a) precondition: n is an integer

postcondition:

1> s is an integer in [0, 10]

2> if n is a positive integer with even number of digits, (n rem 11) = (11-s); if n is a positive integer with odd number, (n rem 11) = s. if n is a negative integer with even number of digits, (n rem 11) = s; if n is a negative integer with odd number, (n rem 11) = (11-s). if n is 0, n=s=0.

3> DivisibleBy11(n) returns whether the alternating sum of the digits in n, read from left to right is equal to zero.

b)

lemma 1: $\forall \mathrm{a} \in \mathbb{Z}, \forall \mathrm{i} \in \mathbb{Z}^{+}, \mathrm{a} * 10^{i} \equiv -\mathrm{a} * 10^{i-1} (mod\ 11)$

proof:

$\mathrm{a} * 10^{i} + \mathrm{a} * 10^{i-1} = 11 * \mathrm{a} * 10^{i-1}$

$\mathrm{so}, \mathrm{a} * 10^{i} \equiv -\mathrm{a} * 10^{i-1} (mod\ 11)$ (from the property of congruence modulo)

lemma 2: $\forall \mathrm{a} \in \mathbb{Z}, \mathrm{a\ rem\ } 11 \equiv \mathrm{a}\ (mod\ 11)$ (directly from the property of congruence modulo)

lemma 3: $\forall \mathrm{a} \in \mathbb{Z}, \mathrm{a} \equiv 11 \pm \mathrm{a}\ (mod\ 11)$ (directly from the property of congruence modulo)

lemma 4: for any positive integer n, write n in decimal system, if n has q digits, assume n = m[q]…m[2]m[1] , when the algorithm DivisibleBy11(n) performs the while loop the i th time just finished the 6th line, (notice: for every positive integer, DivisibleBy11(n) performs the while loop at least 1 time) if i is odd, m[i]…m[2]m[1] ≡s$(mod\ 11)$. Else if i is even, m[i]…m[2]m[1]≡-s$(mod\ 11)$

proof by induction:

let p(i)= "for any positive integer n, write n in decimal system, if n has q digits, assume n = m[q]…m[2]m[1] , when the algorithm DivisibleBy11(n) performs the while loop the i th time just finished the 6th line, if i is odd, m[i]…m[2]m[1]≡s$(mod\ 11)$. Else if i is even, m[i]…m[2]m[1]≡-s$(mod\ 11)$."

Base case:

1.When i = 1, s = (n rem 10) = m[1], k = (n div 10) = m[i]… m[3]m[2].

2.(m[1] rem 11) = m[1] = s (from 1)

3. $\mathrm{m}[1] \equiv \mathrm{s}(mod\ 11)$ (from 2)

4.p(1) is true.

5.When i = 2, s $\equiv 11 + \mathrm{s} \equiv$((n div 10) rem 10) – (n rem 10) $\equiv \mathrm{m}[2] - \mathrm{m}[1]$ (from lemma 3)

$-\mathrm{s} \equiv -\mathrm{m}[2] + \mathrm{m}[1] \equiv \mathrm{m}[2] * 10 + \mathrm{m}[1] \equiv \mathrm{m}[2]\mathrm{m}[1]$ (from lemma 1)

6. p(2) is true.

Constructor cases:

7.Assume for every i$<$ q, p(i) is true

9.Assume i is odd, then i+1 is even

10. when executed the i th loop, m[i]…m[2]m[1]≡s$(mod\ 11)$,
k=m[q]…m[i+1].

11. when executed the i+1 th loop, s' = (k rem 10)-s, k' = k div 10.

12.s'= (m[q]…m[i+1] rem 10) $-$s=m[i+1] $-$s

13.$-\mathrm{s}' \equiv -\mathrm{m}[\mathrm{i}+1] + \mathrm{s} \equiv$ m[i]…m[2]m[1] $-\mathrm{m}[\mathrm{i}+1]$(mod 11)

14. $-\mathrm{m}[\mathrm{i}+1] \equiv \mathrm{m}[\mathrm{i}+1] * 10 \equiv -\mathrm{m}[\mathrm{i}+1] * 100 \equiv \cdots \equiv \mathrm{m}[\mathrm{i}+1] * 10^{i} (\mathrm{mod}\ 11)$(from lemma 1 and 9)

15. $-\mathrm{s}' \equiv$ m[i+1]…m[2]m[1](from 13 and 14)

16. p(i+1) is true (from 15)

17. Assume i is even, then i+1 is odd

18. when executed the i th loop, m[i]…m[2]m[1]≡ $-$s$(mod\ 11)$,
k=m[q]…m[i+1].

< 19. when executed the i+1 th loop, s' = (k rem 10)-s, k' = k div 10.

