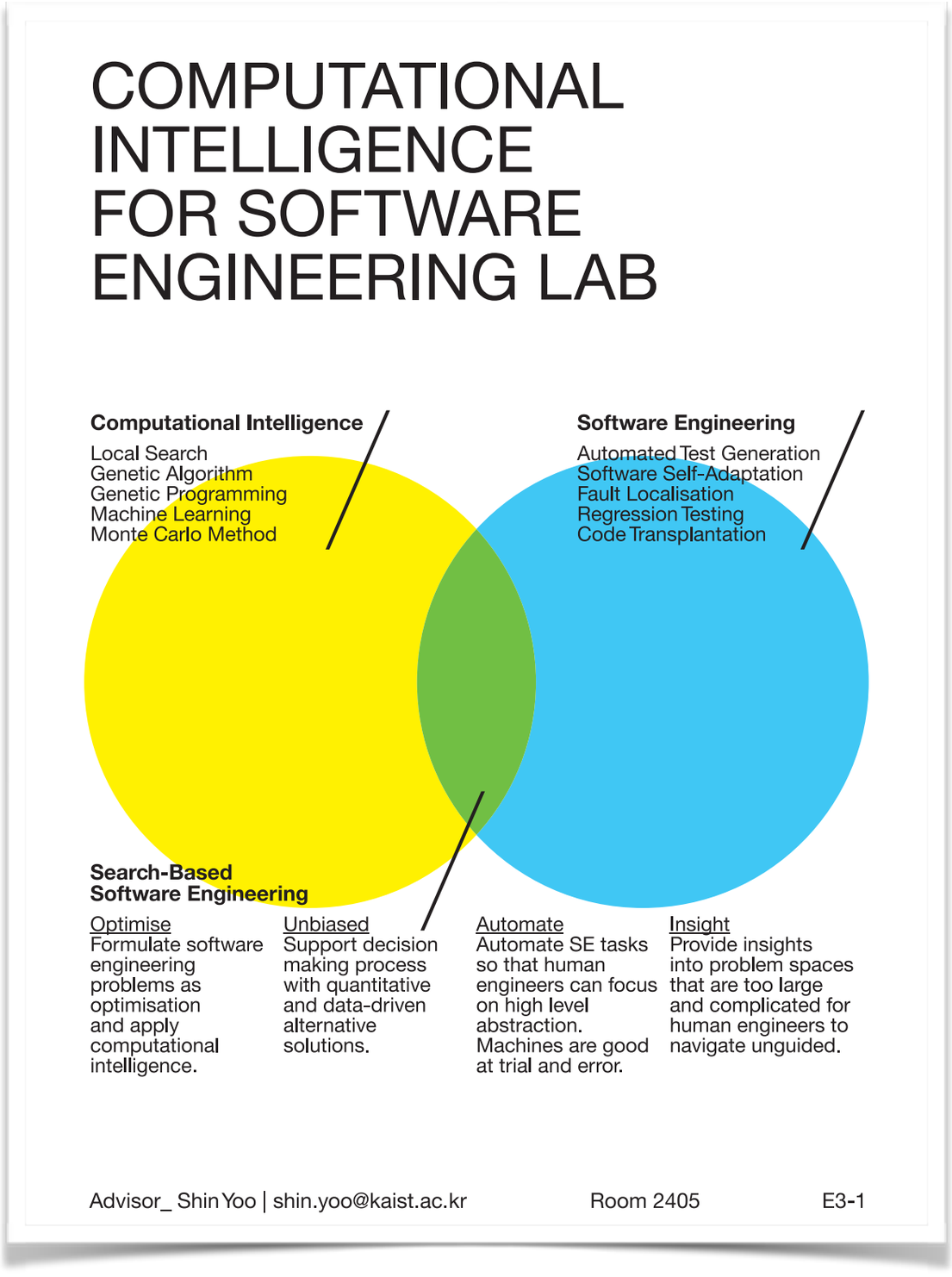
CS402: Introduction to Logic for Computer Science

