CS 332 – Spring 2021, Assignment 6

Colaborators: None

Answer 1

(a) Yes it will. Since the machine M_1 will write on the tape the substring 1010 before it reaches its final step (ie: before it rejects), then H will be able to read that substring and accept, before M_1 halts and H rejects.

(b) No, because at no point before M_2 halts is the string 1010 written on the tape. M_2 on input 101 will halt at the third iteration in H, and as such, H will reject $M_2, 101$.

(c) $L(H) = \{\langle M, w\rangle |$ At some point while running w on M, and before M stops, the substring "1010" will be present on the tape. $\}$

(d) No, since the inner loop for H runs forever unless stopped, and it only stops if the substring is found or if M halts, then if M doesn't halt on every input, and doesn't include the substring for every input, then it is possible for a machine M that taking an input w, will loop forever without ever producing the string, thus H will not be decidable.

Answer 2

(a) Each element of the set W is made up of a three-tuple (s, t, l), with each of these equaling the amount of spaces, tabs and newlines that exist in that specific program. Which means that $s, t, l \in N$, since there can't be a negative nor a fractional amount of any of these things.

From here we form different stages, where in each stage we list the tuples so that for Stage x $s + t + l = x$. This will be very similar to the prove of the tuple $N \times N$ seen in class, since the stages themselves are finite, since there are only so many tuples adding to that stage number. As such, any (s, t, l) will be listed in stage $s + t + l$.

(b) If we know that both the S and T sets are countable, then we can create a function $f(x)$ that maps the elements of S and T into stages, where stage 1 will have the first elements of S and T, and stage 2 the second element of S and T, and hence on hence forward. Thus the union of the two sets is coutable if the sets themselves are countable.

Answer 3

(a) If we list these sequences out with all digits starting at 1, and then increment the first until seven, and then increment the second and repeat, until the second as hit seven, and hence on hence forward, we reach the following f table:

1, 1, 1, 1, 1, 1, 1, 1, ...

2, 1, 1, 1, 1, 1, 1, 1, ...

3, 1, 1, 1, 1, 1, 1, 1, ...

4, 1, 1, 1, 1, 1, 1, 1, ...

5, 1, 1, 1, 1, 1, 1, 1, ...

6, 1, 1, 1, 1, 1, 1, 1, ...

7, 1, 1, 1, 1, 1, 1, 1, ...

1, 2, 1, 1, 1, 1, 1, 1, ...

We get a diagonal of 1, 1, 1, 1, 1, 1, 1, 1, ... Yet if we build a $b = 2, 3, 4, 5, 6, 7, 1, 2, ...$ then $b \neq f(1)$, and $b \neq f(2)$, and .... So $b \notin f$. Thus the sequence of Tetrominos is not countable.

(b) If we list the sequences of collision free numbers in an f table as seen below:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, ...

2, 3, 4, 5, 6, 7, 8, 9, 10, 11, ...

3, 4, 5, 6, 7, 8, 9, 10, 11, 12, ...

4, 5, 6, 7, 8, 9, 10, 11, 12, 13, ...

5, 6, 7, 8, 9, 10, 11, 12, 13, 14, ...

6, 7, 8, 9, 10, 11, 12, 13, 14, 15, ...

7, 8, 9, 10, 11, 12, 13, 14, 15, 16, ...

8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ...

9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ...

10, 11, 12, 13, 14, 15, 16, 17, 18, 19, ...

11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...

We get a diagonal of 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, ... If we build a $b = 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, ...$ then $b \neq f(1)$, and $b \neq f(2)$, and .... So $b \notin f$. Thus the sequence of collision-free numbers is not countable.

Answer 4

(a) If M accepts on input w, then $L(N) = \{\langle M, w\rangle|$ If M accepts w then accept, otherwise reject. $\}$. In this case $L(N)$ is excited.

(b) If M does not accept on input w then $L(N) = \{\langle M, w\rangle|$ If $w = 01$ accept, otherwise reject. $\}$. In this case $L(N)$ is not excited.

(c) (i) accept — (ii) reject

Answer 5

(a) $UA = \{\langle M\rangle|$ If there is a string $x = 1^n$ for some $n \in Z_0^+$ where M accepts x $\}$.

(b) If at any point one of these unary strings causes a loop in M, then the entire machine will just loop forever, when it could have, potentially, accepted a later string. Thus this is not a recognizable TM for UA.

(c) Let s_1, s_2, s_3 be all the unary strings. The following TM recognizes UA:

"On input $\langle M\rangle$, where M is a TM:

1. Repeat the following for $i = 1, 2, 3, ...$:
2. Run M for i steps on each input s_1, s_2, s_3, ...
3. If M accepts in any of these, accept. Otherwise continue."

(d) Assume, for the sake of contradiction, that AU is decidable by some TM R. That is, there is a TM R that accepts $\langle M\rangle$ if $L(M)$ includes a unary string, and rejects if there are no unary strings in $L(M)$. We will use R to construct a new TM S that decides A_{TM}.

Input: A basic TM M over the alphabet $\{0, 1\}$, and a string $w \in \{0, 1\}^*$

1. Construct the following TM M_1:
   $M_1 =$ "On input x:
   1. If x has any number of 0's reject
   2. If x is only made up of 1's, then run M on input w, if it accepts accept.
2. Run R on $\langle M_1\rangle$, if it accepts accept.

This TM will only accept if M accept input w, otherwise it will reject, thus this TM will decide A_{TM}, something we know not to be possible, as such there can be no TM R.