CMSC 373 Artificial Intelligence
Fall 2023
07-Logic

Deepak Kumar
Bryn Mawr College

1

Representing Knowledge using Logic

- Aristotle: What are correct arguments/thought processes?
- Formal Logics:

  Socrates is human.
  All humans are mortal.
  Therefore, Socrates is mortal.

- Laws of thought govern the operation of the mind.
- Logic uses **declarative symbolic representations**.

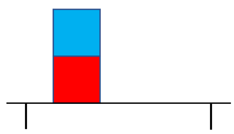
2

2

Example

- BLOCKSWORLD with declarative knowledge

if a block is on top of another block
then
the latter block is not clear



Facts

blue block is on top of red block
red block is on the table
blue block is clear

∴ red block is not clear

3

3

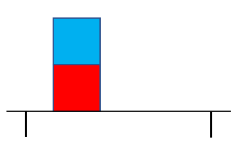
Example

Represents the "what" and not "how".

- BLOCKSWORLD with declarative knowledge

if a block is on top of another block
then
the latter block is not clear

These are called **propositions**.



Facts

blue block is on top of red block
red block is on the table
blue block is clear

∴ red block is not clear

4

4

Propositions

- Propositions are something that can be expressed by a declarative sentence of English (or another language)

  B is a block.
  The sky is blue.
  The snark was a boojum.
  Dinosaurs were warm-blooded.
  The stock market index will double its current value within two years.

- These sentences are true if the proposition they express holds. It is false if the proposition does not hold.

- If a sentence is assumed to be **true**, one can answer questions about it without knowing what the words in it mean.

5

5

Propositions

- Propositions are something that can be expressed by a declarative sentence of English (or another language)

  B is a block.
  The sky is blue.
  The snark was a boojum.
  Dinosaurs were warm-blooded.
  The stock market index will double its current value within two years.

- These sentences are true if the proposition they express holds. It is false if the proposition does not hold.

- If a sentence is assumed to be **true**, one can answer questions about it without knowing what the words in it mean:

  What kind of a thing is a snark?
  A boojum.

  Is it true that a snark was either a beejum or a boojum?
  Yes, because a snark is a boojum.

  If no boojum is ever a beejum, was the snark ever a beejum?
  No.

  What is an example of something that was a boojum?
  The snark.

6

6

Entailment

- Entailment enables us to answer questions from sentences without knowing what the words in a sentence stand for.

  Deepak is in the classroom or in his office.
  Nobody is in the classroom.
  So: Deepak is in the office.

  Evan is married to Chris or Pat.
  Nobody is married to Chris.
  So: Evan is married to Pat.

  The frumble is frimble or framble.
  Nothing is frimble.
  So: The frumble is a framble.
- Without knowing what the words mean, we can arrive at a correct conclusion.

7

7

Logical Entailment

- A collection of sentences S_1, S_2, …, S_n **logically entails** another sentence S if the truth of S is implicit in the truth of the S_i sentences.

- That is, no matter what certain words (like *boojum*, *framble*, etc.) in sentences S_i mean, if S_i are all true, then the sentence S is also true.

  The snark was a boojum.

  Logically entails

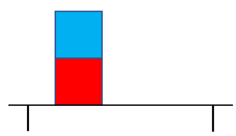
  Something was a boojum.

8

8

Knowledge Representation in Logic

- Syntax: How sentences are formed/written
- Semantics: Meaning/Interpretation of sentences
- Computation: How sentences are manipulated



