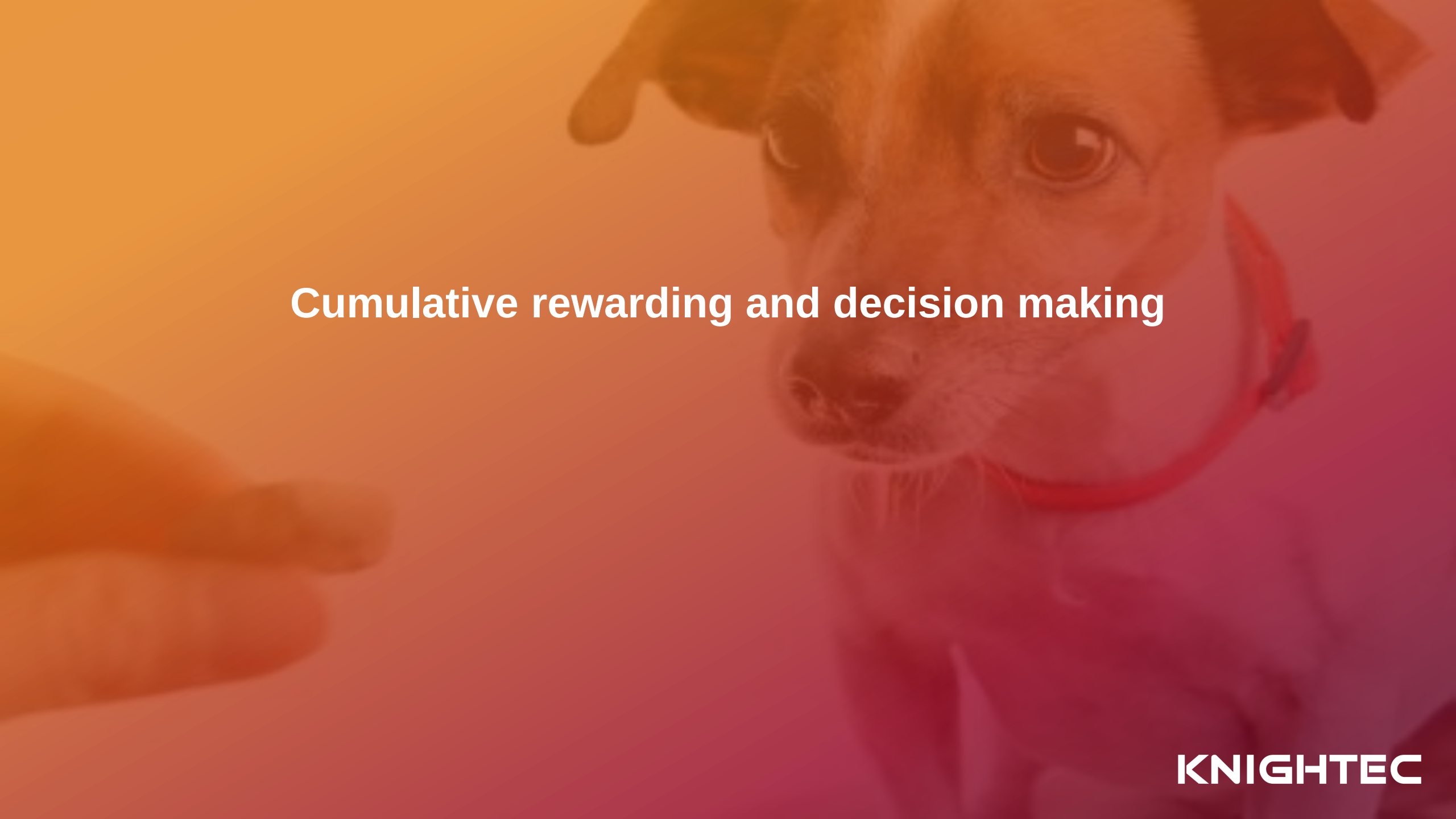


Problem construction is important, and time- and space complexity must be taken into account!

Why is BFS worse then DFS?

KNIGHTEC

Cumulative rewarding and decision making



KNIGHTEC

How can the ghosts know how to reach Pac-man?

Blinky the ”chaser”

Pinky the ”ambusher”

Inky the ”random”

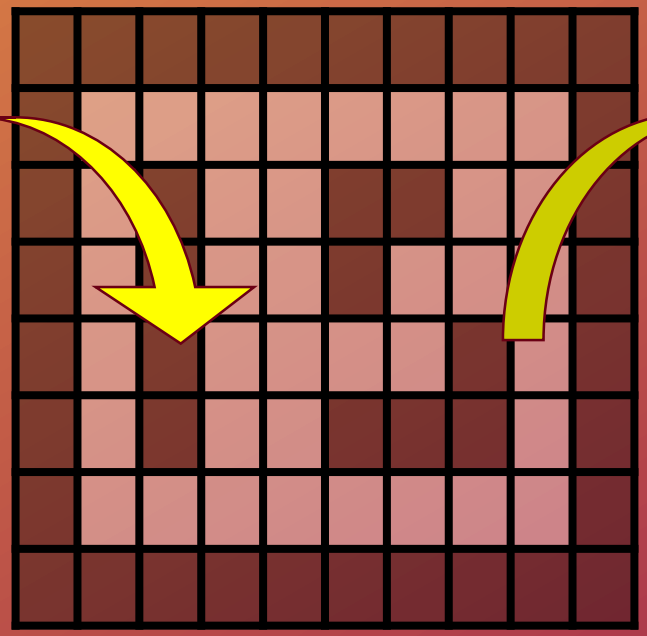
Clyde the ”switcher”

KNIGHTEC

Numerical matrix

1	2	2	2	2	2	2	2	2	3
4	0	0	0	0	0	0	0	0	8
4	0	X	0	0	X	X	0	0	8
4	0	X	0	0	X	0	0	0	8
4	0	X	0	0	0	0	X	0	8
4	0	X	0	0	X	X	X	0	8
4	0	0	0	0	0	0	0	0	8
5	6	6	6	6	6	6	6	6	7

Class-based representation



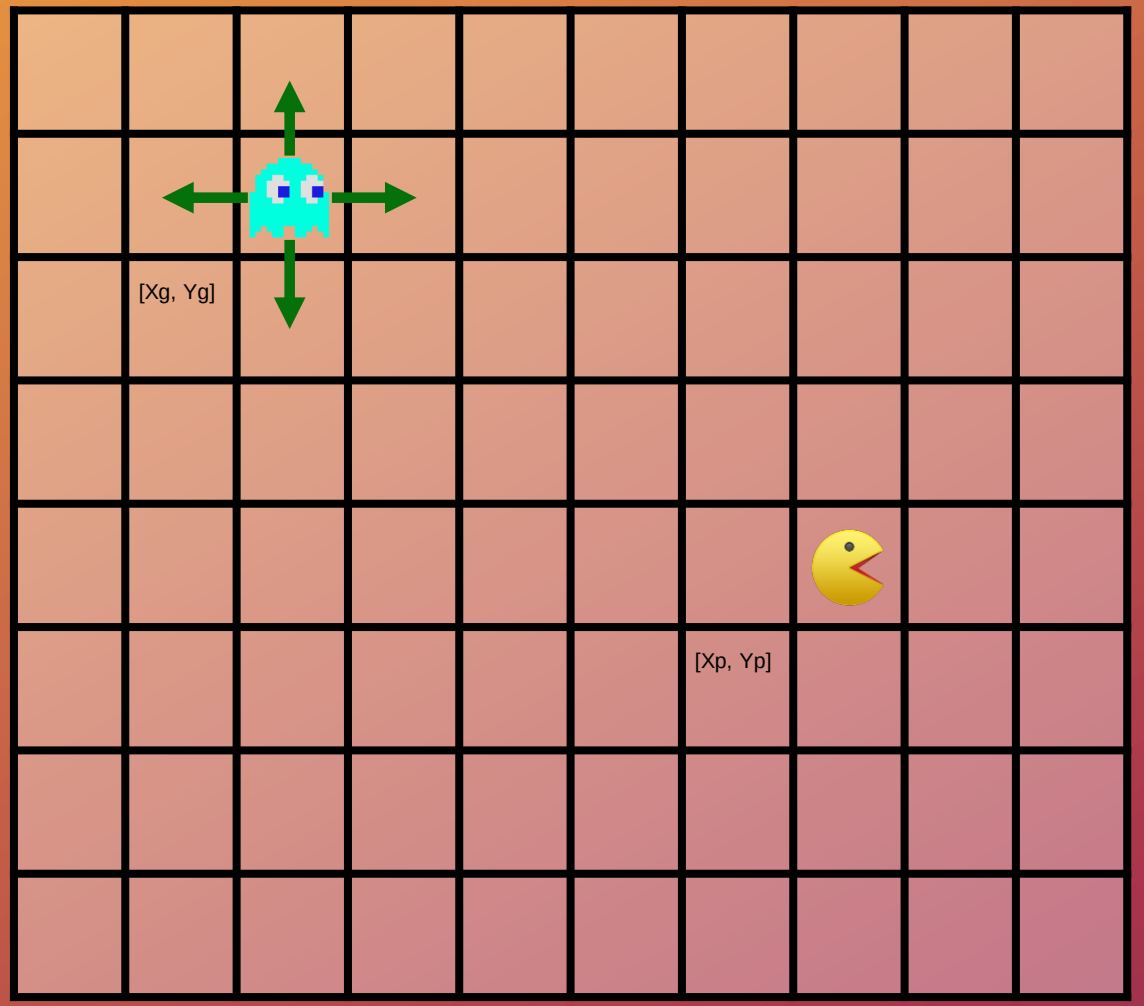
Reward-based construct, based on current states

