UNIVERSITY OF DUBLIN

TRINITY COLLEGE

Faculty of Science

school of mathematics

SF/JS/SS Maths/TSM **Michaelmas Term 2018**

Mathematics 2361: Computation and Logic

Friday, 14 December 2018 RDS Simmonscourt 9.30–11.30

Prof. Colm Ó Dúnlaing

Attempt 3 questions
Please fold and glue the bottom right-hand corner of each answer book, to ensure anonymity.

1. (a) (4 marks) Give a Turing machine with input alphabet $\{0, 1, 2\}$ which, on input x, interpreted as an integer to the base 3, halts with $x + 1$ (base 3) on the tape. Neither x nor $x + 1$ need have leading zeroes suppressed. What does the machine do on input λ?

Answer

$$q_0 00 R q_0 \quad q_0 11 R q_0 \quad q_0 22 R q_0 \quad q_0 BBL q_1 \quad q_1 20 L q_1$$
$$q_1 12 L q_2 \quad q_1 01 L q_2 \quad q_1 B1 L q_2 \quad q_2 00 L q_2 \quad q_2 11 L q_2$$
$$q_2 22 L q_2 \quad q_2 BBR q_3$$

On input λ:

$$q_0 \vdash q_1 \vdash q_2 B1 \vdash q_3 1$$

(b) (6 marks) Describe how to encode all possible Turing machines T (with binary input alphabet) as bitstrings.

Answer

Let $|\Gamma| = n$ (size of tape alphabet) and $|K| = k$ (number of states). We encode tape symbols, states, and left and right moves in the form $\bar{i} : 10^{i+1}10^{N-i}1$ where $N = n + k + 2$. Let $a_0, \ldots, a_{n-1}$ be the input alphabet, with $a_0 = 0$, $a_1 = 1$, and $a_2 = B$ (the blank symbol). Let $q_0, \ldots, q_{k-1}$ be the set of states with q_0 the initial state.

Represent a_i by $\bar{i}$, q_j by $\overline{n+j}$, move left (L) by $n+k$, and move right (R) by $n+k+1$. Represent every quintuple Q_i as $\hat{Q}_i$ by concatenating five such bitstrings, and encode the Turing machine as

$$1\hat{Q}_1 \ldots \hat{Q}_r 1$$

Crucially: for any bitstring x, there exists at most one factorisation $x = yz$ where y encodes a Turing machine.

(c) (6 marks) Given

$$\text{HALTING} = \{xy : \ x \text{ encodes a Turing machine which halts on input } y\}$$

prove that HALTING is not recursive, i.e., its characteristic function cannot be computed by a Turing machine T which halts on all inputs.

Answer

Suppose that T existed. Construct a new Turing machine T', which on input x, first 'doubles' it so its tape contains xx, then imitates T on input xx; but, if T would halt with output 1 then T' loops, and if T would halt with output 0, then T' halts.

Thus: on input x, if $xx \in \text{HALTING}$ then T' loops, and if $xx \notin \text{HALTING}$ then T' halts.

Let c be the encoding of any Turing machine. If $cc \in \text{HALTING}$ then cc is of the form xy where $T_x(y)$ halts. But the encoding ensures that $x = y = c$, so $T_c(c)$ halts, whereas $T'(c)$ loops. If $cc \notin \text{HALTING}$, then $T'(c)$ halts. Also, cc is not of the form xy where x encodes a Turing machine which halts on input y. In particular, $T_c(c)$ loops, whereas $T'(c)$ halts. In either case, T' differs from T_c, so T' does not exist and T does not exist.

(d) (4 marks) One can show (using a universal Turing machine) that HALTING is recursively enumerable. Is its complement, $\{0,1\}^* \backslash \text{HALTING}$, recursively enumerable? Give a reason.

Answer

No. If HALTING and its complement were recursively enumerable, then HALTING would be recursive.