Facts

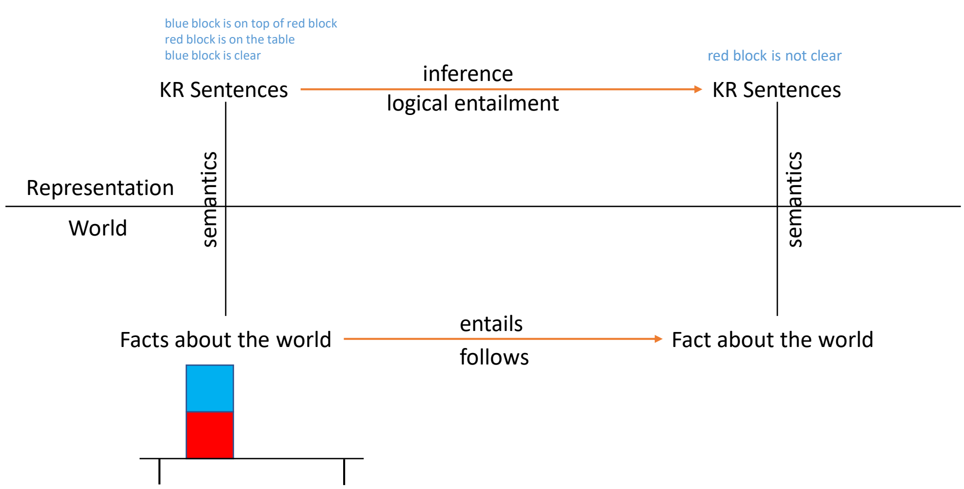
if a block is on top of another block
then the latter block is not clear
blue block is on top of red block
red block is on the table
blue block is clear

∴ red block is not clear

9

9

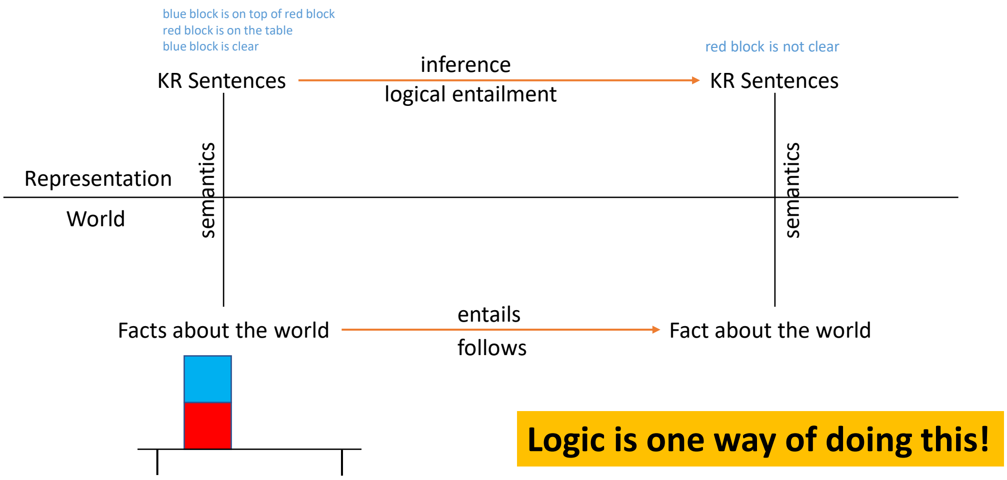
Knowledge Representation



10

10

Knowledge Representation



11

11

What is a logic?

- Study of correct inferences

Premises

…
… } True
…

Conclusion

∴ … } is also True

12

12

What is a logic?

- Study of correct inferences

Premises

…
… } True
…

Conclusion

∴ … } is also True

If it is cold then my car will not start.
My car will not start.

∴ It is cold.

If it is cold then my car will not start.
It is cold.

∴ My car will not start.

13

13

What is a logic?

- Study of correct inferences – Truth preserving consequences

Premises

…
… } True
…

Conclusion

∴ … } is also True

If it is cold then my car will not start.
My car will not start.

∴ It is cold.

If it is cold then my car will not start.
It is cold.

∴ My car will not start.

14

14

What is a logic?

- Study of correct inferences – Truth preserving consequences

Premises

…
… } True
…

Conclusion

∴ … } is also True

If it is cold then my car will not start.
My car will not start.

∴ It is cold.

← This is NOT Truth preserving.

If it is cold then my car will not start.
It is cold.

∴ My car will not start.

← This is Truth preserving.

15

15

What is a logic?

- Study of correct inferences – Truth preserving consequences

Premises

…
… } True
…

Conclusion

∴ … } is also True

If it is cold then my car will not start.
My car will not start.

∴ It is cold.

← This is NOT Truth preserving. This is sometimes used in diagnosis! (aka abduction)

If it is cold then my car will not start.
It is cold.