X	X	X	X	X	X	X	X	X	X
X	3		3	5	3	5	3	2	X
X	5	X	6	7	X	X	5	4	X
X	9	X	5	7	X	6	5	7	X
X	3	X	3	4	4	6	X	6	X
X	3	X	2	1	X	X	X	4	X
X	5	6	1	3	1		2	3	X
X	X	X	X	X	X	X	X	X	X

KNIGHTEC

Ghost

- *Pac-man position*
- *Ghost position*

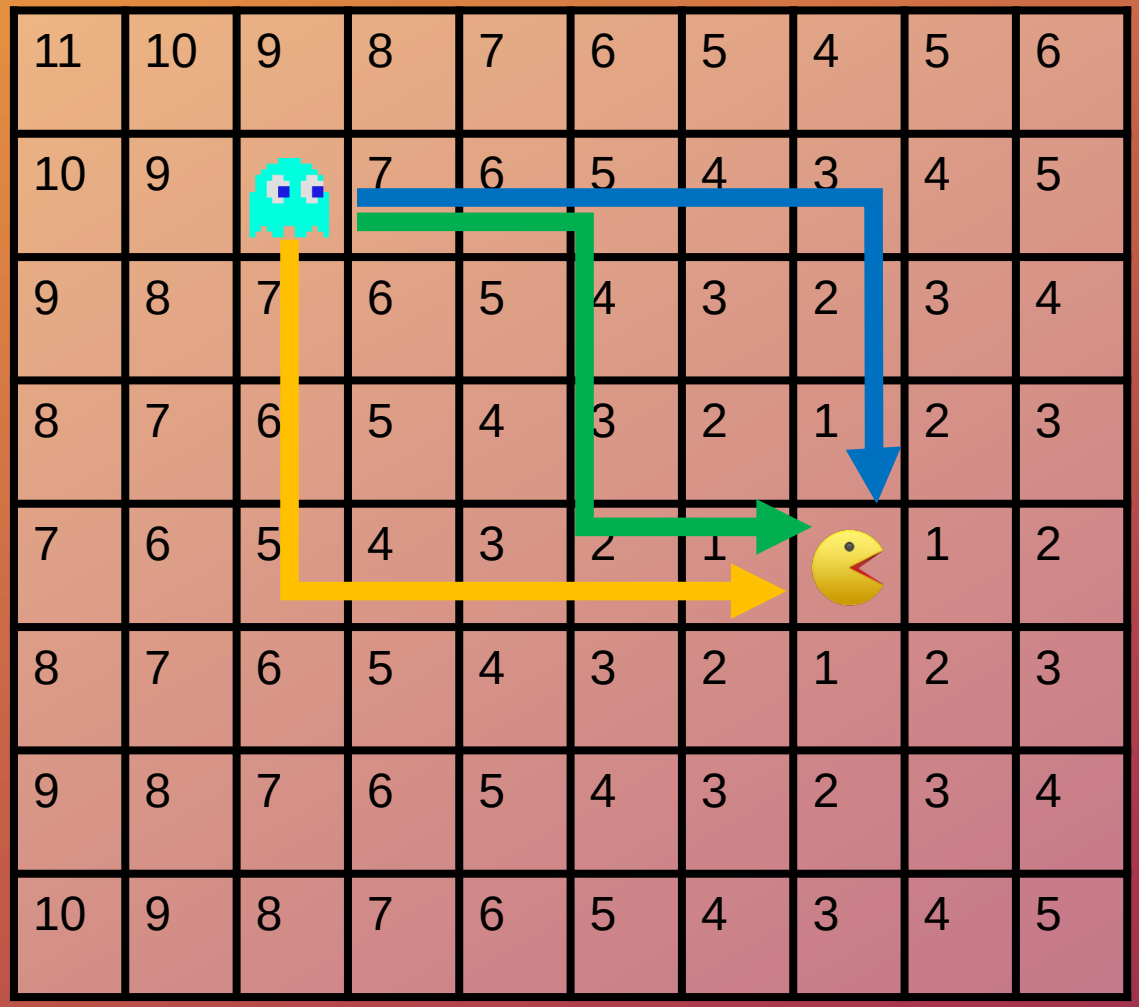


Square-class

- *Position [X, Y]*
- *Expanded/visited?*
- *Possible directions*
- *Actions performed from this square*

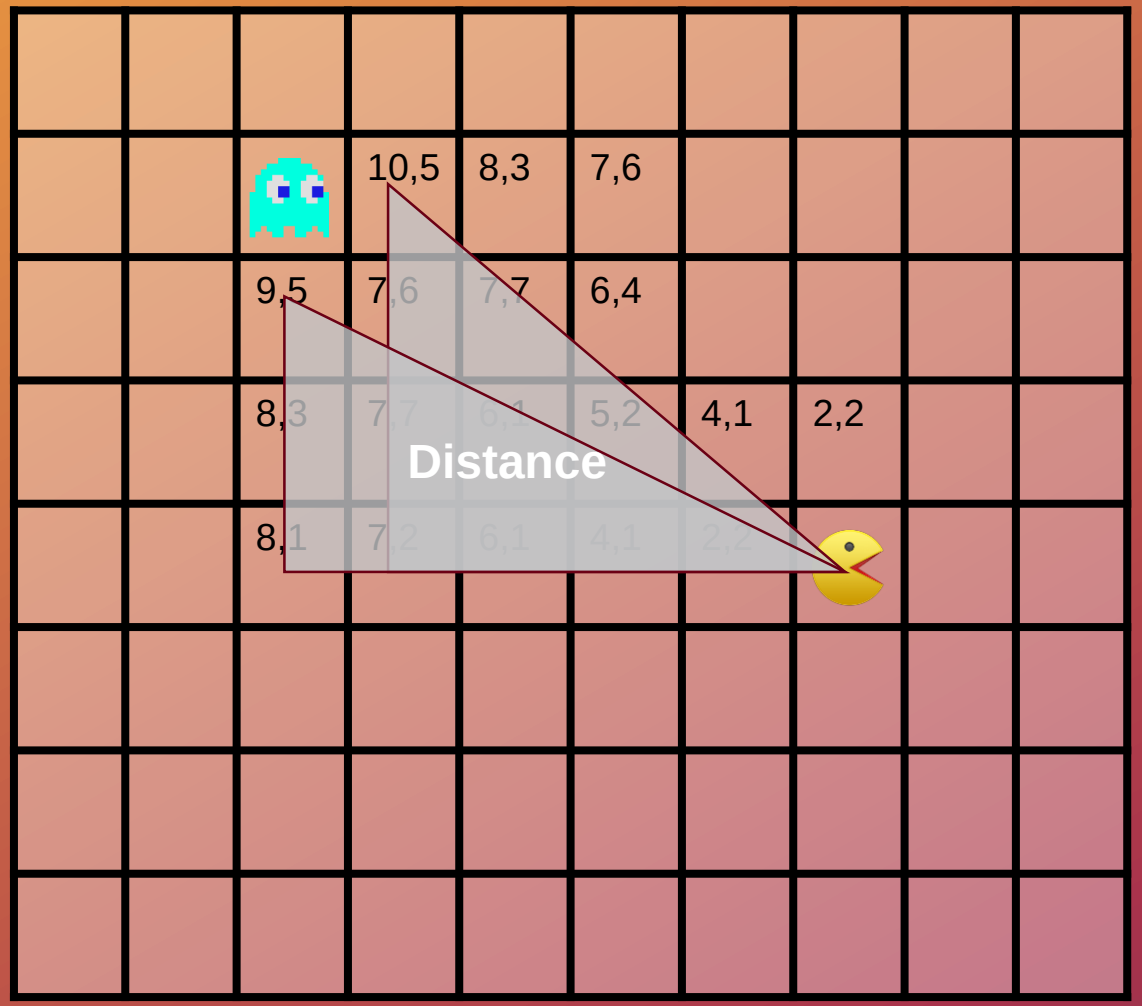
KNIGHTEC

How many steps are required to reach Pac-man?



