Module A Evaluation

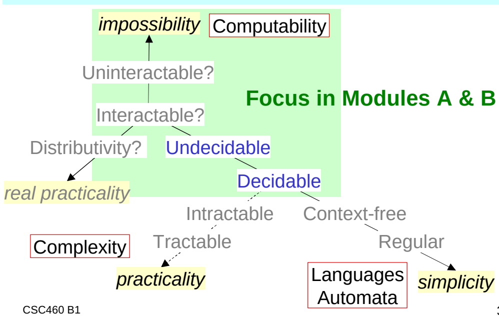
Review your portfolio

- My comments are sporadic and scattered on Review Ex, Comprehensive Ex, and Supp. Notes.
- You are encouraged to clarify and discuss my comments.
- You can keep the folder till the next class; then, return it to me.

Unit B1: Overview

- Understand Module B learning goals
- Review Module A
  - Comprehensive/Review Exercises
  - Practice
- Understand a systematic way of simulating a TM using a TM
- Preview Exercise B1 "Universal TM"

Overview: Theory of Computation



Module B Overview

B1 Computability: Universal TM
B2 Diagonalization: Math techniques
B3 Halting Problem: The main problem
B4 Reduction: Comparing problems
B5 Church-Turing Thesis: Equivalence
B6 Evaluation Workshop (similar to Mod A)
B7 Review (TBA)

Module B Content Goals

CG4 Computability

- Universal TM
- Diagonalization technique
- Halting problem and infinite-loop detection
- "Reduction"
- Church-Turing thesis
- Can explain (i.e., teach) this goal to CS students outside this class.

CG7 Power set

- Understood (i) that the power set is always "larger" than the original set and (ii) why the power set of a countable set is uncountable.

CG3 Interactive computation

- Limitations of the traditional, algorithmic approach to "computability."

Module B Performance Goals

PG4 Critical attitude

- *Critically* analyzed the following points:
  (i) the infinite tape in TM
  (ii) the diagonalization technique
  (iii) no tolerance to errors
  (iv) the lack of interaction in TM
  (v) any other aspects
- *Critically* analyzed the usefulness of the "computability"
- *Critically* analyzed the course materials, cf. your learning

PG6 Initiative

- Chose a research question for the mini research project and started to explore it.

Module A Comprehensive Exercise

- Unknown: “koregawakarukane”
  - Synonym list: “korega” = “maa”, “ruka” = “nnee”, “neene” = “darona”, etc.
  - Sentences whose meaning are known: “maawakanneedarona”, “mosikasitarawakarukamone”, etc.
- Analysis/translation: “<u>korega</u>wakarukane” → “*maa*waka<u>ruka</u>ne” → “maawakan<u>*neene*</u>” → “maawakannee*darona*”

Module A Comprehensive Exercise

- **Task 1**: The collection of sentences whose meaning can be deciphered (in this problem) may be defined as a “theory” derived from axioms and rules of inference. **Identify** the axiom(s) and rule(s) of inference.
- **Task 2**: Represent this problem as a set. Use the predicate notation and try to make the description part (the right of ‘|’) as precise as possible so that each instance could be used as an input to some Turing machine.
- **Task 3**: It has been known that this problem does not admit any algorithm. This suggests that any general programming attempt must be either wrong or forced to loop on at least some input. Suppose that you came up with a Turing machine that makes no errors but could loop. Informally **describe** (part of) the mechanism/behavior of the TM.
- **Task 4**: Since there is no algorithm, this problem is not decidable. But what about the other properties/classes: undecidable, TM-recognizable, non-TM-recognizable, and semi-decidable? For each of these classes, **analyze/explain** whether this problem belong to it (i.e., say whether or not the problem is undecidable, TM-recognizable, etc.).

Module A Review Exercise

Decidable, Semi-decidable, or Non-TM-rec.?

- Option 1: Floor Tiling
- Option 2: Arithmetic System

Group Exercise 1 (Review)

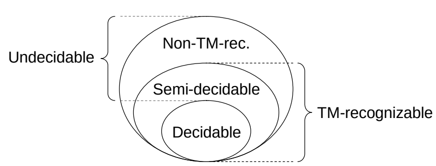
- Consider a couple whose characters are completely opposite. The woman notices the positive aspect of everything they encounter, while the man notices the negative aspect. Note that there is no intermediate state.
  1. Characterize the woman and the man with respect to the computability classes
  2. Give black box representations of TMs M_1 and M_2 that captures the behaviors of woman and man, resp.
  3. Represent the behavior of the couple as a whole by properly connecting M_1 and M_2

schematically...