∴ My car will not start.

← This is Truth preserving. (aka **deduction**)

16

16

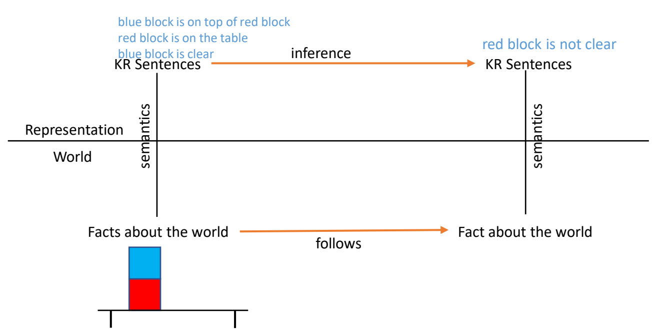
What is a logic?

- Study of correct inferences
- Formalize the notion of correct inference
  - Step1: Define a formal language to write sentences – syntax
    well-formed sentences (wffs)
  - Step 2: What do the wffs mean? – Semantics/Model Theory
    Need an interpretation for wffs
  - Step 3: Rules of Inference – Proof Theory

17

17

Rules of Inference



Properties

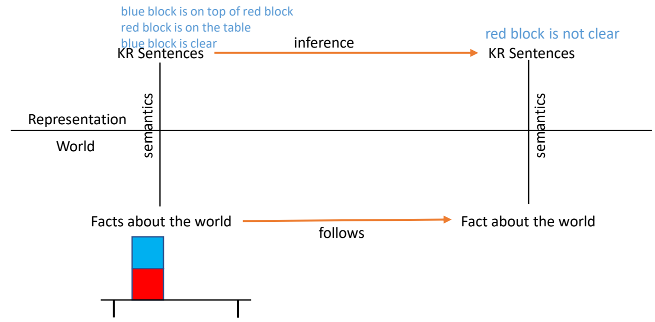
Soundness: Every new sentence that can be derived from KR is a valid consequence.

Completeness: Every valid consequence of KR can be formally derived.

18

18

Rules of Inference



Properties

Soundness: Every new sentence that can be derived from KR is a valid consequence. ← This is easy to show

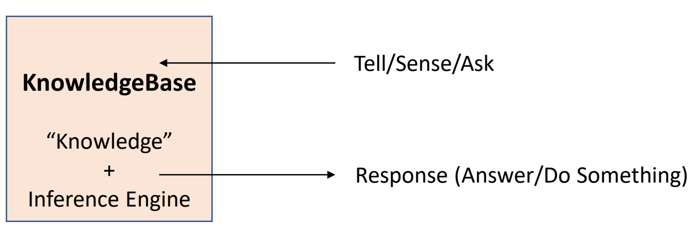
Completeness: Every valid consequence of KR can be formally derived. ← This is true for some logics.

19

19

Knowledge Representation & Reasoning

- Knowledge Representation & Reasoning (KRR) Systems



- Knowledge – set of sentences that describe facts about the world (or domain)
- Inferences – procedures/rules that operate on facts to infer new facts

20

20

There are many types of logic

- Propositional Logic
- First-Order Logic
- Second-Order Logic
- Temporal Logic
- Modal Logic
- Constraint Logic
- Etc.

21

21

There are many types of logic

- Propositional Logic
- First-Order Logic (semi-formally!)
- Second-Oder Logic
- Temporal Logic
- Modal Logic
- Constraint Logic
- Etc.

22

22

First-Order Predicate Calculus (FOPC)

Components

- Object constants: A, B, Deepak, etc.
- Function Constants: *fatherOf*, *colorOf*, etc.
- Relation Constants: Parent, On, Clear, Sibling, etc.
- Variables: x, y, z, u, v, w, etc.
- Connectives: ∨, ∧, ¬, ⇒
- Quantifiers: ∀, ∃
- Delimiters: (,), [,]

23

23

Objects, Functions, Relations

Object constants: these are objects or individuals in the domain…

A, B, Deepak, Red, Car54, etc.