KNIGHTEC

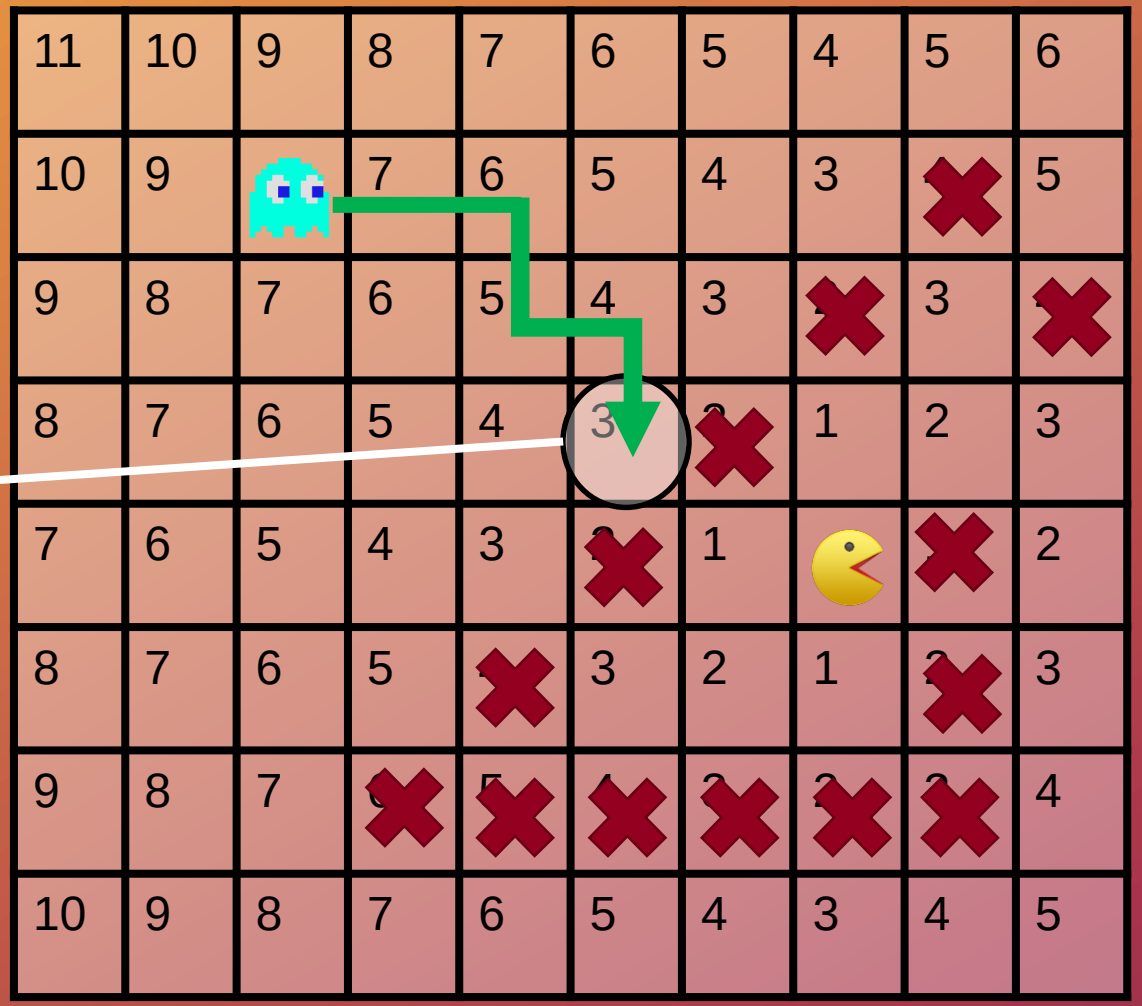
Birds-way-distance?



KNIGHTEC

How do we solve this??? 0o

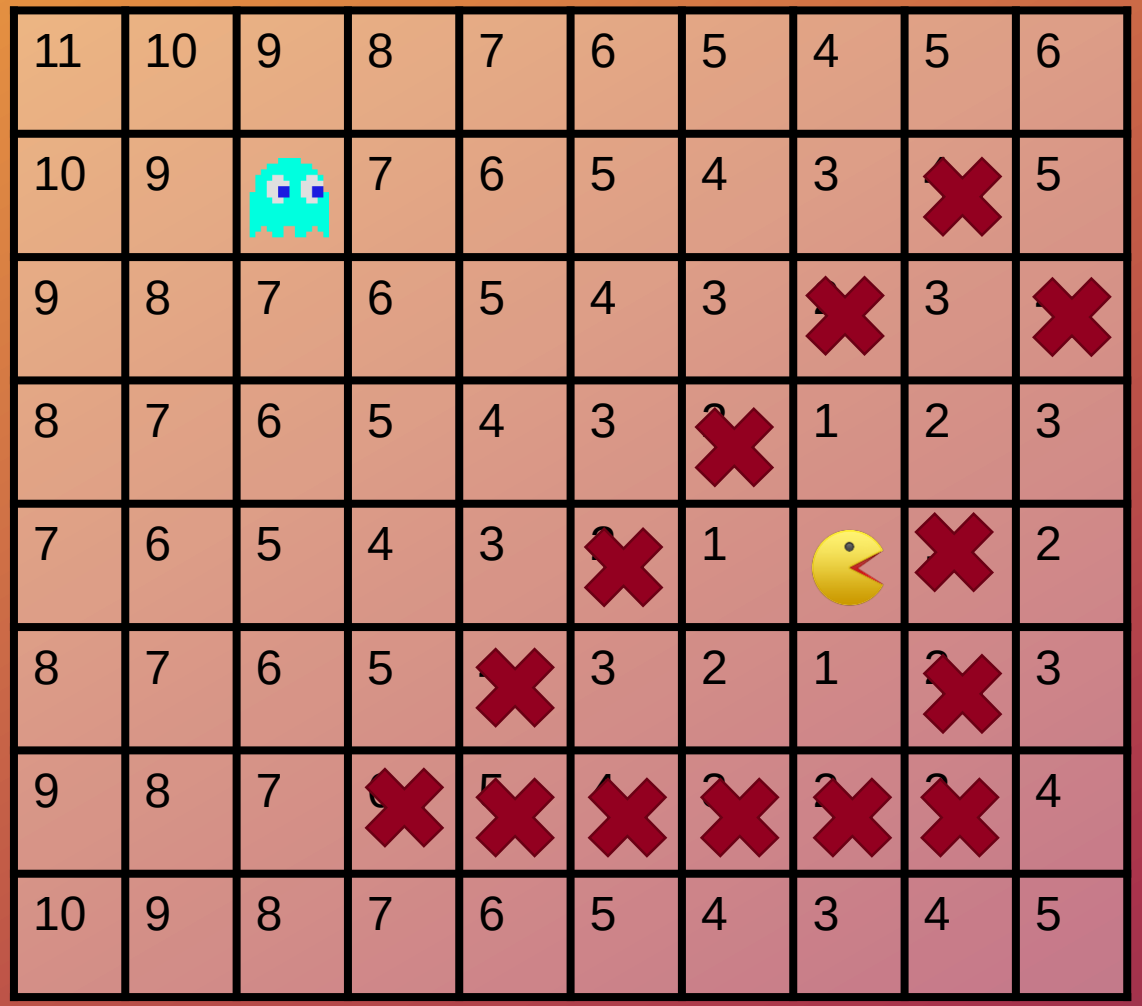
= Maze wall



This is appears to be the best way… but we are stuck!

