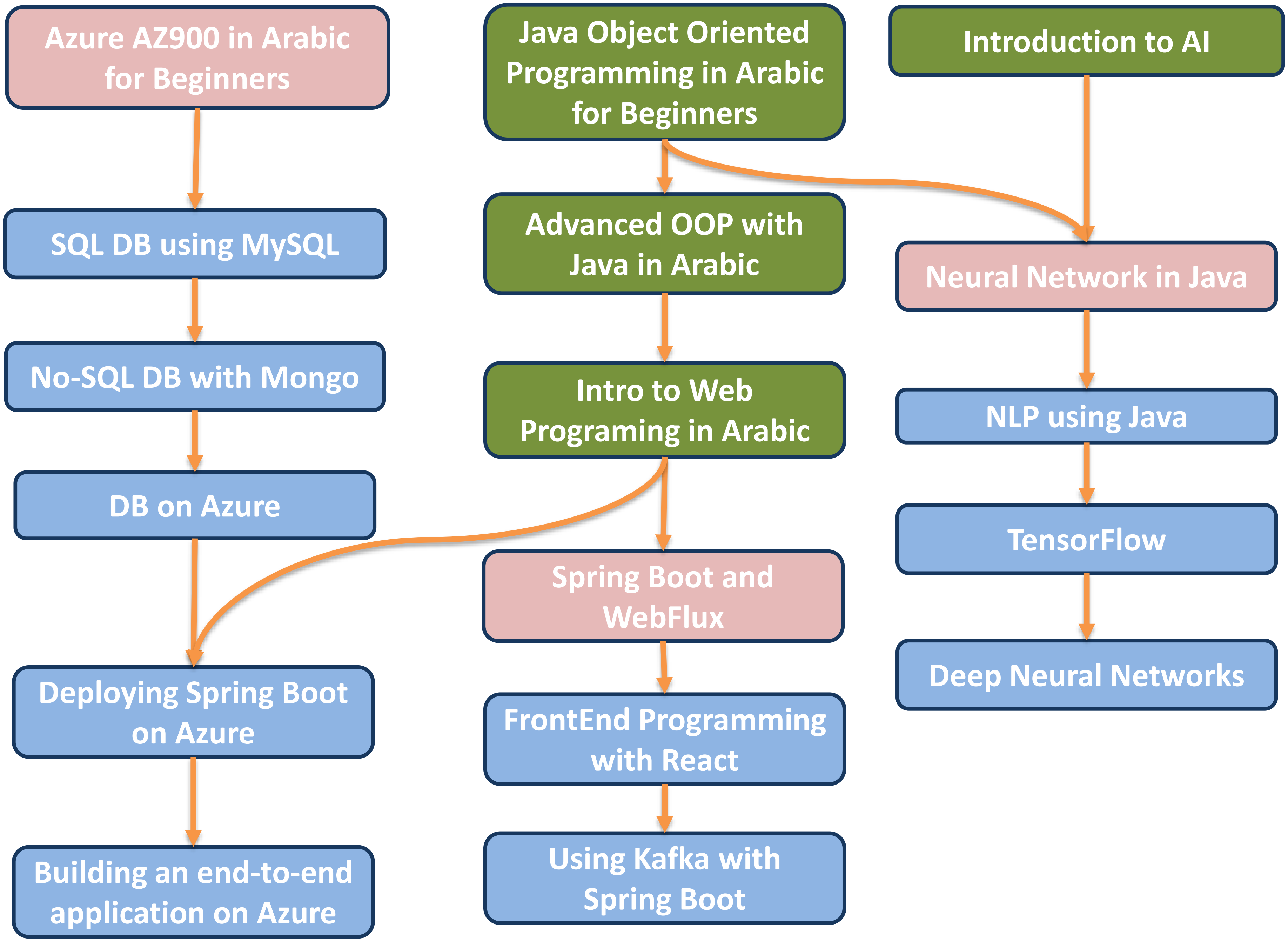

Azure AZ900 in Arabic for Beginners
SQL DB using MySQL
No-SQL DB with Mongo
DB on Azure
Deploying Spring Boot on Azure
Building an end-to-end application on Azure
Java Object Oriented Programming in Arabic for Beginners
Advanced OOP with Java in Arabic
Intro to Web Programing in Arabic
Spring Boot and WebFlux
FrontEnd Programming with React
Using Kafka with Spring Boot
Introduction to AI
Neural Network in Java
NLP using Java
TensorFlow
Deep Neural Networks

Introduction to Artificial Intelligence

Dr. Magdi AMER

Why AI

- ***Question:*** Is a word processor an AI application?
  - ***Answer:*** it depends
- When to use AI?
  - Search space is too big
  - Problem too complex to solve using equations
  - There are so many constrains that
    - finding a solution is difficult
    - finding the optimal solution is practically impossible

Classical AI
(Expert System)

First Order Logic

- All men are mortal.

