Math 17

Winter 2015

Turing Machine UD

Here are instructions for a Turing machine UD with the following property: Every Diophantine set $X \subseteq \mathbb{N}$ has a code (N, b, c) such that, when UD is started with input (a, N, b, c), it halts if and only if $a \in X$. (UD stands for "universal Diophantine"; it is a single machine that semidecides every Diophantine set.)

If

$$X = \{a \mid (\exists x_1 \cdots \exists x_n)(P(a, x_1, \ldots, x_n) = Q(a, x_1, \ldots, x_n))\}.$$

then b and c are codes for P and Q. Interpreted as Cantor codes for sequences of length n, they give sequences of numbers that can be interpreted as instructions for evaluating the monomials in P and Q.

1. STAR

   Place the tape head at the beginning of the tape.

2. NEW

   Extends the input sequence to (a, N, b, c, x), with $x = 0$.

3. VACANT;WRITE(7)

   The symbol 7 will serve as a marker.

4. COPY(3)

5. DECODEX$^{(2)}$

   This produces the code for P, decoded into a sequence of numbers representing instructions, interspersed with blocks of the symbol X.

6. VACANT;WRITE(7)

7. COPY(1)

8. COPY(5)

9. DECODEX$^{(2)}$

   This produces the sequence of inputs a, x_1, ..., interspersed with blocks of the symbol X.

10. WRITE(5)

11. EVAL

    This machine, still to be described, will produce a sequence of numbers that are the values of the monomials of P with inputs a, x_1, ... That is, it will produce a string of 1's and 0's, and the number of 1's will be $P(a, x_1, \ldots)$.

12. WRITE(5)

13. VACANT;WRITE(7)

14. COPY(4)

15. DECODEX$^{(2)}$

    This produces the code for Q, decoded into a sequence of numbers representing instructions, interspersed with blocks of the symbol X.

16. VACANT;WRITE(7)

17. COPY(1)

18. COPY(5)

19. DECODEX$^{(2)}$

    This produces the sequence of inputs a, x_1, $\ldots$, interspersed with blocks of the symbol X.

20. WRITE(6)

21. EVAL

    Now this machine will produce a string of 1's and 0's, and the number of 1's will be $Q(a, x_1, \ldots)$.

22. WRITE(6)

23. CHECK

    This machine answers YES or NO, depending on whether the number of 1's between the 5's equals the number of 1's between the 6's.

    (a) If CHECK answers YES, halt in state YES.

    (b) If CHECK answers NO, go on to the next step.

24. STAR

25. MOVE RIGHT UNTIL THE FIRST CELL MARKED WITH 7

26. REPLACE SYMBOLS IN THIS CELL AND ALL CELLS TO THE RIGHT WITH λ

    Now the tape contains only the extended input (a, N, b, c, x).

27. INC

    Add 1 to x.

28. GO TO STEP (3).

Math 17
Winter 2015
Machine EVAL

Here are instructions for a Turing machine EVAL with the following property. Suppose that after the next-to-last occurrence of the symbol 7 on the tape is found a sequence of numbers representing instructions for computing a polynomial P, possibly interspersed with blocks of the symbol X, and after the last occurrence of the symbol 7 on the tape is found a sequence of inputs a, x_1, ... for P, again possibly interspersed with blocks of the symbol X. Then EVAL will write at the end of the tape a string of 0's and 1's, in which the number of 1's is equal to $P(a, x_1, \dots)$.

1. VACANT

2. MOVE LEFT UNTIL THE SYMBOL 7 IS FOUND

   We will say "MOVE LEFT TO 7".

3. MOVE LEFT TO 7

   We are now at the beginning of the instructions.

4. MOVE RIGHT TO 0

   We search until we find the beginning of an instruction.

5. WRITE(X)

   We will cross out instructions as we implement them.

6. MOVE RIGHT TO 0 OR 1

7. IF READ(0) THEN HALT IN STATE DONE

   OTHERWISE, GO ON TO THE NEXT STEP

   The number 0 represents the instruction to stop, as the polynomial has been computed.

8. WRITE(X)

   Since this cell wasn't a 0, it must be a 1. We cross it out and continue.

9. MOVE RIGHT TO 0 OR 1

10. IF READ(0) THEN NEW; INC; GO TO STEP 1.

    OTHERWISE GO ON TO NEXT STEP

    If this cell was a 0, we just finished reading the string 01, meaning the number 1, which represents the instruction "start a new monomial, and set its value to 1." We do that, then return to the first step, look for the next instruction.

    Note, we don't cross out this 0, since it belongs to an instruction we haven't started on yet.

11. WRITE(X)

    We cross out the 1 we just read. So far, we know the instruction will be to multiply the monomial by some input. Since we've just started on the instruction, the next input is the first input, namely a.

12. MOVE RIGHT TO 0 OR 1

13. IF READ(0) THEN GO TO STEP 20

    OTHERWISE GO TO THE NEXT STEP

    If this cell was a 0, we just finished reading the string $01 \cdots 1$, meaning some number that represents the instruction "multiply the current monomial by the next input." Step 20 will begin that process.

    Otherwise, the instruction will be "multiply the current monomial by x_i," where x_i is some further input. The next input is not the one we want, so we will mark it to indicate that.

14. WRITE(X)

    We are still reading, and crossing out, the current instruction.

15. MOVE RIGHT TO 7

    This finds the location of the inputs.

16. MOVE RIGHT TO 0

17. WRITE(8)

    The effect of this step is to temporarily mark the next input as not the input wanted by this instruction.

18. MOVE LEFT TO 7; MOVE LEFT TO 7; MOVE RIGHT TO 0 OR 1.

    Keep on reading the instruction we are working on now.

19. GO TO STEP 13

    Step 13 continues to carry out the instruction.

20. MOVE RIGHT TO 7

    This finds the location of the inputs.

21. MOVE RIGHT TO 0

    This finds the next input (the first one we have not marked by changing its 0 to an 8).

22. MULT(NOW)

    This multiplies the current monomial by that input.

23. VACANT; MOVE LEFT TO 7

    This puts us at the beginning of the inputs.

24. MOVE RIGHT, REPLACING EVERY 8 WITH 0, UNTIL READ(λ)

    We restore the inputs for the next instruction.

25. GO TO STEP 1

    We have carried out this instruction, and will now look for the next one.