KNIGHTEC

Best-first search, using A* and recursion



f(n) = g(n) + h(n)

Break limit

Solution count limit

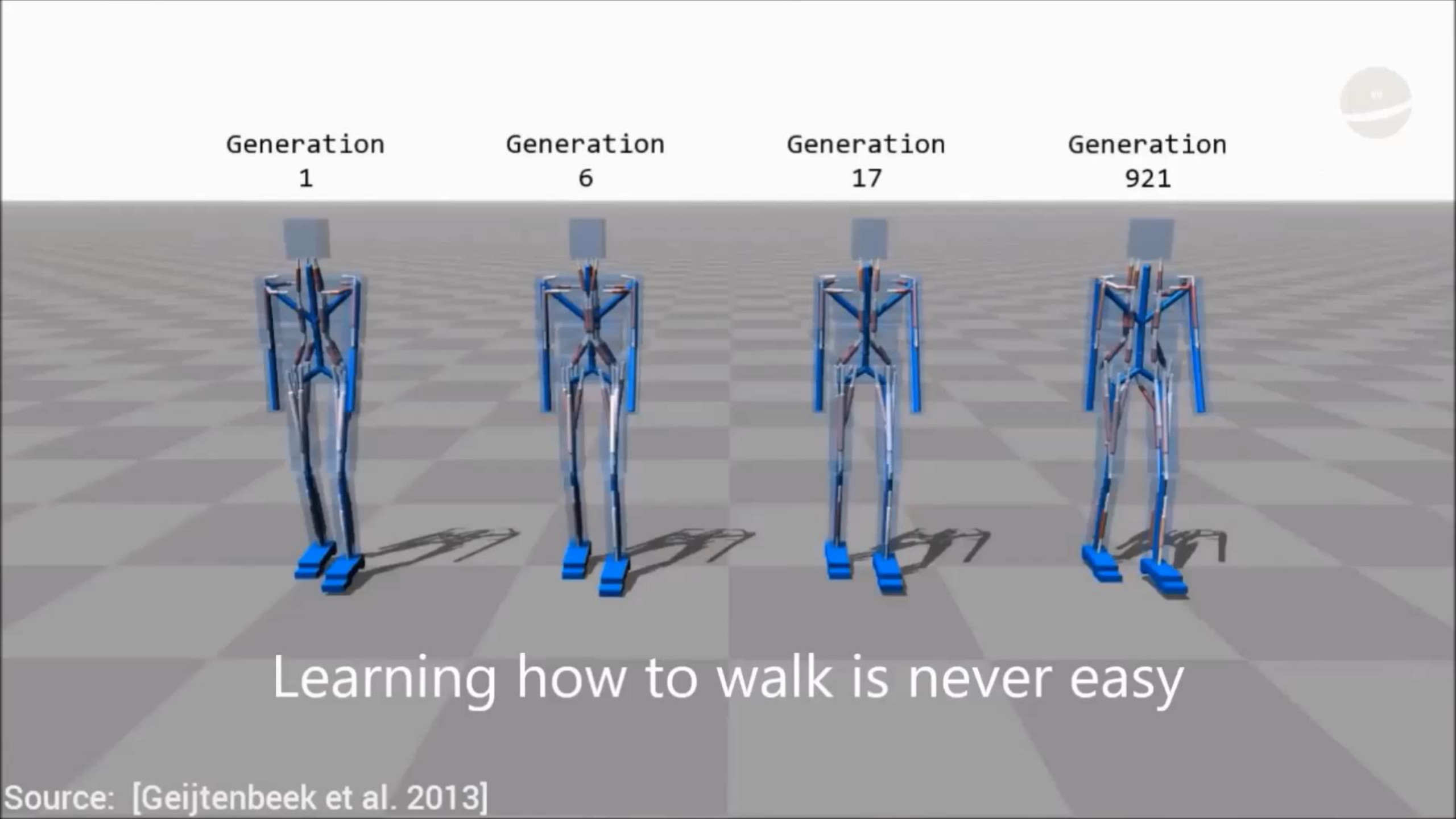
Number of steps limit

Random select

KNIGHTEC

Demo

KNIGHTEC


Generation 1
Generation 6
Generation 17
Generation 921
Learning how to walk is never easy
Source: [Geijtenbeek et al. 2013]

Recommended reading:

“Q-Learning”

“Intelligent agents”

Q –learning A famous unsupervised learning algorithm, used among other things, for learning how to play games such as Super Mario, through “Trial and error”.

Did you notice the example running on the computer in this office?

Q-learning baby;)

KNIGHTEC

GA - Evolution through ”Survival of the fittest”

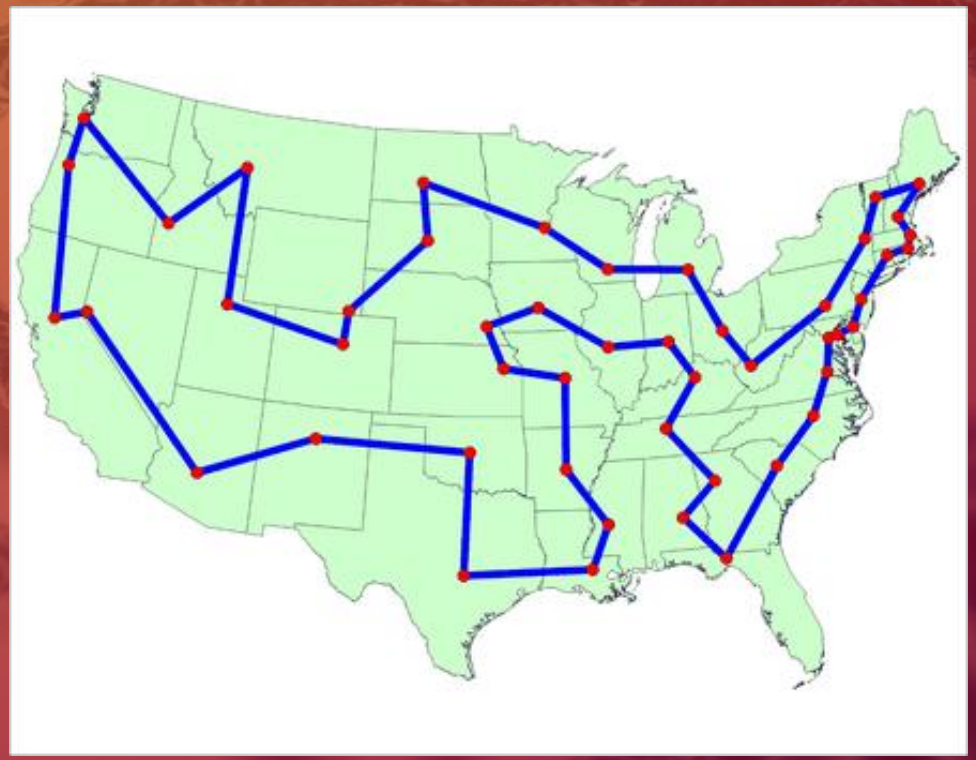
- Population of individuals – A breed or species to evolve
- Parents and offspring – The parents “compete” for the right to mate
- DNA/Chromosomes – Pass on the genetic inheritance to the offspring from both parents
- Mutation – Preserve genetic diversity (inbreeding control)
- Selection and Fitness function – Who will be allowed to breed?