Relation Constants: Denote properties of/between objects

Parent^2, On^2, Clear^1, Sibling^2, etc.

24

24

FOPC - Syntax

Terms

- An object is a term

Wffs – represent propositions

- Atomic Sentences (wffs) relation-constant^n(term$_1$, term$_2$,…term$_n$)

  e.g. Parent(Ginny, Lily)

- Propositional wffs
  If ω_1 and ω_2 are wffs, then so are
  - ω_1
  - $\omega_1 \vee \omega_2$ disjunction
  - $\omega_1 \wedge \omega_2$ conjunction
  - $\omega_1 \Rightarrow \omega_2$ implication
  - $\neg\, \omega_1$ negation

25

25

FOPC - Syntax

Terms

- An object is a term

Wffs – represent propositions

- Atomic Sentences (wffs) relation-constant(…)

  e.g. Parent(Ginny, Lily)

- Propositional wffs
  If ω_1 and ω_2 are wffs, then so are
  - ω_1
  - $\omega_1 \vee \omega_2$ disjunction
  - $\omega_1 \wedge \omega_2$ conjunction
  - $\omega_1 \Rightarrow \omega_2$ implication
  - $\neg\, \omega_1$ negation

Save for later…

Variables: x, y, z, u, v, w, etc.
Quantifiers: ∀

26

26

FOPC - Semantics

Objects The world can have infinite objects/individuals

- concrete
- abstract
- fictional

As long as it is a name and we need to say something about it.

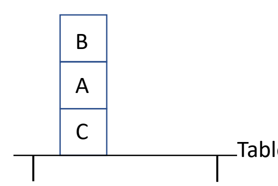
Relations Denote properties

Heavy(x) On(A, B)
Big(y) Clear(A)
Human(Deepak) Block(A)

Interpretation

- An atomic wff is True/False just in case the relation denoted by it holds for its arguments.

FOPC Example - Blocksworld



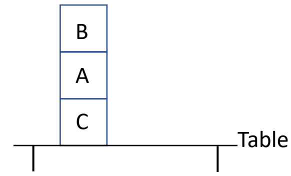
- **Objects** A, B, C, Table
- **Relations** On^2, $Clear^1$

On(C, Table) On(A, C) On(B, A)
Clear(B) Clear(A) Clear(C) Clear(Table)

- Example Knowledge Base

$$\Delta = \{On(C, Table), On(A, C), On(B, A), Clear(B)\}$$

FOPC Example - Blocksworld



- **<u>Objects</u>** A, B, C, Table
- **<u>Relations</u>** On^2, $Clear^1$

On(C, Table) On(A, C) On(B, A)
Clear(B) Clear(A) Clear(C) Clear(Table)

- Example Knowledge Base

But, how to make general Statements about the world?

Δ = {On(C, Table), On(A, C), On(B, A), Clear(B)}

29

29

Variables & Quantifiers

- **<u>Variable</u>**s – A variable is a term. i.e. it denotes/can denote an object.
- **<u>Universal Quantifier (∀ - "for-all")</u>**

if ω is a wff and x is a variable then

$(\forall x)\ \omega$ is a wff

$\forall x\ (\omega)$ is a wff

$\forall x\ [\omega]$ is a wff

ω is the scope of the variable.
E.g.

$\forall x\ [\ P(x) \Rightarrow R(x)\ ]$

Read as: All P's are R's.
Or: If something is a P, then it is an R

30

30

Variables & Quantifiers

- **<u>Variable</u>**s – A variable is a term. i.e. it denotes/can denote an object.
- **<u>Universal Quantifier (∀ - "for-all")</u>**

if ω is a wff and x is a variable then

$(\forall x)\ \omega$ is a wff

$\forall x\ (\omega)$ is a wff

$\forall x\ [\omega]$ is a wff

ω is the scope of the variable.

E.g.

$\forall x\ [\ P(x) \Rightarrow R(x)\ ]$

These are also called conditional sentences.