- Socrates is a man.

- Conclusion: Socrates is mortal.

- For all X; if X is a bird $\rightarrow$ X has feathers

- Brady is a bird

- Conclusion: Brady has feathers

What if Brady is a penguin?

Expert Systems

- Set of facts
- Set of rules used to produce new facts
- 2 modes:
  - Forward chaining
    - Produce all possible facts
  - Backward chaining
    - From the result, apply reverse rules till all pre-condition are true

Expert System Drawbacks

- Major drawback:
  - If the information is not there, (for example search if John is a Student), then we cannot deduce that the opposite is true (John is not a Student is unknown).
  - It is hard to represent exceptions
  - It is not possible to represent different opinions
  - It cannot deal with inconsistency

Reactive Architecture

Introduction to Reactive Architecture

- In mid 80s, to control a robot using classic AI:
  - All info from sensors written as facts
  - Use rules
    - to deduce the current state
    - to choose best action.
- In 1986, Rodney Brooks, a PhD student at MIT , built the first reactive architecture
  - divided the control of the robot into layers
  - each layer limited to a few sensors
  - each layer is responsible for an action (preventing collision, movement, …).
  - An upper layer decision overwrites a lower layer.

Introduction to Reactive Architecture

- With this simple system, Brooks was capable of realizing advanced functions that classical AI systems could not achieve.
- This was the first proof that alternative architectures can perform intelligent behavior better than classical AI.

1) Heuristics

Heuristics

- Code rules that comes from:
  - Knowledge of the Domain
  - Model of the entities of the domain knowledge, believes and motivations.

Heuristics: Using knowledge from the domain

- Solve the problem

  SEND + MORE = MONEY

  Where each letter represents a digit, and no repetition is allowed.
- If this problem is solved traditionally, a program will implement all the possible combinations and test each combination.
- Using some knowledge from the mathematical domain, the complexity of the problem can be significantly reduced.

Heuristics: Using knowledge from the domain

- Take the most significant digits:
  - S+M+(1) = 10*M + O ➔ +(1): in case of a carry on
  - M must be 1 ➔ S is either 9 or 8, O is 0 or 1
  - But O cannot be 1 (no repetition)➔ O = 0

S E N D	➔	8 or 9 E N D
M O R E		1 0 R E
M O N E Y		1 0 N E Y

Heuristics: Using knowledge from the domain

- E + 0 +(1) = N ➔ no repetition ➔ E+1 = N
- And E cannot be 9 (N bigger single digit)

  ➔E+ 0 cannot generate a carry on

  ➔S must be equal to 9

S E N D	➔		9	E	E+1	D
M O R E			1	0	R	E
M O N E Y		1	0	(E+1)	E	Y

Heuristics: Using knowledge from the domain

- E + 1 + R + (1) = E + 10 $\rightarrow$ R = 9 - (1)
  - But R cannot be 9 $\rightarrow$ R = 8
- D + E = Y + 10
  - D + E > 10 $\rightarrow$ 9, 8 taken, N = E +1
    - E max value = 6
    - If E = 6, D = 5 $\rightarrow$ Y = 1 but 1 is taken
    - Let E = 5 $\rightarrow$ N = 6 and D = 7 $\rightarrow$ Y =2
- 9567+1085 = 10652

Heuristic:
Using the model of the entities

Give it a try with the following problem. Two men meet on the street. They haven't seen each other for many years. They talk about various things, and then after some time one of them says: "Since you're a professor in mathematics, I'd like to give you a problem to solve. You know, today's a very special day for me: All three of my sons celebrate their birthday this very day! So, can you tell me how old each of them is?"

"Sure," answers the mathematician, "but you'll have to tell me something about them."

"OK, I'll give you some hints," replies the father of the three sons, "The product of the ages of my sons is 36."

"That's fine," says the mathematician, "but I'll need more than just this."

"The sum of their ages is equal to the number of windows in that building," says the father pointing at a structure next to them.

The mathematician thinks for some time and replies, "Still, I need an additional hint to solve your puzzle."

"My oldest son has blue eyes," says the other man.

"Oh, this is sufficient!" exclaims the mathematician, and he gives the father the correct answer: the ages of his three sons.