KNIGHTEC

TSP – The Traveling salesman problem

Rules

- Visit all cities once, and once only
- Minimize the total traveled distance
- Start and end the journey in the same city

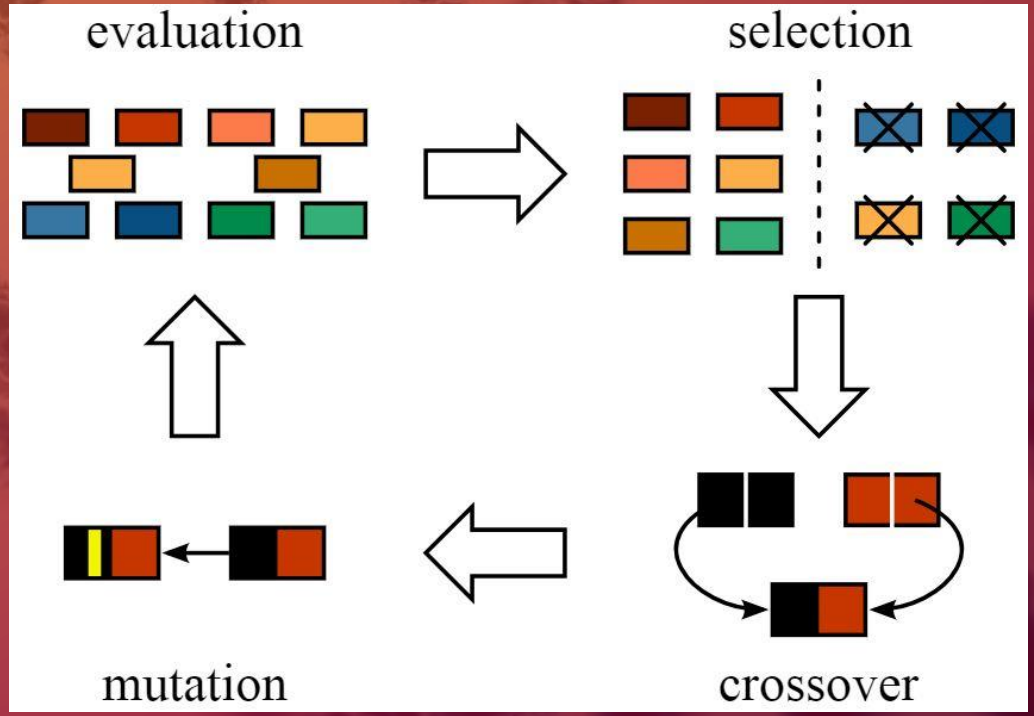


<u>Question</u> – In terms of TSP, what corresponds to

1. the DNA/chromosomes?
2. a solution?
3. a population?
4. a parent?
5. an offspring?
6. the fitness?

KNIGHTEC

1. Initiate the population randomly - Give each parent a random set of chromosomes (a list of cities/areas).
2. Select parents for breeding, based on the fitness function (best round-trip-distance)
3. Mate the set of selected parents together into couples
4. Pass DNA/genes to offspring, from both parents
5. Apply genetic mutations randomly on the new genetic composition
6. Repeat process on the new generation of individuals…

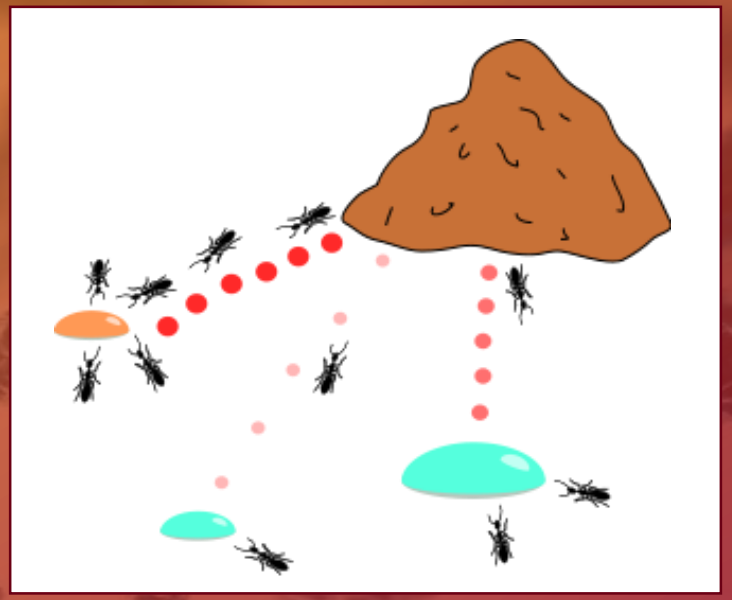


KNIGHTEC

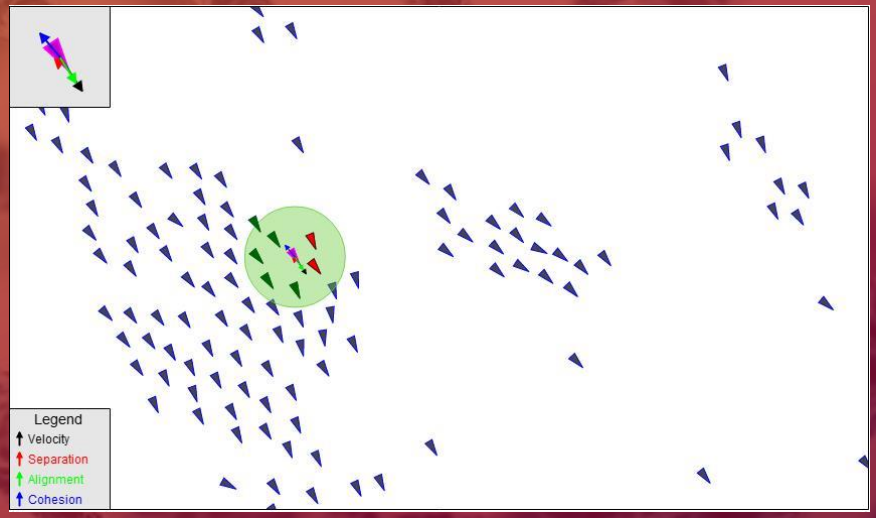
Demo

KNIGHTEC

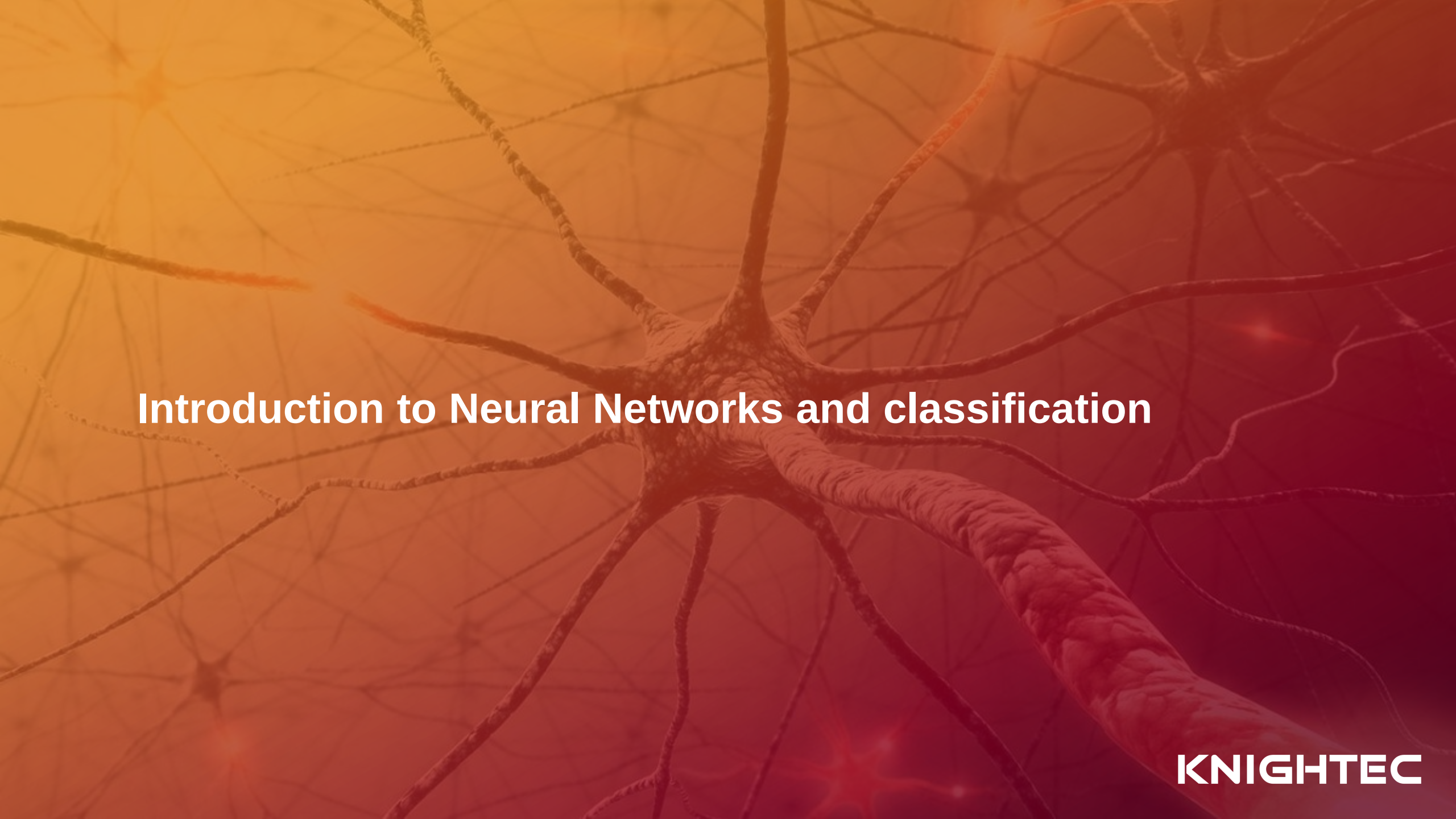
Recommended further reading



- “Ant colony”
- “Flocking Birds”