Read as: All P's are R's.
Or: If something is a P, then it is an R

31

31

Semantics of ∀

In a given domain

$\forall x\ \omega(x)$ has value True

Just in case ω(x) has value True <u>for all</u> assignments of x to objects in the domain.

e.g. On^2, $Clear^1$

$\forall x\ [\ On(x, C) \Rightarrow \neg Clear(C)\ ]$ *If anything is on C then it is not clear.*

Just in case x=A/B/C/Table are True.

32

32

Example

Tell King(John)
Tell Person(Richard)
Tell $\forall x$ [King(x) ⇒ Person(x)]

Queries

1. King(John)?

   True

2. Who is a person?

   Person(x)?

   x = Richard

3. What about John??? Requires inference!

Is Person(x) a King(x)?
Since King(John)
Person(John)

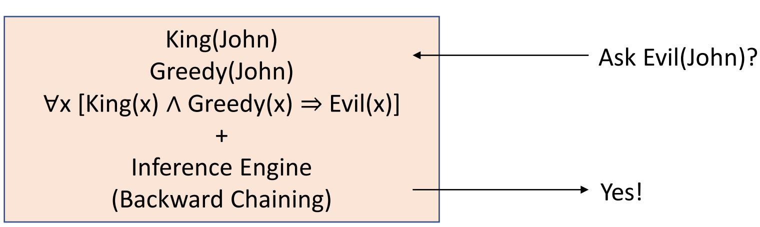
x = John

33

33

KRR Systems

- Tell-Ask Systems



34

34

Example

Tell King(John)
Tell Person(Richard)
Tell $\forall x$ [King(x) ⇒ Person(x)]

Queries

1. King(John)?

   True

2. Person(x)?

   x = Richard
   x = John

Tell ∀x [King(x) ∧ Greedy(x) ⇒ Evil(x)]
Tell King(John)
Tell Greedy(John)

Query

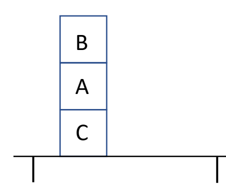
Evil(John)?
True

35

35

Blocksworld

On(x, y) : x is on top of y
Clear(x) : x is clear
Block(x) : x is a block

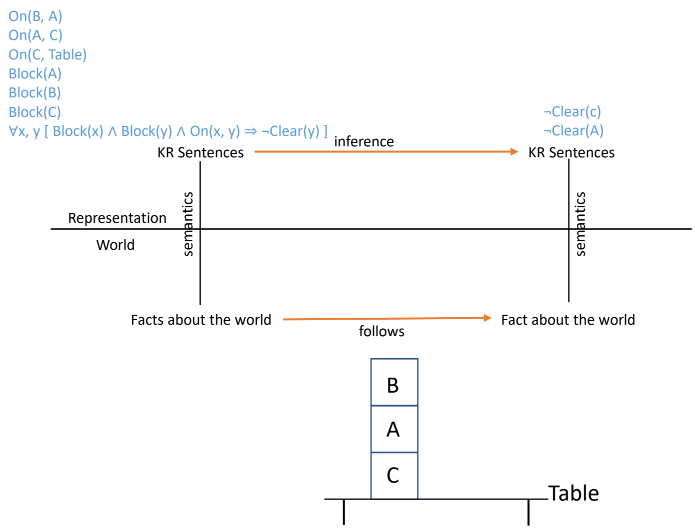


On(B, A)
On(A, C)
On(C, Table)
Block(A)
Block(B)
Block(C)
∀x, y [Block(x) ∧ Block(y) ∧ On(x, y) ⇒ ¬Clear(y)]

36

36

Reasoning with Logic



37

37

Blocksworld - Alternatives

On(x, y) : x is on top of y

Clear(x) : x is clear

Block(x) : x is a block

Isa(x, y) : x is a y

Isa(A, Block)
Isa(B, Block)
Isa(C, Block)

On(B, A)
On(A, C)
On(C, Table)
Block(A)
Block(B)
Block(C)
∀x, y [Block(x) ∧ Block(y) ∧ On(x, y) ⇒ ¬Clear(y)]