Heuristic: Using the model of the entities

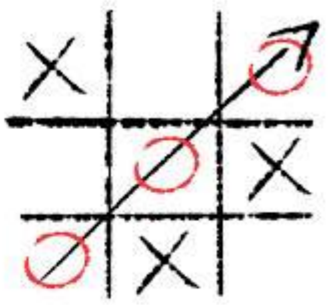
Age of son 1	Age of son 2	Age of son 3
36	1	1
18	2	1
12	3	1
9	4	1
9	2	2
6	6	1
6	3	2
4	3	3

$$
\begin{array}{ccccccc}
36 & + & 1 & + & 1 & = & 38 \\
18 & + & 2 & + & 1 & = & 21 \\
12 & + & 3 & + & 1 & = & 16 \\
9 & + & 4 & + & 1 & = & 14 \\
9 & + & 2 & + & 2 & = & 13 \\
6 & + & 6 & + & 1 & = & 13 \\
6 & + & 3 & + & 2 & = & 11 \\
4 & + & 3 & + & 3 & = & 10
\end{array}
$$

- If the number of windows was 38, 21, 16, 14, 11 or 10, the mathematician would have answered immediately.
- The fact that he needed another info, means that there are more than one solution. ➔ the number of windows is 13
- In the third info, the man has an Oldest son ➔ the ages are: 9, 2, 2

Heuristic in practice: Chess

- tic-tac-toe (X O)
  - Get tree of all possible actions
  - Cannot be defeated



- Chess
  - Tree is too large
  - Get sub-tree of all possible actions for X turns
  - Use heuristic to evaluate the positions
  - Choose the action whose worst-case scenario is best for you.

2) Fuzzy Logic

Fuzzy Logic Education Program

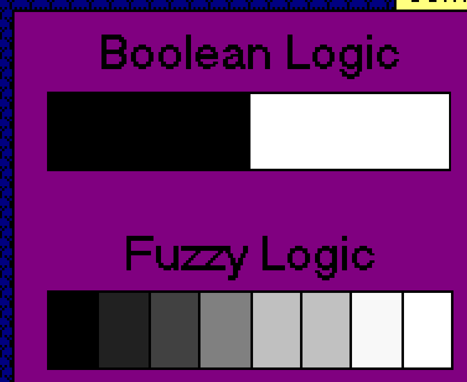
In the mid 1960's, Professor Lotfi Zadeh of the University of California at Berkeley recognized that the true-or-false nature of Boolean logic did not account for the many shades of gray found in the real world.

To account for the infinite gradations between true and false, Zadeh expanded the idea of a classical set to what he termed a fuzzy set. Unlike Boolean logic, fuzzy logic is multi-valued. Instead of an element being 100% this or that, or a proposition that is either entirely true or completely false, fuzzy deals in *degrees* of membership and *degrees* of truth--that is, something can be partially true and partially false at the same time.

It has been proven by Bart Kosko that Boolean logic is a special case of fuzzy logic.