Shin Yoo

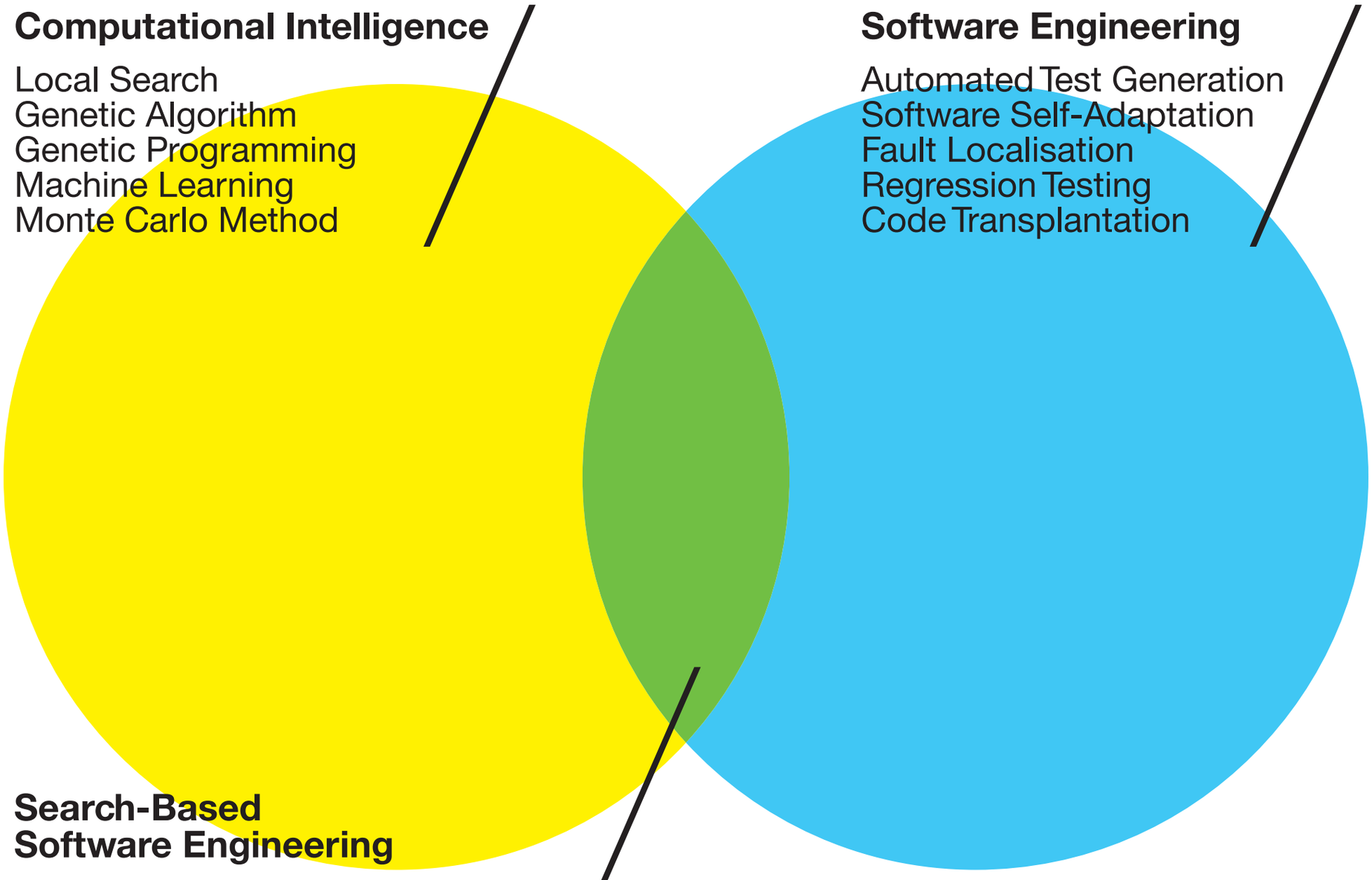
Spring 2016

Me

- Shin Yoo, joined KAIST CS in August 2015
  - PhD at King's College London, UK
  - Assistant Professor at University College London, UK
- COINSE (Computational Intelligence for Software Engineering) Lab
- Research interest: SBSE, regression testing, automated debugging, evolutionary computation, information theory, program analysis…
- Office hours: 10-12 Tuesdays, 2405, E3-1
- shin.yoo@kaist.ac.kr



Disclaimer: this course has **nothing** to do with COINSE Lab :p



Search-Based Software Engineering

Optimise
Formulate software engineering problems as optimisation and apply computational intelligence.

Unbiased
Support decision making process with quantitative and data-driven alternative solutions.

Automate
Automate SE tasks so that human engineers can focus on high level abstraction. Machines are good at trial and error.

Insight
Provide insights into problem spaces that are too large and complicated for human engineers to navigate unguided.

Introduction to SBSE (1/2)

Shin Yoo
CS492D, Fall 2015, School of Computing, KAIST

There will, hopefully, be a CS492 for Search Based Software Engineering in Autumn 2016.

Syllabus

- This course is about basics of logic used in computer programming. Topics covered in this course are: propositional calculus, predicate calculus, axiomatic theories, skolemization, unification, and resolution.

- We will also try to incorporate practical application of logic systems using various tools.