38

38

Blocksworld - Alternatives

On(x, y) : x is on top of y

Clear(x) : x is clear

Block(x) : x is a block

Isa(x, y) : x is a y

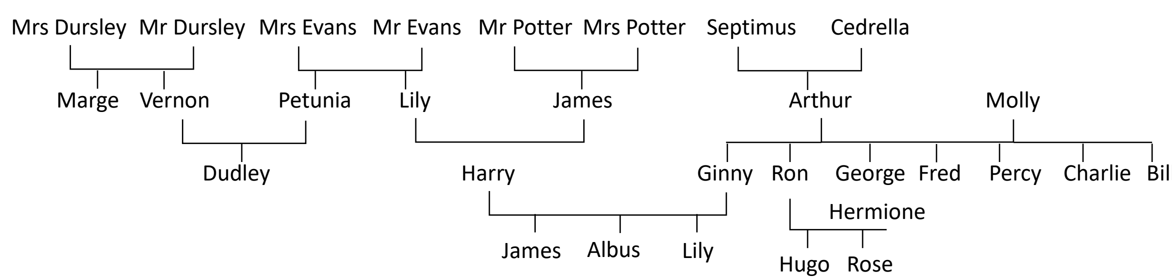
Isa(A, Block)
Isa(B, Block)
Isa(C, Block)

On(B, A)
On(A, C)
On(C, Table)
Block(A)
Block(B)
Block(C)

$\forall x, y$ [Isa(x, Block) ∧ Isa(y, Block) ∧ On(x, y) ⇒ ¬Clear(y)]

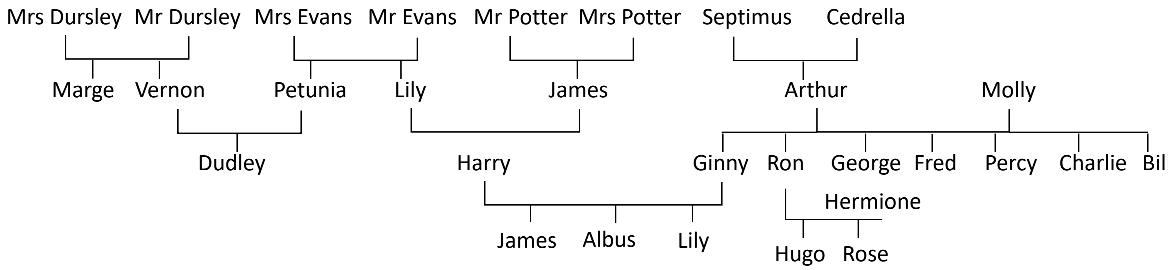
39

39



40

40

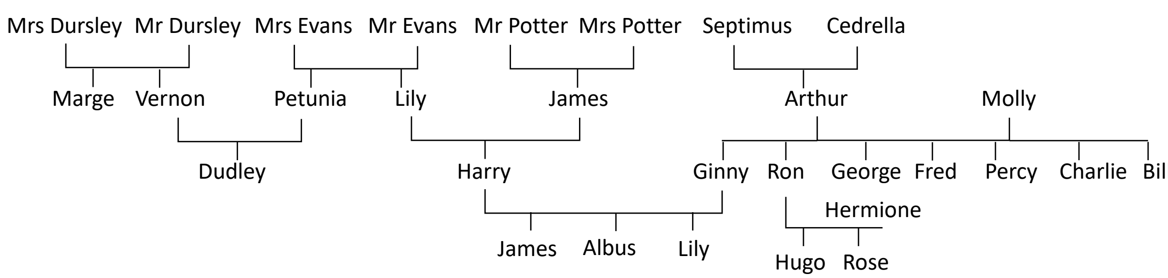


Predicates

1. Female(x) : x is female
2. Male(x) : x is male
3. Parent(x, y) : x is a parent of y
4. Married(x, y) : x is married to y

41

41



Predicates

1. Female(x) : x is female
2. Male(x) : x is male
3. Parent(x, y) : x is a parent of y
4. Married(x, y) : x is married to y