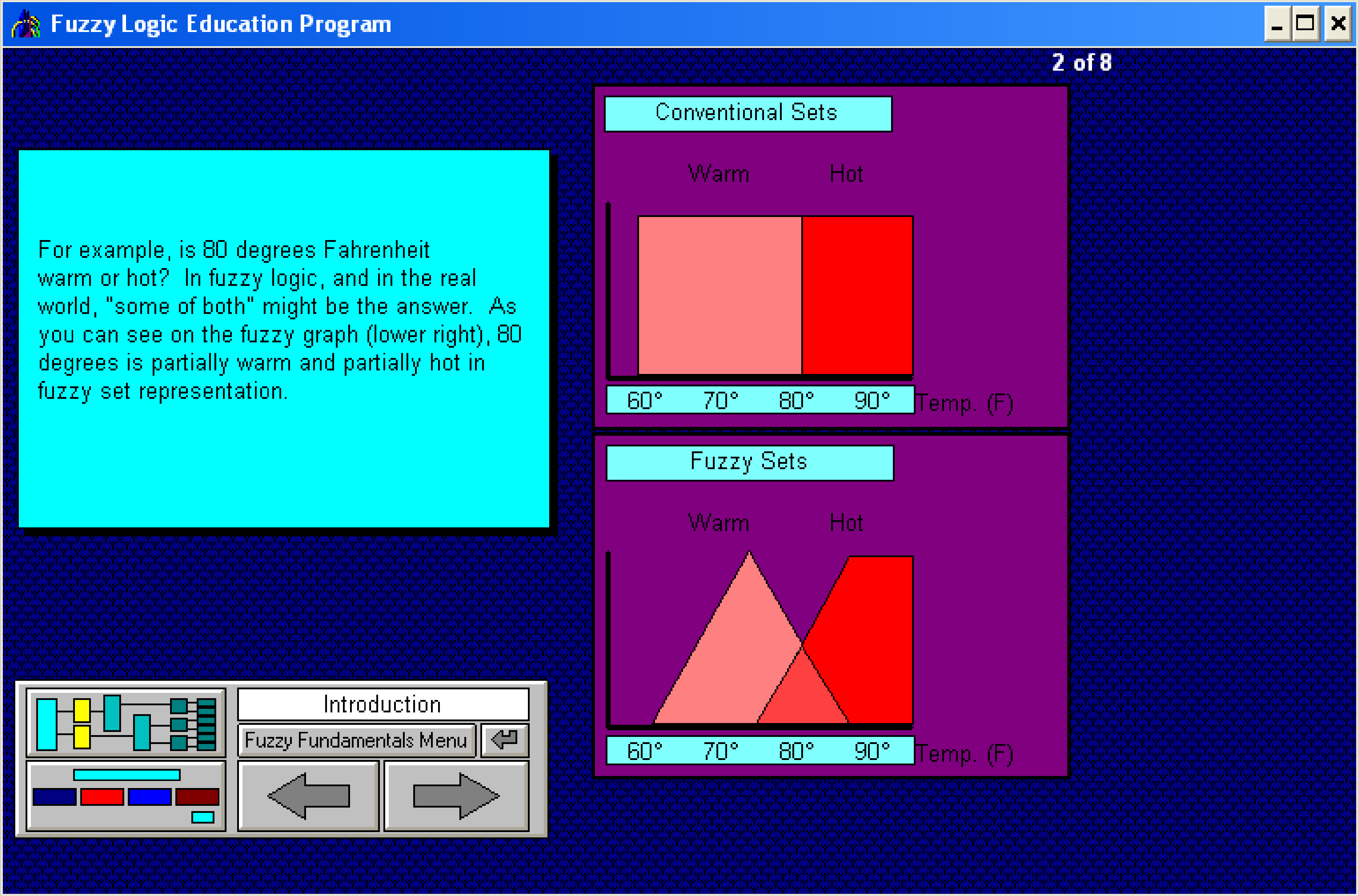
Lotfi Zadeh

Click here for Professor Zadeh's comments



Introduction

Fuzzy Fundamentals Menu

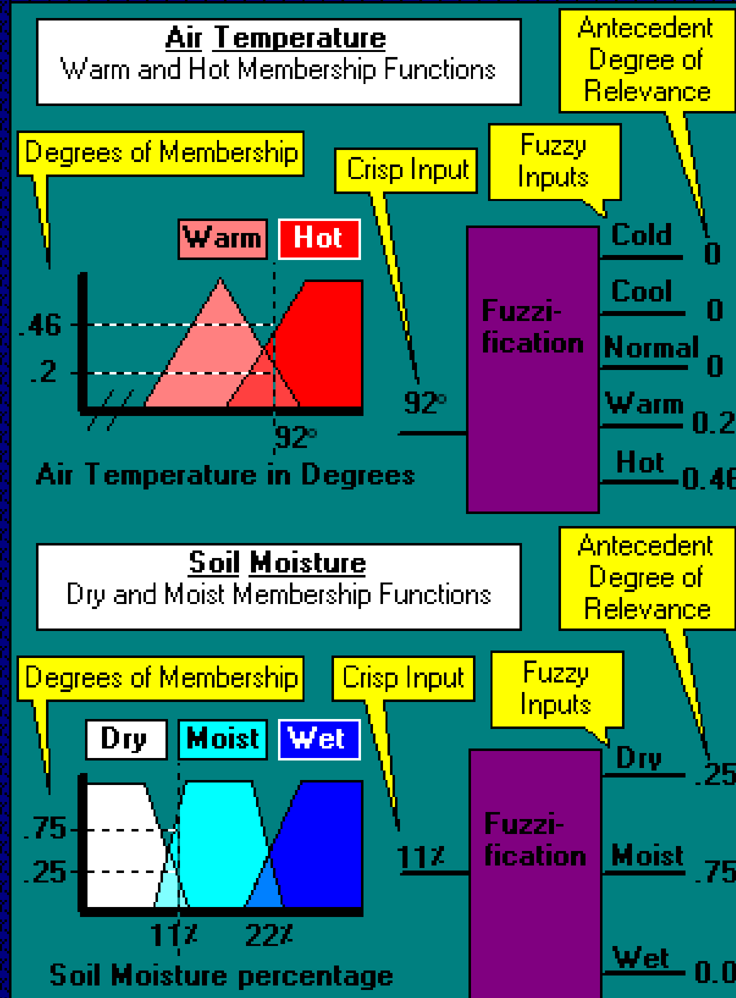

Fuzzy Logic Education Program
2 of 8
For example, is 80 degrees Fahrenheit warm or hot? In fuzzy logic, and in the real world, "some of both" might be the answer. As you can see on the fuzzy graph (lower right), 80 degrees is partially warm and partially hot in fuzzy set representation.
Conventional Sets
Warm
Hot
60° 70° 80° 90°
Temp. (F)
Fuzzy Sets
Warm
Hot
60° 70° 80° 90°
Temp. (F)
Introduction
Fuzzy Fundamentals Menu

Fuzzy Logic Education Program

The next step in the rule evaluation process is to evaluate the relevance or degree of membership of each rule's antecedent.