KNIGHTEC



Introduction to Neural Networks and classification

KNIGHTEC

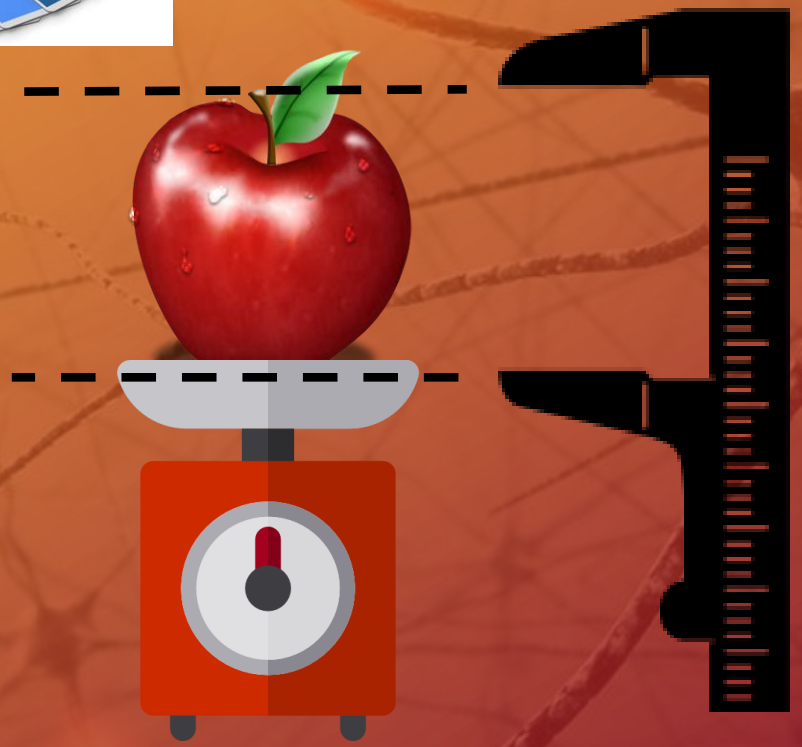
Imagine we would build a machine that could identify fruit

KNIGHTEC

What properties could be used by our machine, in order to distinguish fruits from each other?

Gather data

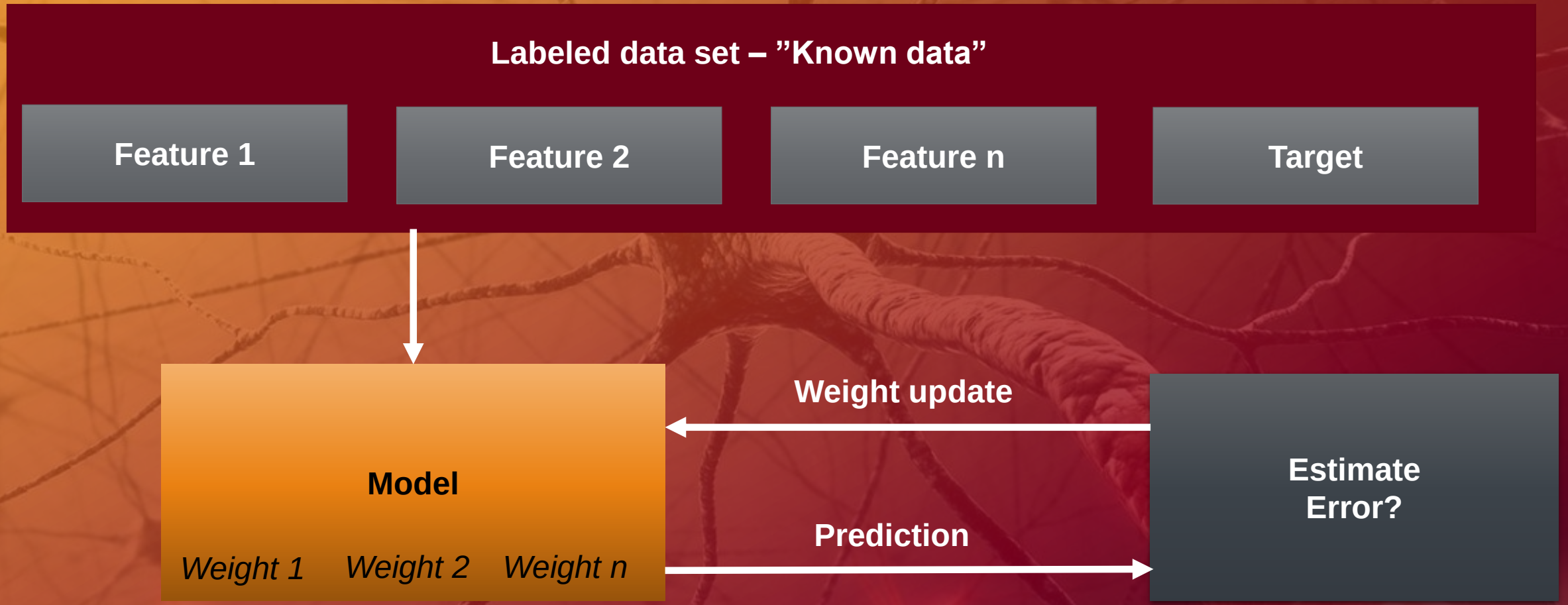
”Known data set”/ ”Labeled data”



Width	Height	Weight	Color	Calories	Target
161.34	140.32	150.6	[249, 29, 42]	154	Apple
139.2	154.9	143.78	[80, 236, 51]	161	Pear
110-56	120.3	200.05	[228, 139, 63]	115	Orange
143.01	161.84	161.6	[235, 25, 62]	134	Apple
205.2	356.09	532.56	[190, 121, 42]	536	Pineapple
…	…	…	…	…	…
…	…	…	…	…	…
…	…	…	…	…	…
…	…	…	…	…	…
…	…	…	…	…	…

