1.

WTS：term(x) terminates.

Proof:

.

Let be x after the ith iteration and be y after the ith iteration.

Define p(i): after the ith iteration of the loop (if it occurs), and and . I will prove that using simple induction on i.

Base case: (by initialization). Since , , hence . By code (line4) we know that if there is a iteration then Since , x is an integer, hence is also an integer. So p(0) follows.

Inductive step:

Let and assume p(i), that is and and . Show that p(i+1) follows. If there is an (i+1)th loop iteration.

Then by code, , (# by induction hypothesis we know that . If there is a next iteration, then by code (line4), we know that . Since there is an (i+1)th loop iteration, by the loop condition we know that , and by induction hypothesis we know that hence , and by induction hypothesis we know that , hence , since , hence , hence . Since by induction hypothesis, , is also an integer. So p(i+1) follows.

Hence we’ve proved that after the ith iteration of the loop (if it occurs), and and . Then we will prove termination by this loop iteration.

Try the sequence , since by loop iteration we know that . Hence Hence . Hence each element of the sequence is a natural number. It remains to show that the sequence is strictly decreasing. Suppose that there is an (i+1)th iteration of the loop, then by loop iteration we know that, hence since is monotonic increasing. So the sequence is strictly decreasing.

Since a strictly decreasing sequence in is finite, and hence has a last (smallest) element. Thus , the loop terminates.

Hence we’ve proved that term(x) terminates.

4.

(a)

WTS:

Let RE be the set of regular expressions over the alphabet ={0,1}, Define p(r):, I will show that by structural induction on r. Let

Basis: Let r,

For r, then , hence by the definition of RE, we know that , and , hence . And , hence .

For r, then , hence by the definition of RE, we know that , and , hence (since ). And , hence .

For r, then , hence by the definition of RE, we know that , and , hence (since ). And , hence .

For r, then , hence by the definition of RE, we know that , and , hence (since ). And , hence .

So p(r) holds.

Inductive step:

Let t, s, assume p(t), p(s), that is , . Let be those regular expression. I will show that (t+s), (ts), () follows. And I will prove this in 3 cases.

1. To show that (t+s) follows.

Take , since and , hence by the definition of RE, we know that . And we will show that and .

First, show that , let , then , let xx be that regular expression, since xx, xx must be one of the element of , hence must be one of the element of (since by inductive hypothesis and ), hence , hence .

Second, show that , let , hence must be one of the element of . Then , y=, let yy be that regular expression (since by inductive hypothesis and ), hence , hence , that is

Hence and . That is .

So (t+s) follows.

1. To show that (ts) follows.

Take , since and , hence by the definition of RE, we know that. And we will show that and .

First, show that , let , then , let xx be that regular expression, since xx, xx must be a concatenation of one of the element of , Let’s show it as

Hence must be one of the element of must be one of the element of (since by inductive hypothesis and ), hence , hence .

Second, show that , let , hence must be a concatenation of one of the element of . Let show it as , let them be their value. Then , =, let be that regular expression (since by inductive hypothesis and ), hence , hence , that is

Hence and .

So (ts) follows.

1. To show that () follows.

Take , since , hence by the definition of RE, we know that . And we will show that and .

First, show that , let , then , let be that regular expression, since , must be a star of some of the elements of ,let those elements be (since the string are always finite, we only suppose 3 to be the number of the element, and this will cover all the cases) and . And by the definition of the reverse of the string, we know that , and must be one of the element of (since by inductive hypothesis ), hence , hence .

Second, show that , let , hence must be one of the element of , hence y must be a concatenation of some elements of , let them be (since the string are always finite, we only suppose 3 to be the number of the element, and this will cover all the cases).Then y= . And ,===, let be that regular expression (since by inductive hypothesis ), and by the definition of the reverse of the string , and since hence That is y=

Hence and . That is .

So () follows.

Hence we’ve proved that

(b) WTS:

Let RE be the set of regular expressions over the alphabet ={0,1}, Define p(r):, I will show that by structural induction on r. Let

Basis: Let r,

For r, then , hence by the definition of RE, we know that , and , hence . And , hence .

For r, then , hence by the definition of RE, we know that , and , hence (). And , hence .

For r, then , hence by the definition of RE, we know that , and , hence (). And , hence .

For r, then , hence by the definition of RE, we know that , and , hence (). And , hence .

So p(r) holds.

Inductive step:

Let t, s, assume p(t), p(s), that is , . Let be those regular expression. Hence we can write the above as . I will show that (t+s), (ts), () follows. And I will prove this in 3 cases.