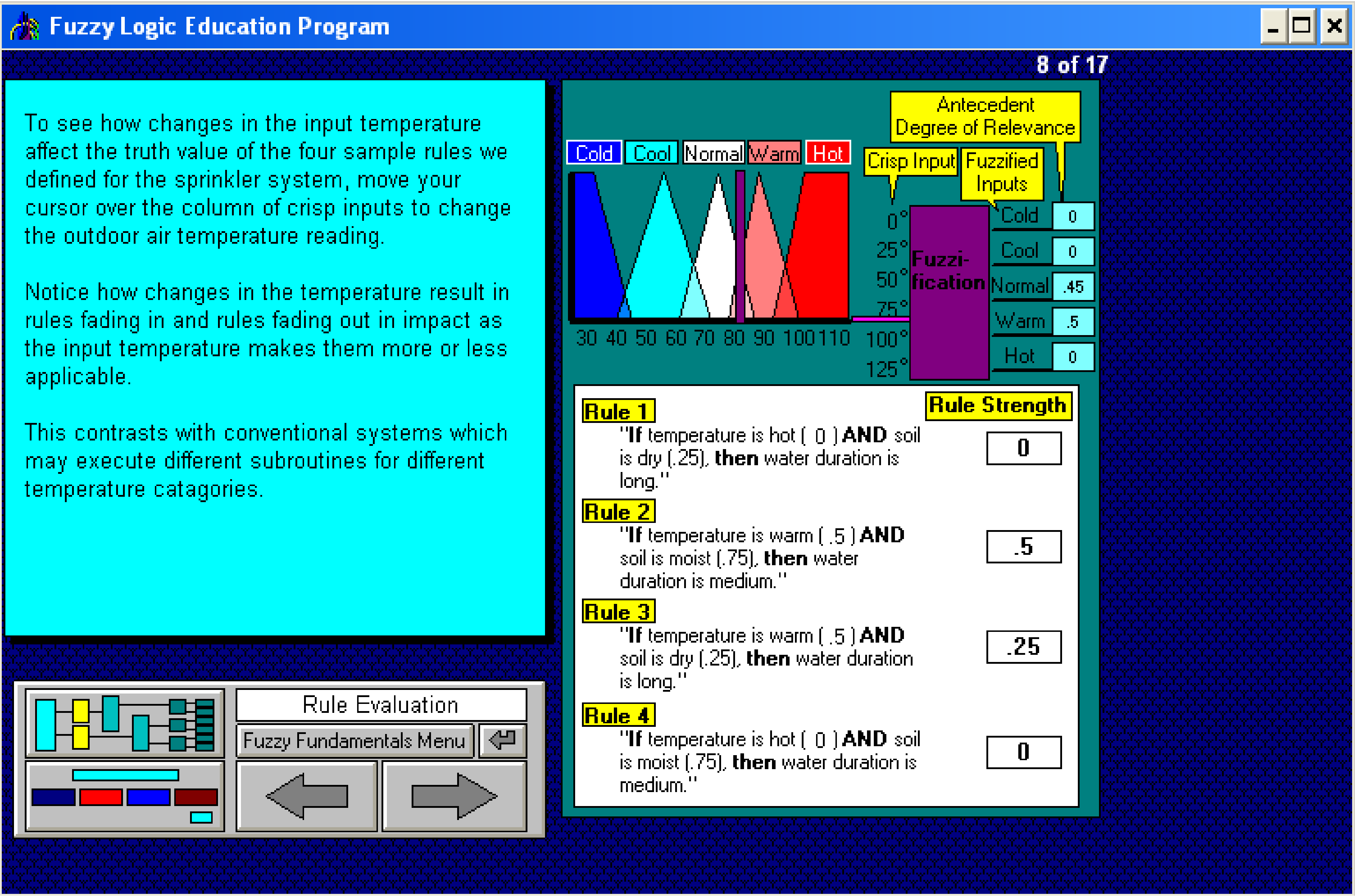
To find the relevance of each antecedent, extend a vertical reference line through the crisp input (x-value) and find the y-value where it intersects the membership functions.

In the sprinkler system, the input of 92° F air temperature would be found at the intersection of 0.2 of fuzzy set "warm" and 0.46 of fuzzy set "hot". The input of 11% soil moisture would be found at the input of intersection of 0.25 of fuzzy set "dry" and 0.75 of fuzzy set "moist".



Rule Evaluation

Fuzzy Fundamentals Menu


Fuzzy Logic Education Program
8 of 17
To see how changes in the input temperature affect the truth value of the four sample rules we defined for the sprinkler system, move your cursor over the column of crisp inputs to change the outdoor air temperature reading.
Notice how changes in the temperature result in rules fading in and rules fading out in impact as the input temperature makes them more or less applicable.
This contrasts with conventional systems which may execute different subroutines for different temperature catagories.
Cold
Cool
Normal
Warm
Hot
30 40 50 60 70 80 90 100 110
Antecedent Degree of Relevance
Crisp Input
Fuzzified Inputs
0°
25°
50°
75°
100°
125°
Fuzzi-fication
Cold 0
Cool 0
Normal .45
Warm .5
Hot 0
Rule Strength
Rule 1
"If temperature is hot (0) AND soil is dry (.25), then water duration is long."
0
Rule 2
"If temperature is warm (.5) AND soil is moist (.75), then water duration is medium."
.5
Rule 3
"If temperature is warm (.5) AND soil is dry (.25), then water duration is long."
.25
Rule 4
"If temperature is hot (0) AND soil is moist (.75), then water duration is medium."
0
Rule Evaluation
Fuzzy Fundamentals Menu