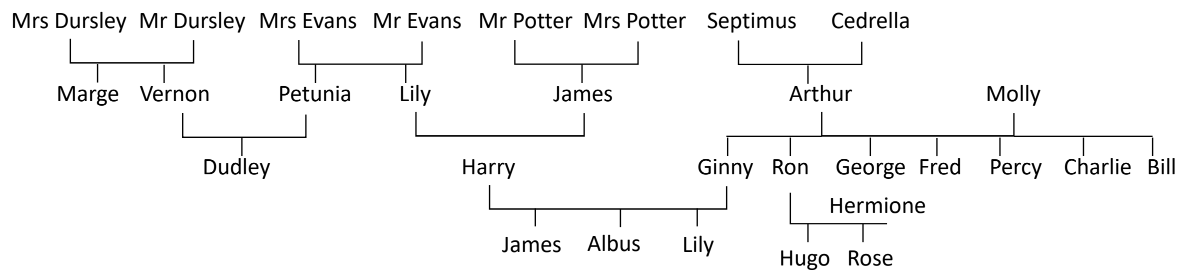
Female(Marge)	**Married(Vernon, Petunia)**	**Parent(Harry, James2)**
Female(Petunia)	**Married(Lily, James1)**	**Parent(Ginny, James2)**
Female(Lily)	**Married(Arthur, Molly)**	**Parent(Petunia, Dudley)**
Male(James1)	**Married(Harry, Ginny)**	**Parent(Vernon, Dudley)**
Male(Vernon)	**Married(Ron, Hermione)**	**Parent(Molly, Bill)**
Etc.	**Etc.**	**Etc.**

Symbols

Marge, Vernon, James1, James2,
Albus, Lily, Harry, etc.

42

42



Predicates

1. Female(x) : x is female
2. Male(x) : x is male
3. Parent(x, y) : x is a parent of y
4. Married(x, y) : x is married to y

Symbols

Marge, Vernon, James1, James2, Albus, Lily, Harry, etc.

Female(Marge)	**Married(Vernon, Petunia)**	**Parent(Harry, James2)**
Female(Petunia)	**Married(Lily, James1)**	**Parent(Ginny, James2)**
Female(Lily)	**Married(Arthur, Molly)**	**Parent(Petunia, Dudley)**
Male(James1)	**Married(Harry, Ginny)**	**Parent(Vernon, Dudley)**
Male(Vernon)	**Married(Ron, Hermione)**	**Parent(Molly, Bill)**
Etc.	**Etc.**	**Etc.**

Facts

43

43

Modeling Relationships

- Spouse(x, y) : x is a spouse of y
- Wife(x, y) : x is a wife of y
- Husband(x, y)
- Father(x, y), Mother(x, y)
- Sibling(x, y), Brother(x, y), Sister(x, y)
- GrandParent/GrandMother/GrandFather
- GrandChild/GrandDaughter/GrandSon
- Aunt/Uncle/AuntOrUncle
- Cousin
- Niece/Nephew/NieceOrNephew
- Child/Daughter/Son
- Etc.

44

44

Modeling Relationships

- Spouse(x, y) : x is a spouse of y
- Husband/Wife
- Mother/Father

Modeling Relationships

- Spouse(x, y) : x is a spouse of y

  ∀x, y [Married(x, y) ⇒ Spouse(x, y)]

- Husband/Wife*

  ∀x, y [Female(x) ∧ Married(x, y) ⇒ Wife(x, y)]

- Father/Mother *

  ∀x, y [Female(x) ∧ Parent(x, y) ⇒ Mother(x, y)]

Modeling Relationships

- Spouse(x, y) : x is a spouse of y
- Wife(x, y) : x is a wife of y
- Husband(x, y)
- Father(x, y), Mother(x, y)
- Sibling(x, y), Brother(x, y), Sister(x, y)
- GrandParent/GrandMother/GrandFather
- GrandChild/GrandDaughter/GrandSon
- Aunt/Uncle/AuntOrUncle
- Cousin
- Niece/Nephew/NieceOrNephew
- Child/Daughter/Son
- Etc.

47

47

References

- M. Wooldridge, *A Brief History of Artificial Intelligence*. Flatiron Books, 2020.
- Michael Genesereth & Nils Nilsson, *Logical Foundations of Artificial Intelligence*, Morgan Kauffman, 1987.
- Hector Levesque, *Thinking as Computation*, MIT Press, 2017.

48

48