1. To show that (t+s) follows.

Take , since and , hence by the definition of RE, we know that . And we will show that and .

First, show that , let then , let y be that value, since , xy must be one of the element of , if xy is a element of L(t), then then (by inductive hypothesis ,if xy is a element of L(s), then then (by inductive hypothesis

Hence .

Second, show that , let (by inductive hypothesis , hence must be one of the element of. If x is an element of that is If x is an element of that is Hence

Hence and . That is .

So (t+s) follows.

1. To show that (ts) follows.

Take , since and and , hence by the definition of RE, we know that. And we will show that and .

First, show that , let , then , let y be that value, since , xy must be a concatenation of one of the element of , Let’s show it as

Hence if |x|<||, then x is some first part of then . (by induction hypothesis

If |, then x is the concatenation of with some first part of, then (by induction hypothesis

Hence .

Second, show that , let by inductive hypothesis ,

If , then , let y be that value, let , then xys, since RE is over hence , hence =, hence .

If , then , hence , hence .

So (ts) follows.

1. To show that () follows.

Take , since , hence by the definition of RE, we know that . And we will show that and .

First, show that , let , , hence (since the string are always finite, we only suppose 3 to be the number of the element, and it will cover all the cases).

if |x|<||, then x is some first part of then . (by induction hypothesis

If ||+||>|, then x is the concatenation of with some first part of, then (by induction hypothesis

If, then x is the concatenation of with some first part of, then (by induction hypothesis

Hence

Second, we will show that .Let ==. Since the string are always finite, we only suppose the first 3 sets to be the constraint of x, and it will cover all the cases, that is = (by inductive hypothesis ,

If , then , let y be that value, then xy , hence , hence .

If , then , hence , hence .

If , then , hence , hence .

So () follows.

Hence we’ve proved that

(c)

WTS: If does not contain the Kleene star, then |L(r)| is finite.

Proof:

Let RE be the set of regular expressions over the alphabet ={0,1}, Define p(r):, I will show that by structural induction on r. Let

Basis: Let r,

For r, obviously does not contain the Kleene star. And L(r)={}, hence |L(r)|=0, hence .

For r, obviously does not contain the Kleene star. And L()={}, hence |L(r)|=1, hence .

For r, obviously does not contain the Kleene star. And L()={}, hence |L(r)|=1, hence .

For r, obviously does not contain the Kleene star. And L()={}, hence |L(r)|=1, hence .

So p(r) holds.

Inductive step:

Let t, s, assume p(t), p(s), that is , . I will show that (t+s), (ts), () follows. And I will prove this in 3 cases.

1. To show that (t+s) follows.

I will show that (t+s) follows in 4 cases.

Hence by the induction hypothesis, we know that and . (, .)

Since , t+s must also .

And must be a finite number since and .

Hence p(r) holds in this case.

Then t+s must also contain the Kleene star.

Since the assumption is false, p(r) is vacuously true in this case.

Then t+s must also contain the Kleene star.

Since the assumption is false, p(r) is vacuously true in this case.

Then t+s must also contain the Kleene star.

Since the assumption is false, p(r) is vacuously true in this case.

Hence (t+s) follows.

1. To show that (ts) follows.

I will show that (ts) follows in 4 cases.

Hence by the induction hypothesis, we know that and . (, .)

Since , ts must also .

And must be a finite number since and .

Hence p(r) holds in this case.

Then ts must also contain the Kleene star.

Since the assumption is false, p(r) is vacuously true in this case.

Then ts must also contain the Kleene star.

Since the assumption is false, p(r) is vacuously true in this case.

Then ts must also contain the Kleene star.

Since the assumption is false, p(r) is vacuously true in this case.

Hence (ts) follows.

1. To show that () follows.

Since () itself contains the Kleene star.

Since the assumption is false, p(r) is vacuously true in this case.

Hence () follows.

Hence we’ve proved that If does not contain the Kleene star, then |L(r)| is finite.

5.

WTS: any DFA that accepts has at least nine states, not including dead states.

Proof:

Let ={a,b,c} and , I will prove that any DFA that accepts has at least nine states, not including dead states by contradiction. The proof is the following.

Assume, for the sake of contradiction, the negation of what we are proving, that is there is a DFA that accepts has less than 9 states not including dead states. That it be that value.

Then for this DFA, first ignoring the dead state, we know that if we choose 9 strings over , then there must be at least two strings that will end with the same states since we’ve assumed that this DFA has less than 9 states.