3) Game Theory

Game Theory

- **Let's play a game. You have 100$ and a button.**
- **When you click the button, every other player loses 2$.**
- **If one or more players push the button and you have pushed your button, you lose 1$ instead of 2$.**
- **This game is:**
  - **Interactive (actions of one player affect the others).**
  - **Symmetric (same rule apply to you apply to others).**
  - **Common knowledge (all the players know all the rules and know that others know all the rules).**
- **Are you going to push the button, or not?**
- **You must have a strategy to decide what actions are more appropriate with your goals.**
- **Between 30% to 70% choose to push the button.**
- **Will you decision be different if you were playing for 1 million or manipulating nuclear missiles?**

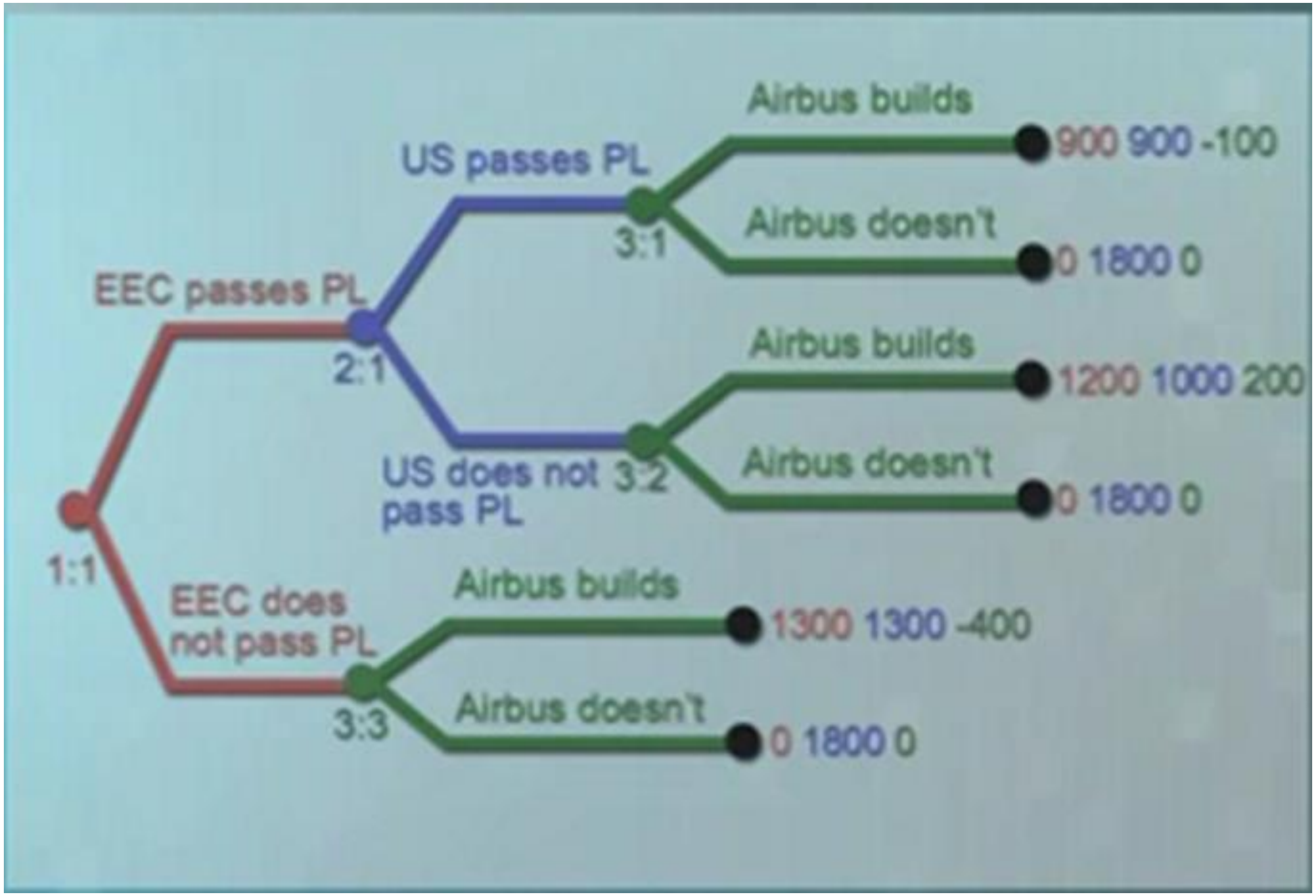
What is Game Theory

- **Game theory == Strategic Interactive Decision Making**
- **Game theory can be applied to board games and card games, but its main goal is to provide a person with a better strategy when facing a conflict in business or life.**
- **Game theory handles simple problem, but it help develop intuition on how to solve more complex, real-life problems.**
- **Game theory applications in economics led to a series of Nobel price winners**