(Unseen)

---

2. (a) (5 marks) Prove by resolution that the following clauses are inconsistent

$$C\overline{D}, \overline{C}D,$$
$$ABC, A\overline{B}\,\overline{C}, \overline{A}B\overline{C}, \overline{A}\,\overline{B}C,$$
$$AB\overline{D}, A\overline{B}D, \overline{A}BD, \overline{A}\,\overline{B}\,\overline{D}$$

Answer

$$ABC, \overline{C}D \mapsto ABD; \quad ABD, AB\overline{D} \mapsto AB;$$
$$A\overline{B}\,\overline{C}, C\overline{D} \mapsto A\overline{B}\,\overline{D}; \quad A\overline{B}, \overline{D}, A\overline{B}D \mapsto A\overline{B}$$
$$AB, A\overline{B} \mapsto A$$
$$\overline{A}B\overline{C}, C\overline{D} \mapsto \overline{A}B\overline{D}; \quad \overline{A}B\overline{D}, \overline{A}BD \mapsto \overline{A}B;$$
$$\overline{A}\,\overline{B}C, \overline{C}D \mapsto \overline{A}\,\overline{B}D; \quad \overline{A}\,\overline{B}D, \overline{A}\,\overline{B}\,\overline{D} \mapsto \overline{AB}$$
$$\overline{A}B, \overline{A}\,\overline{B} \mapsto \overline{A}$$
$$A, \overline{A} \mapsto \square$$

(Unseen)

(b) (4 marks) To 'shortcut' resolution by cancelling two pairs of complementary literals is a serious mistake. Explain why, using UVW and $\overline{V}\,\overline{W}X$ as an example. (The invalid 'resolvent' would be UX.)

Answer

The truth-assignment $U \mapsto 0, V \mapsto 0, W \mapsto 1, X \mapsto 0$ makes UVW and $\overline{V}\,\overline{W}X$ both true but the 'resolvent' UX is false. This simultaneous cancellation does not preserve truth in the interpretation, so it is invalid.

(c) (5 marks) Give a proof in propositional logic (Sentential Calculus, or SC) of the following result.

$$\neg\neg A \vdash_{\mathrm{SC}} A$$

You may assume $\vdash_{\mathrm{SC}} X \implies X$ for every formula X.

Answer

Proof.
1. $\neg\neg A$ (given)
2. $\neg A \implies \neg\neg A$ (1,I,MP)
3. $\neg A \implies \neg A$ (permitted assumption).
4. $(\neg A \Rightarrow \neg\neg A) \implies ((\neg A \Rightarrow \neg A) \Rightarrow A)$ (Ax III)

5. A (2,3,4, MP twice).

(d) (6 marks) Sketch a proof that if A is a tautology then $\vdash_{\text{SC}} A$. You may assume that the empty clause can be derived from a CNF equivalent to $\neg A$ by repeated resolution, and that if $C \vee L$ and $D \vee \neg L$ are clauses with $C \vee D$ nonempty, then the resolvent $C \vee D$ can be deduced from these clauses in the Sentential Calculus.

Answer

Construct a CNF

$$C_1 \wedge \ldots \wedge C_N$$

provably equivalent to $\neg A$. From this formula, each clause C_j can be deduced. From the previous part of this question, resolvents can be deduced within SC so long as they remain nonempty. The empty clause can be derived, so there exists some literal L such that

$$C_1, \ldots, C_N \vdash L \quad \text{and} \quad C_1, \ldots, C_N \vdash \neg L$$

Thus, since $\neg A$ is equivalent to this conjunction, and invoking the Deduction Theorem,

$$\neg A \implies \neg L \quad \text{and} \quad \neg A \implies L$$

Using Axiom III:

$$\neg A \implies \neg L \implies ((\neg A \implies L) \implies A)$$

and MP twice, we deduce A. ∎

---

3. (a) (4 marks) Given a first-order language, and an interpretation I of that language, define, by induction on the complexity of formulae, the relation

$$I, \sigma \models A$$

where σ is any snapshot.

You need not define the objects $\sigma_{i \mapsto d}$ nor t^σ (the value of t under snapshot σ) where t is a term.

Answer

- A an atomic formula $P(t_1, \ldots, t_n)$: $P^I(t_1^\sigma, \ldots, t_n^\sigma)$.
- A is $\neg B$: $I, \sigma \models A$ iff not $I, \sigma \models B$.