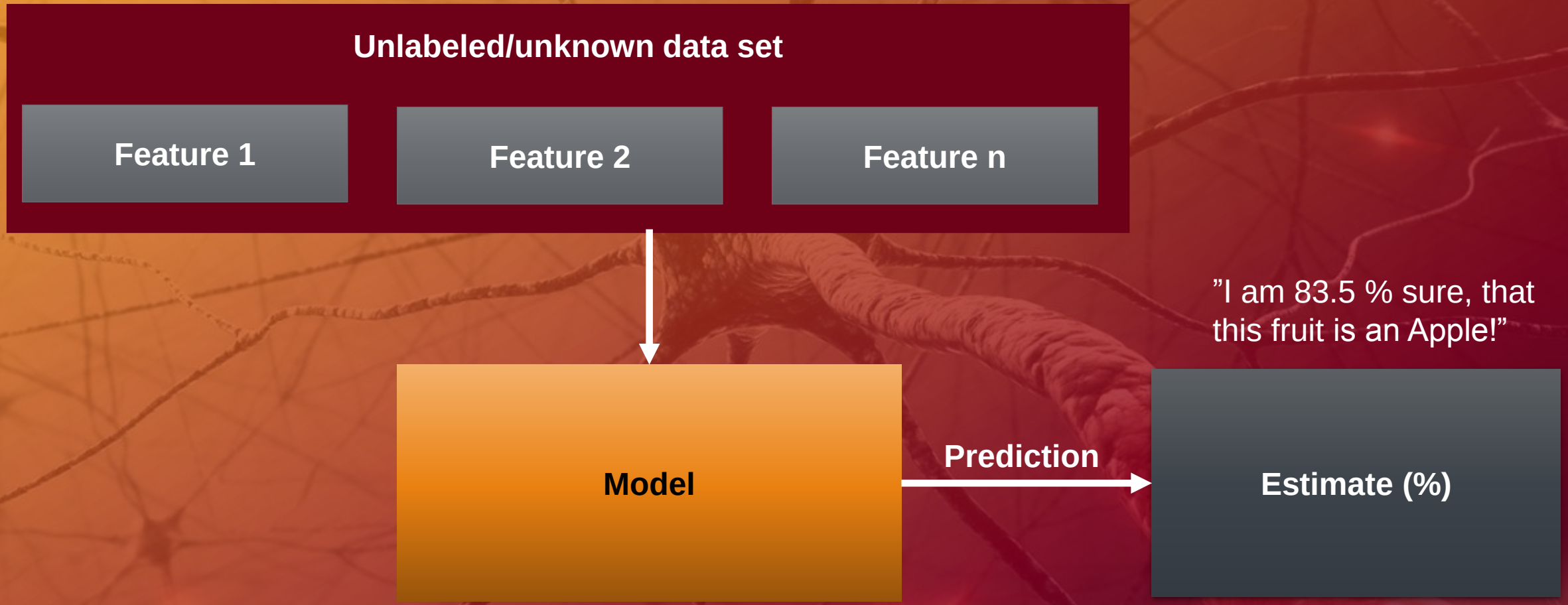
KNIGHTEC

Training and model construction



KNIGHTEC

Classification and prediction



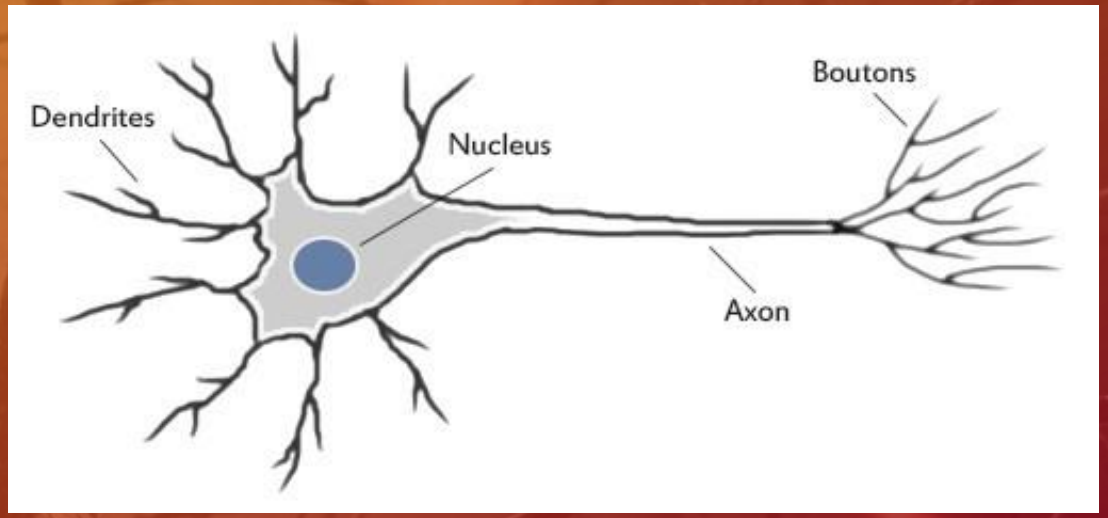
KNIGHTEC

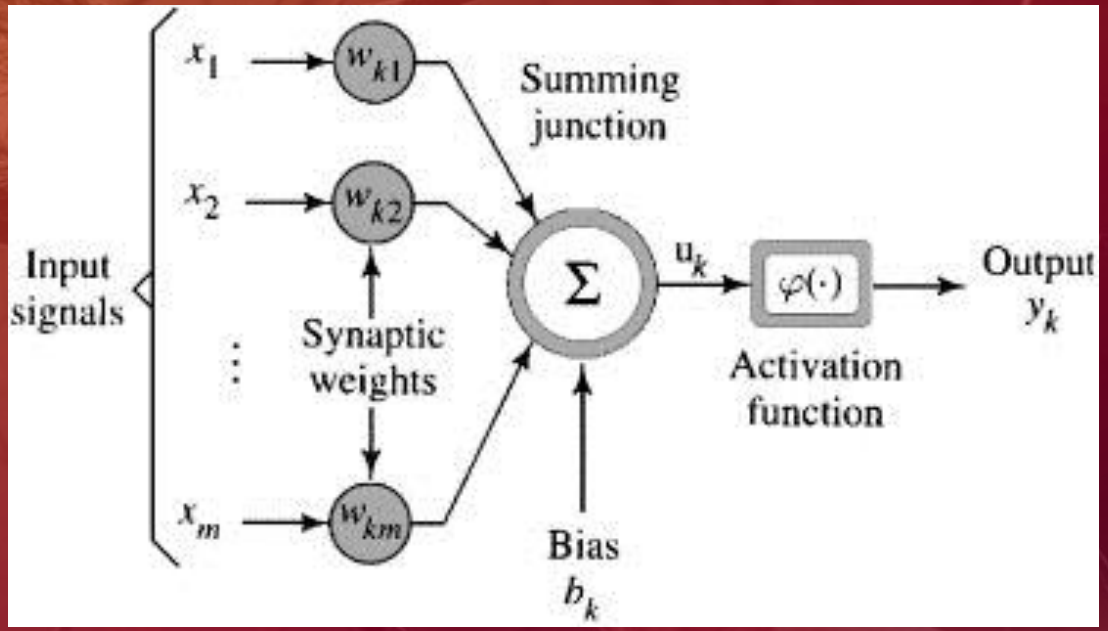
Artificial Neuron

We don’t have the required time to cover all the necessary math today, but I recommend you read about the different mathematical properties, and why they are used.

Don’t panic now! Just try to accept that there is more to the mathematical construct of ANN, then the simplified explanations that follows.

Let’s just try to get the basic idea =)