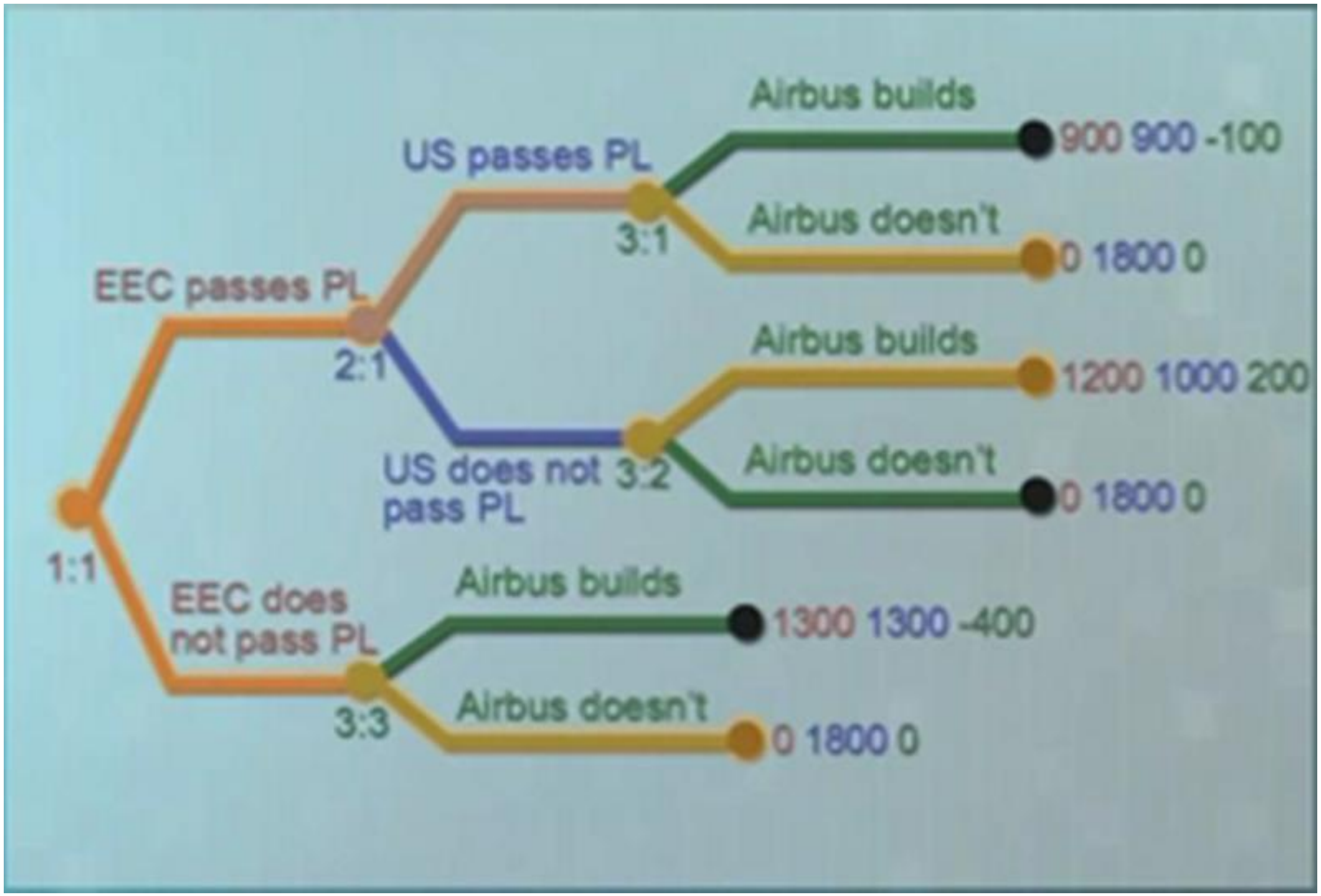
Strategic Game Example

- **Example of a simultaneous game: Airbus competing in medium airplanes market with Boeing.**
  - **Start up cost: 1 Billion**
  - **Europe and USA are two separate market, any monopoly in either of these market generates 900M from each market.**
  - **A competition reduces the profit to 300M for each company, but makes the country gain 700M (lower cost, better quality).**
  - **A country can generate a Protective Legislation that does not allow manufactures from another region to enter its market.**
  - **A country gains the profit of its local airplane manufacture and the market gain, if any.**

Game Theory – Decision Trees



Game Theory – Decision Trees



4) Genetic Algorithm

Genetic Algorithm

- All living organisms consist of cells. In each cell there is the same set of **chromosomes.**
- During reproduction, first occurs **recombination** (or **crossover**). Genes from parents form in some way the whole new chromosome.
- The new created offspring can then be **mutated**. Mutation means, that the elements of DNA are a bit changed. This changes are mainly caused by errors in copying genes from parents.
- The fitness of an organism is measured by success of the organism in its life.
- **Genetic algorithms** uses this model to find sub-optimal solutions.