- A is $B \implies C$: *not* $I, \sigma \models A$ iff $I, \sigma \models B$ and not $I, \sigma \models C$
- A is $\forall x_i B$: $I, \sigma \models A$ if for every $d \in D$ (the domain of I), $I, \sigma_{i \mapsto d} \models B$.

(b) (3 marks) Define what it means for a term t to be free for a variable x_i in a formula $A(x_i)$.

Answer

No free occurrence of x_i in $A(x_i)$ is within the scope of a quantifier $(\forall x_j \ldots)$ where x_j is any variable occurring in t.

(c) (9 marks) Let $A(x_i)$ be a formula, t a term free for x_i in $A(x_i)$, I an interpretation, σ a snapshot, and $\tau = \sigma_{i \mapsto t^\sigma}$. Then one can prove, by induction on the complexity of A, that

$$I, \sigma \models A(t) \quad \text{if and only if} \quad I, \tau \models A(x_i)$$

Prove two cases: (i) where A is an atomic formula, and (ii) where $A(x_i)$ has the form $\forall x_j B(x_i, x_j)$, where $x_j \neq x_i$ and x_i occurs free in $B(x_i, x_j)$.

You may assume without proof that if x_j does not occur in t then t^σ is independent of σ_j, and if $t_r(x_i)$ is any term then $t_r(x_i)^\tau = t_r(t)^\sigma$.

Answer

(i) A is $P(t_1(x_i), \ldots, t_n(x_i))$. Then $I, \sigma \models A(t)$ if and only if $P^I(t_1(t)^\sigma, \ldots, t_n(t)^\sigma)$, and $I, \tau \models A$ if and only if $P^I(t_1(x_i)^\tau, \ldots, t_n(x_i)^\tau)$. But $t_r(x_i)^\tau = t_r(t)^\sigma$, $1 \leq r \leq n$, so the truth-values are the same.

(ii) A is $(\forall x_j B(x_i, x_j))$ where $x_j \neq x_i$ and x_i occurs free in $B(x_i, x_j)$. Since x_i occurs free within the scope of $(\forall x_j \ldots)$, x_j does not occur in t.

$I, \sigma \models A(t)$ if and only if for all $d \in D$,

$$I, \sigma_{j \mapsto d} \models B(t, x_j)$$

Given d, suppose $\sigma' = \sigma_{j \mapsto d}$, so

$$I, \sigma' \models B(t, x_j)$$

By induction, this is equivalent to

$$I, \sigma'_{i \mapsto t^{\sigma'}} \models B(x_i, x_j)$$

Since x_j does not occur in t, $t^\sigma = t^{\sigma'}$. So the above is equivalent to

$$I, \sigma'_{i \mapsto t^\sigma} \models B(x_i, x_j)$$

Since $\sigma'_{i \mapsto t^\sigma} = \tau_{j \mapsto d}$, this is equivalent to:

$$I, \tau_{j \mapsto d} \models B(x_i, x_j)$$

for given d. Thus for all d,

$$I, \sigma_{j \mapsto d} \models B(t, x_j) \quad \text{if and only if} \quad I, \tau_{j \mapsto d} \models B(x_i, x_j)$$

so $I, \sigma \models A(t)$ if and only if $I, \tau \models A(x_i)$. ▮

(d) (4 marks) Give a formula $A(x_1)$ with just x_1 free, and a suitable interpretation I such that $I \models A(x_1)$ but not $I \models A(x_2)$.

Answer

Let I have domain $\mathbb{N}$, with equality interpreted as usual.

$$\text{Let } A(x_1) \text{ be } \exists x_2 (x_1 \neq x_2), \quad \text{true under every snapshot; and}$$
$$A(x_2) : \exists x_2 (x_2 \neq x_2) \quad \text{false under every snapshot.}$$

(Mentioned in passing in the notes.)

---

4. (a) (6 marks) Define when two sets X, Z of natural numbers are *recursively inseparable.* Using the Fixed Point Theorem (without proving it), show that

$$C_0 = \{m : \ \phi_m(0) \downarrow 0\} \quad \text{and} \quad C_1 = \{m : \ \phi_m(0) \downarrow 1\}$$

are recursively inseparable.