KNIGHTEC

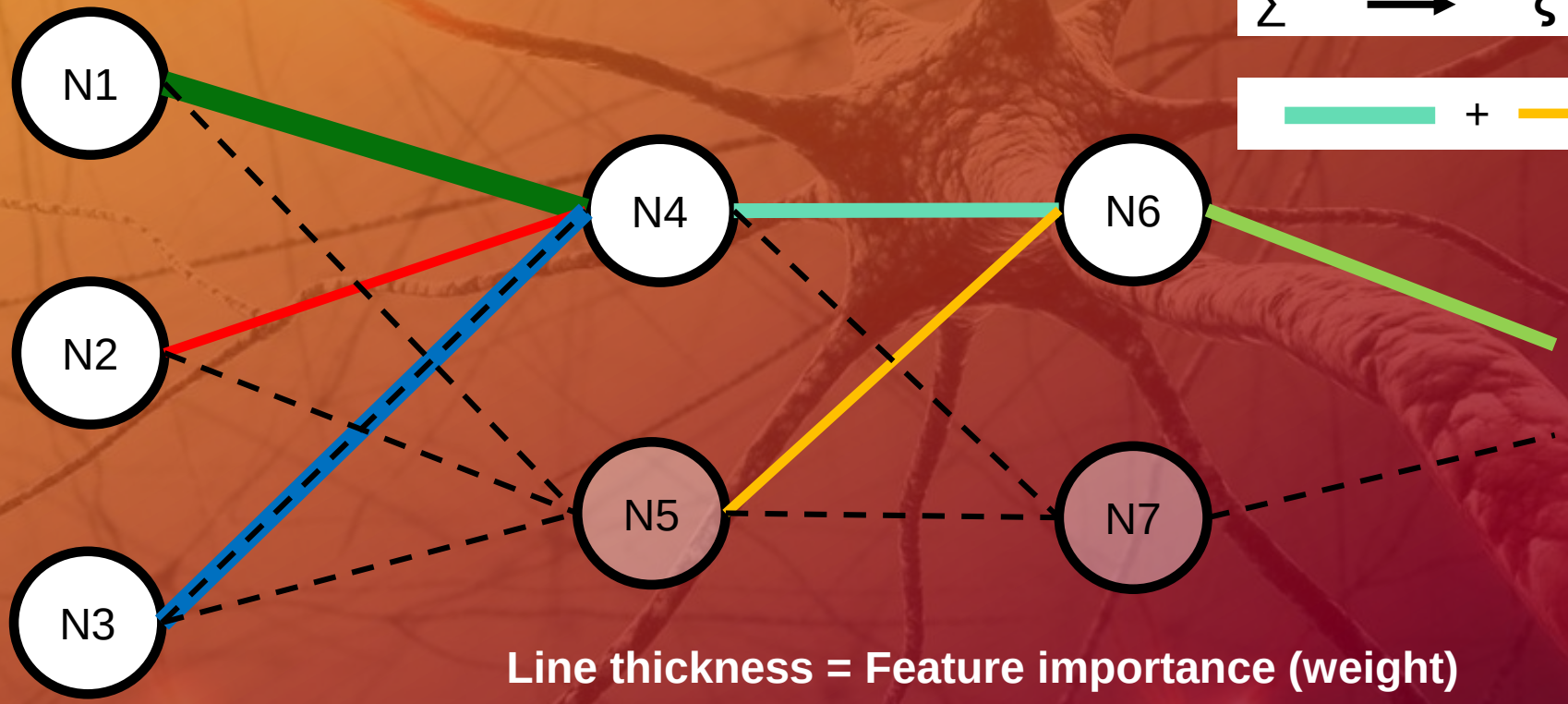
Neuron outputs in terms of RGB color

RGB[100, 220, 180] =

Feature	R	G	B
Weight (w)	100	220	180

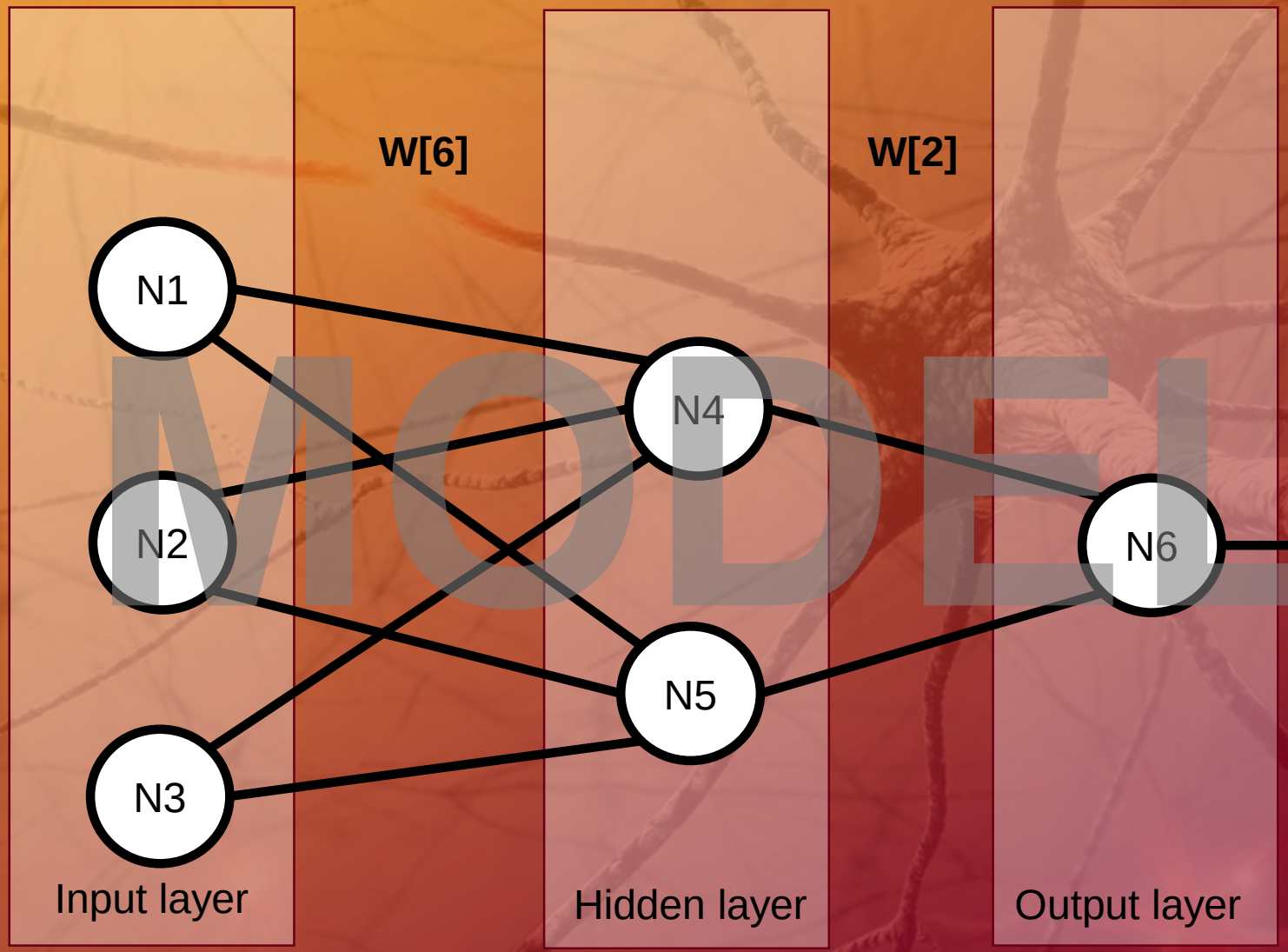
∑ → ς → **neuron output**

\+ =



Line thickness = Feature importance (weight)

KNIGHTEC



1) Set initial weights
2) Calculate layers output, left -> right
3) Update layer weights, right->left, with respect to the estimation error.

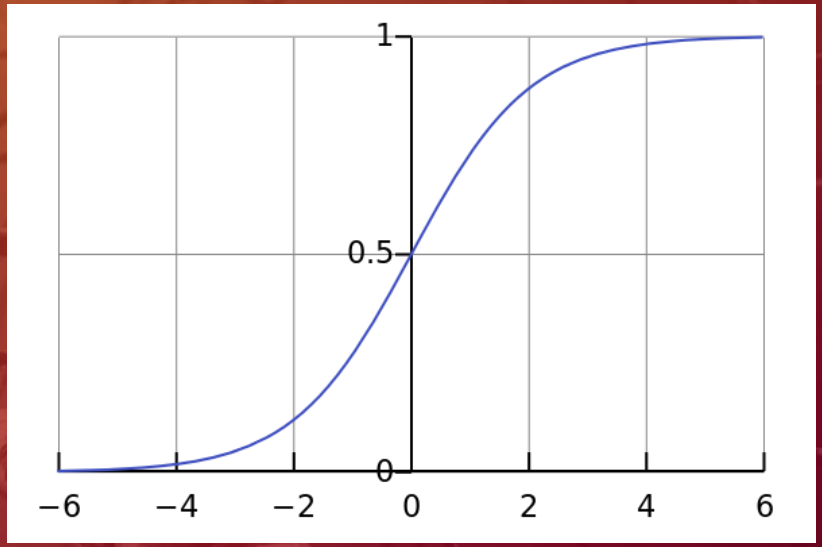
KNIGHTEC

Neuron output, and the “Sigmoid” activation function

“An acceptable mathematical representation of a biological neurons behavior.”

There is a lot more to it, but in short, we use this mathematical property to get an estimation in percentage (%), from the neuron output.

This could for example correspond to *“how likely it is, that this particular fruit is an apple”.*



Other activation functions exists, and depending on the desired neuron output characteristics, a different activation function could be used (step function, linear function, etc).

KNIGHTEC

How do we change our weights?

Backpropagation and gradient decent

“Error function” - Difference between the estimate and the target value.

“Learning rate” – A numeric property on how “fast/steep” a correction will be (how aggressive the change of the updated weights will be).

The resulting ΔW is subject to both the estimation error, and the learning rate.

In the backpropagation, we change our weights with a calculated ΔW.

New_W = prev_W + ΔW (simplified notation)

KNIGHTEC

Demo

KNIGHTEC

How could we use these tools on previous examples?

- The autonomous robots and the simulation environments?
- The Pac-man game?
- The TSP example?
- Titanic example?

Examples will be available in a GIT-repository for you to try out after the event. Experiment! =)

KNIGHTEC

KNIGHTEC

THANK YOU FOR ATTENDING