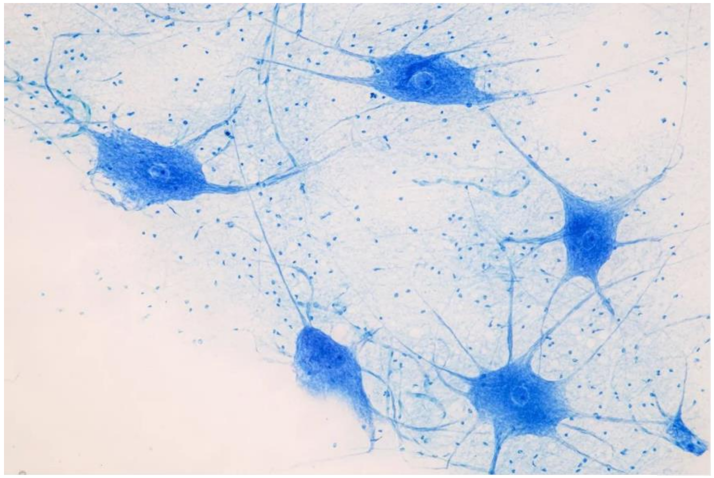
Genetic Algorithm

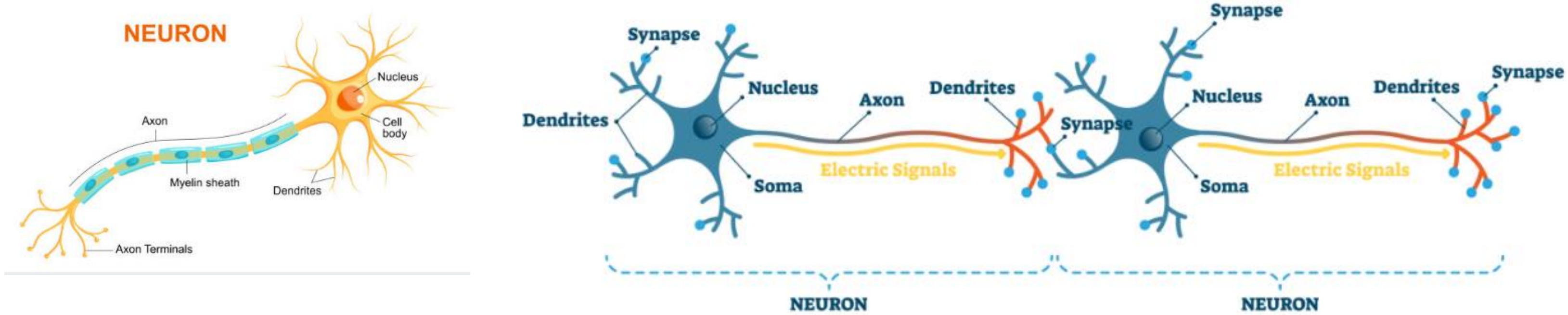
1. **[Start]** Generate random population of *n chromosomes (suitable solutions for the problem)*
2. **[Fitness]** Evaluate the fitness *f(x) of each chromosome x in the population*
3. **[New population]** Create a new population by repeating following steps until the new population is complete
    1. **[Selection]** Select parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)
    2. **[Crossover]** With a crossover probability cross over the parents to form a new offspring (children).
    3. **[Mutation]** With a mutation probability mutate new offspring at each locus (position in chromosome).
    4. **[Accepting]** Place new offspring in a new population
4. **[Replace]** Use new generated population for a further run of algorithm
5. **[Test]** If the end condition is satisfied, stop, and return the best solution in current population
6. **[Loop]** Go to step 2

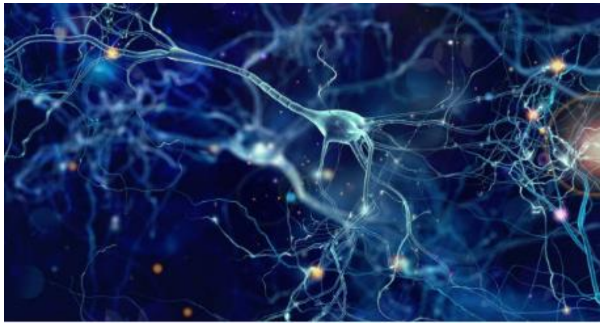
5) Neural Networks

Neural Networks

- ***How does the brain work***
- ***How does the brain learn***







Neural Networks

- ***Frank Rosenblatt*** proposed in 1958 the use of single layer neural network (***Perceptron***) as a linear classifier.