Answer

'Recursively inseparable' means: for every $Y \subseteq \mathbb{N}$, if $X \subseteq Y$ and $Y \cap Z = \emptyset$ then Y cannot be recursive.
Let Y be any subset of $\mathbb{N}$ such that $C_0 \subseteq Y$ and $C_1 \cap Y = \emptyset$.
Choose any $a \in C_0$ and $b \in C_1$. Let $f : \mathbb{N} \to \mathbb{N}$ be

$$m \mapsto \begin{cases} b & \text{if } m \in Y \\ a & \text{if } m \notin Y \end{cases}$$

If Y is recursive then f is recursive, and there exists an m such that $\phi_{f(m)} = \phi_m$.
Note $f(m) \in \{a, b\}$, so $\phi_m = \phi_a$ or $\phi_m = \phi_b$.
If $\phi_m = \phi_a$ then $m \in C_0$, so $f(m) = b$ and $\phi_{f(m)} = \phi_b \neq \phi_m$.
If $\phi_m = \phi_b$ then $m \in C_1$, so $f(m) = a$ and $\phi_{f(m)} = \phi_a \neq \phi_m$.
This contradiction shows that Y is not recursive. ▮

(b) (8 marks) Recall the existence of a primitive recursive relation

$$\text{Result}\,(m, n, y, S)$$

to describe Turing machine computations, with a corresponding formula $\text{Result}\,(x_1, x_2, x_3, x_4)$ of Peano Arithmetic (PA) expressing that relation.
(i) Give a formula of PA expressing the relation

$$\phi_m(n) \downarrow y$$

(ii) Deduce that the set XX of theorems of PA, and the set ZZ of formulae of PA false in $\mathbb{N}$, are recursively inseparable.

Answer

(i) The formula

$$\text{Converges}\,(x_1, x_2, x_3): \quad \exists x_4\, \text{Result}\,(x_1, x_2, x_3, x_4)$$

expresses the relation $\phi_m(n) \downarrow y$.
(ii) If $m \in C_0$, then $\phi_m(0) \downarrow 0$, so there exists an encoding S of a computation such that

$$\text{Result}\,(\overline{m}, \overline{0}, \overline{0}, \overline{S})$$

is a theorem of PA, and therefore

$$\vdash_{\text{PA}}\ \text{Converges}\,(\overline{m}, \overline{0}, \overline{0})$$

If $m \in C_1$, then there exists an encoding S of a halting computation, where S depends uniquely on m, such that

$$\text{Result}\,(m, 0, 1, S)$$

is true of natural numbers m, S; therefore

$$\text{Result}\,(\overline{m}, \overline{0}, \overline{1}, \overline{S})$$

is a theorem of PA and therefore true in $\mathbb{N}$. Since S is unique, for all S', whether or not $S' = S$,

$$\text{Result}\,(m, 0, 0, S')$$

is false, so

$$\exists x_4\, \text{Result}\,(\overline{m}, \overline{0}, \overline{0}, x_4)$$

is false in $\mathbb{N}$, so

$$\text{Converges}\,(\overline{m}, \overline{0}, \overline{0})$$

is false in $\mathbb{N}$.
Therefore, $C_0 \subseteq$ XX and $C_1 \subseteq$ ZZ, so XX and ZZ are recursively inseparable. ∎

(c) (6 marks) Deduce that XX, ZZ, and TT , the set of formulae *true* in $\mathbb{N}$, are not recursive.

Answer

Since $\mathbb{N}$ is a model of PA, XX and ZZ are disjoint. Therefore XX separates C_0 from C_1. Hence XX cannot be recursive.

Also, ZZ separates C_0 from C_1 and is not recursive. Now,

$$\mathbb{N} = \text{TT} \cup \text{ZZ} \cup W$$

where W consists of those natural numbers which do not encode any formula of PA. It is recursive. So, if TT were recursive, then its complement $\text{ZZ} \cup W$ would also be recursive, and since W is recursive, ZZ would be recursive, which is isn't. Therefore TT is not recursive.