Learning Outcome

- Essentially, we learn
  - Logic systems, different flavours and their differences in expressiveness
  - How to reason formally and algorithmically
- But we also train ourselves to
  - Think and reason formally and symbolically

Course Webpage

- Lecture Materials and Links: http://coinse.kaist.ac.kr/teaching/2016-cs402/

- Coursework Online Submission: KLMS

Pre-requisite

- ~~Knowledge of greek alphabets~~
- ~~Skill to use proper typesetting systems~~
- CS204 Discrete Math
- Ability to build and install *nix software

Because this lies ahead…

Ax. 1. $\{P(\varphi) \wedge \Box\, \forall x[\varphi(x) \rightarrow \psi(x)]\} \rightarrow P(\psi)$
Ax. 2. $P(\neg\varphi) \leftrightarrow \neg P(\varphi)$
Th. 1. $P(\varphi) \rightarrow \Diamond\, \exists x[\varphi(x)]$
Df. 1. $G(x) \iff \forall\varphi[P(\varphi) \rightarrow \varphi(x)]$
Ax. 3. $P(G)$
Th. 2. $\Diamond\, \exists x\, G(x)$
Df. 2. $\varphi \text{ ess } x \iff \varphi(x) \wedge \forall\psi\, \{\psi(x) \rightarrow \Box\, \forall y[\varphi(y) \rightarrow \psi(y)]\}$
Ax. 4. $P(\varphi) \rightarrow \Box\, P(\varphi)$
Th. 3. $G(x) \rightarrow G \text{ ess } x$
Df. 3. $E(x) \iff \forall\varphi[\varphi \text{ ess } x \rightarrow \Box\, \exists y\, \varphi(y)]$
Ax. 5. $P(E)$
Th. 4. $\Box\, \exists x\, G(x)$

Ax. 1. $\{P(\varphi) \wedge \Box\, \forall x[\varphi(x) \rightarrow \psi(x)]\} \rightarrow P(\psi)$
Ax. 2. $P(\neg\varphi) \leftrightarrow \neg P(\varphi)$
Th. 1. $P(\varphi) \rightarrow \Diamond\, \exists x[\varphi(x)]$
Df. 1. $G(x) \iff \forall\varphi[P(\varphi) \rightarrow \varphi(x)]$
Ax. 3. $P(G)$
Th. 2. $\Diamond\, \exists x\, G(x)$
Df. 2. $\varphi \text{ ess } x \iff \varphi(x) \wedge \forall\psi\,\{\psi(x) \rightarrow \Box\, \forall y[\varphi(y) \rightarrow \psi(y)]\}$
Ax. 4. $P(\varphi) \rightarrow \Box\, P(\varphi)$
Th. 3. $G(x) \rightarrow G \text{ ess } x$
Df. 3. $E(x) \iff \forall\varphi[\varphi \text{ ess } x \rightarrow \Box\, \exists y\, \varphi(y)]$
Ax. 5. $P(E)$
Th. 4. $\Box\, \exists x\, G(x)$

Kurt Gödel's ontological proof
of God's existence.

We will examine this in due course.

But seriously,

- Consider learning how to write in LaTeX: it makes writing logical symbols so much easier.
- Critical skill if you’re going into grad-school ;)
- Think HTML-like mark-up language for typesetting.
- There are online editors such as sharelatex.com

Evaluation

- Mid Term Exam: 30%
- Final Term Exam: 30%
- Various coursework, quiz, etc: 30%
- Class participation: 10%

Questions?

Since we've all done CS204…

First coursework

- Warming-up!
- Due 10:00AM 8th March
- Submit through **KLMS**
- **No. Late. Submission.**

CS402
Introduction to Formal Logic

Coursework 1: Warming Up
Due on 10:00AM, 8 March 2016

1 The Basic

Sherlock Holmes famously stated "Once you eliminate the impossible, whatever remains, no matter how improbable, must be the truth". Suppose Holmes and Watson have five suspects, $\{A, B, C, D, E\}$, and evidence eliminated all but C. Describe the application of the above rule of elimination to this case, in the form of formal propositional logic.

2 The Joke

Fill in the blank in the following joke.

> Three logicians walked into a bar.
> The bartender asked: 'do you all want beer?'
> The first logician said: 'I don't know.'
> The second logician said: 'I don't know.'
> The third logician said: '_____.'
> The bartender then served three pints of beer.

3 The Valid

Represent each of the following in propositional logic, and decide whether it is valid or not.

- "You must be the criminal. I'm sure of this, because you walk with a slight limp. We all know that the criminal walks with a slight limp."
- "We will be able to have class, only if either Shin brought his laptop, or there is a computer in the lecture room. There is no computer in the room. So we will not be able to have class. I say this, of course, because Shin forgot to bring his laptop."

4 The Proof

Prove the following.

- $P \leftrightarrow \neg Q \vdash \neg P \leftrightarrow Q$