Let , ,,,, and , be these 9 strings. Since each of them can be transferred into a string of by concatenating some string after them, hence by the definition of DFA, none of them are in the dead state. Then at least two strings of them will end with the same states as proved above.

For the strings end with the same states, let s be a string over , then the concatenation of these strings with s must also end in the same state, since s=s and these strings end with the same states.

Hence these concatenations must be both accepted or both rejected.

Since at least two strings of will end with the same states, that means there are two strings from for any string s over , the concatenations of these two with s must always be both accepted or both rejected. (1)

We also know that:

Pair 1 and :

Choose s = aaa.

Thens = aaa, rejected; s = aaaa, accepted.

Pair 2 and :

Choose s = aa.

Then s = aa, rejected; s = aaaa, accepted.

Pair 3 and :

Choose s = a.

Then s = a, rejected; s = aaaa, accepted.

Pair 4 and :

Choose s = .

= is rejected; = aaaa is accepted.

Pair 5 and :

Choose s = bbb.

Then s = bbb, rejected; s = bbbb, accepted.

Pair 6 and :

Choose s = c.

Thens = c, rejected; s = cccc, accepted.

Pair 7 and :

Choose s = b.

Then s = b, rejected; s = bbbb, accepted.

Pair 8 and :

Choose s = ccc.

Then s = ccc, rejected; s = cccc, accepted.

Pair 9 and :

Choose s = aa.

Then s = aaa, rejected; s = aaaa, accepted.

Pair 10 and :

Choose s = a.

Thens = a, rejected; s = aaaa, accepted.

Pair 11 and :

Choose s = .

= a is rejected; = aaaa is accepted.

Pair 12 and :

Choose s = bbb.

Then s = abbb, rejected; s = bbbb accepted.

Pair 13 and :

Choose s = c.

Then s = ac, rejected; s = cccc accepted.

Pair 14 and :

Choose s = b.

Then s = ab, rejected; s = bbbb accepted.

Pair 15 and :

Choose s = ccc.

Then s = accc, rejected; s = cccc accepted.

Pair 16 and :

Choose s = a.

Then s = aaa, rejected; s = aaaa, accepted.

Pair 17 and :

Choose s = .

= aa is rejected; = aaaa is accepted.

Pair 18 and :

Choose s = bbb.

Then s = aabbb, rejected; s = bbbb accepted.

Pair 19 and :

Choose s = c.

Then s = aac, rejected; s = cccc accepted.

Pair 20 and :

Choose s = b.

Thens = aab, rejected; s = bbbb accepted.

Pair 21 and :

Choose s = ccc.

Then s = aaccc, rejected; s = cccc accepted.

Pair 22 and :

Choose s = .

= aaa is rejected; = aaaa is accepted.

Pair 23 and :

Choose s = bbb.

Then s = aaabbb, rejected; s = bbbb accepted.

Pair 24 and :

Choose s = c.

Then s = aaac, rejected; s = cccc accepted.

Pair 25 and :

Choose s = b.

Then s = aaab, rejected; s = bbbb accepted.

Pair 26 and :

Choose s = ccc.

Then s = aaaccc, rejected; s = cccc accepted.

Pair 27 and :

Choose s = .

= aaaa is accepted; = b is rejected.

Pair 28 and :

Choose s = .

= aaaa is accepted; = ccc is rejected.

Pair 29 and:

Choose s = .

= aaaa is accepted; = bbb is rejected.

Pair 30 and :

Choose s = .

= aaaa is accepted; = c is rejected.

Pair 31 and:

Choose s = c.

Then s = bc, rejected; s = cccc accepted.

Pair 32 and :

Choose s = b.

Then s = bb, rejected; s = bbbb accepted.

Pair 33 and :

Choose s = ccc.

Then s = bccc, rejected; s = cccc accepted.

Pair 34 and :

Choose s = b.

Then s = cccb, rejected;s = bbbb accepted.

Pair 35 and :

Choose s = ccc.

Then s = cccccc, rejected; s = cccc accepted.

Pair 36 and :

Choose s = ccc.

Then s = bbbccc, rejected; s = cccc accepted.

Hence for any two strings of , there is a string s over , make the concatenations of these two with s be one accepted and one rejected.

---><--- contradiction! The conclusion above is exactly the negation of (1) which we’ve assumed before. Since assuming that there is a DFA that accepts has less than 9 states leads to a contradiction, the assumption is false.

Hence we’ve proved that any DFA that accepts has at least nine states, not including dead states.