Review

Computability Classes

- With respect to impossibility, problems can be classified into 3 disjoint sets, based on whether a TM exists and/or always gives an answer.

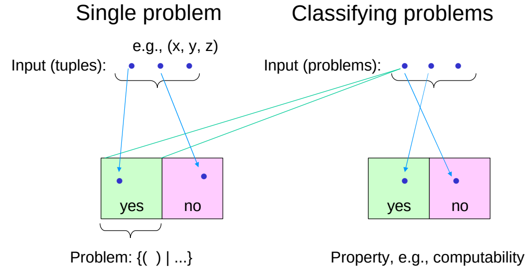


Review

Problems and Properties

Single problem

Classifying problems



Property of “Co-”

Review

- co-*X*
  - Consider the complement of a problem
  - If the complement has property *X*, the original problem is called co-*X*.
- Examples
  - Infinite-loop detection is co-TM-recognizable. I.e., the complement problem (halting problem) is TM-recognizable.
  - Palindrome detection is co-decidable.

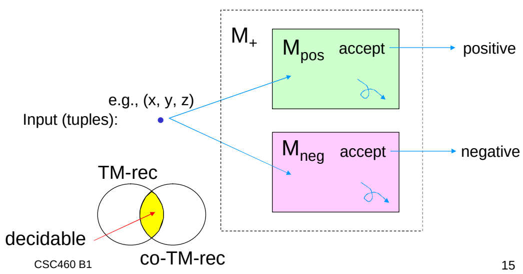
Useful Theorems

Review

- **Theorem**: Decidable ⇔ TM-recognizable *and* co-TM-recognizable
  - Note: Theorem as a statement in a theory, which needs to be proved
- **Corollary** (a theorem easily derivable from another theorem): The complement of a decidable problem is decidable.

Complementary TMs

Decidable ⇔ TM-recognizable *and* co-TM-recognizable



Ex A6/B0

- Part 1: Simulating a TM Using a TM
  - Task 1: Program simulation using a computer
  - Task 2: TM simulation
- Part 2: Mini Research Project
  - Topic(s)?

Relevance to analyzing computability?

Object vs. Meta-Level

- Compiled language vs. compiler language
- Problem vs. properties
  - E.g., computability
- Formal logic vs. informal reasoning
- Your thinking vs. your thought about your thinking
  - E.g., course activities vs. supporting notes

Could we set up a circular connection?

Circularity

- Recursion
- Self-reference
- Reflection
- Self-evaluation

“This sentence is false.”

- The truth value of the above sentence?
- What makes it paradoxical?

Related paradoxes
(to be referred to collectively as liar paradox)
- “I am telling a lie.” (liar paradox)
- “There is a man who shaves all and only the men who don’t shave themselves.” (barber paradox)

Group Exercise 2

- Come up with a statement involving TM(s) corresponding to the liar paradox in the following sense: “This sentence is false.”
  - A TM would either halts or loops.
  - The statement is contradictory regardless of the TM’s halting/looping status.
  - The statement shows the undecidability of one or both of the problems.

To be continued in Ex B1

Universal TM (UTM)

- Given (i) an encoding of the formal definition of a TM, M, and (ii) an input string, i, to M on its tape, simulates the behavior of M on i.
  - I.e.. specification of a TM through the formal definition on the tape.
- Accepts if M accepts i.
  - I.e., exactly mirrors the behavior of M.

UTM Specifics

- Tape input
  - ...□*TM formal def*#*input to the TM*□...
- Mechanism
  - Allocate an unused area of the tape to represent the working tape for the TM (if more space is required, move the surrounding areas outward)
  - Allocate an unused area of the tape to keep record of the internal state of the TM
  - Interpret the TM definition and simulate its behavior on the input to the TM

The Main Problems, Revisited

- Halting problem: $\{(M, i) \mid M \text{ halts on } i\}$
  - Using a UTM, simulate M’s behavior on i
  - If M halts on i: Accept
  - Otherwise: How can we tell?
- Inf.-loop detection: $\{(M, i) \mid M \text{ loops on } i\}$
  - Using a UTM, simulate M’s behavior on i
  - If M loops on i: How can we tell?
  - Otherwise: dies

Unit Summary

- Module B learning goals
- Module A review
- Liar paradox and undecidability
- Universal TM (UTM)
- Preview Exercise B1 “Universal TM”

Midterm Survey

- To be administered and collected by a volunteer