20.s'= (m[q]…m[i+1] rem 10) $-$s=m[i+1] $-$s

21.$\mathrm{s}' \equiv \mathrm{m}[\mathrm{i}+1] - \mathrm{s} \equiv$ m[i]…m[2]m[1]+$\mathrm{m}[\mathrm{i}+1]$(mod 11)

22. $\mathrm{m}[\mathrm{i}+1] \equiv -\mathrm{m}[\mathrm{i}+1] * 10 \equiv \mathrm{m}[\mathrm{i}+1] * 100 \equiv \cdots \equiv \mathrm{m}[\mathrm{i}+1] * 10^{i} (\mathrm{mod}\ 11)$(from lemma 1 and 9)

23. $\mathrm{s}' \equiv$ m[i+1]…m[2]m[1] (mod 11) (from 21 and 22)

24. p(i+1) is true (from 23)

So, p(i)implies(i+1) from (7, 9, 16, 17, 24)

So, p(i) is true for all i $\in \mathbb{Z}^{+}$and $\mathrm{i} \leq \mathrm{q}$.

Lemma 6:

for any negative integer n, write n in decimal system, if n has q digits, assume n = -m[q]…m[2]m[1] , when the algorithm DivisibleBy11(n) performs the while loop the i th time just finished the 6th line, (notice: for every positive integer, DivisibleBy11(n) performs the while loop at least 1 time) if i is odd, -m[i]…m[2]m[1] ≡s$(mod\ 11)$. Else if i is even, -m[i]…m[2]m[1]≡-s$(mod\ 11)$

proof by induction:

let p(i)= "for any negative integer n, write n in decimal system, if n has q digits, assume n = -m[q]…m[2]m[1] , when the algorithm DivisibleBy11(n) performs the while loop the i th time just finished the 6th line, (notice: for every positive integer, DivisibleBy11(n) performs the while loop at least 1 time) if i is odd, -m[i]…m[2]m[1] ≡s$(mod\ 11)$. Else if i is even, -m[i]…m[2]m[1]≡-s$(mod\ 11)$."

Base case:

1.When i = -1, s = (n rem 10) = -m[1], k = (n div 10) = -m[i]… m[3]m[2].

2.(-m[1] rem 11) = -m[1] = s (from 1)

3. m[1] $\equiv \mathrm{s}(mod\ 11)$ (from 2)

4.p(1) is true.

5.When i = 2, s $\equiv 11 + \mathrm{s} \equiv$((n div 10) rem 10) – (n rem 10) $\equiv -\mathrm{m}[2] + \mathrm{m}[1]$ (from lemma 3)

$-\mathrm{s} \equiv \mathrm{m}[2]-\mathrm{m}[1] \equiv -\mathrm{m}[2] * 10+\mathrm{m}[1] \equiv -\mathrm{m}[2]\mathrm{m}[1]$ (from lemma 1)

6. p(2) is true.

Constructor cases:

7.Assume for every i$<$ q, p(i) is true

9.Assume i is odd, then i+1 is even

10. when executed the i th loop, -m[i]…m[2]m[1]≡s(mod 11), k=-m[q]…m[i+1].

11. when executed the i+1 th loop, s' = (k rem 10)-s, k' = k div 10.

12.s'= (-m[q]…m[i+1] rem 10) −s=-m[i+1] −s

13.$-\mathrm{s}' \equiv \mathrm{m}[\mathrm{i}+1]-\mathrm{s} \equiv$ -m[i]…m[2]m[1]+$\mathrm{m}[\mathrm{i}+1]$(mod 11)

14. $\mathrm{m}[\mathrm{i}+1] \equiv -\mathrm{m}[\mathrm{i}+1] * 10 \equiv \mathrm{m}[\mathrm{i}+1] * 100 \equiv \cdots \equiv -\mathrm{m}[\mathrm{i}+1] * 10^{i}(\bmod 11)$(from lemma 1 and 9)

15. $-\mathrm{s}' \equiv$ -m[i+1]…m[2]m[1](from 13 and 14)

16. p(i+1) is true (from 15)

17. Assume i is even, then i+1 is odd

18. when executed the i th loop, -m[i]…m[2]m[1]≡ $-$s(mod 11), k=-m[q]…m[i+1].

< 19. when executed the i+1 th loop, s' = (k rem 10)-s, k' = k div 10.

20.s'= (-m[q]…m[i+1] rem 10) −s=-m[i+1] −s

21.$\mathrm{s}' \equiv -\mathrm{m}[\mathrm{i}+1]-\mathrm{s} \equiv$ -m[i]…m[2]m[1]+$\mathrm{m}[\mathrm{i}+1]$(mod 11)

22.$-\mathrm{m}[\mathrm{i}+1] \equiv \mathrm{m}[\mathrm{i}+1] * 10 \equiv -\mathrm{m}[\mathrm{i}+1] * 100 \equiv \cdots \equiv -\mathrm{m}[\mathrm{i}+1] * 10^{i}(\bmod 11)$(from lemma 1 and 9)

23. $\mathrm{s}' \equiv$ -m[i+1]…m[2]m[1] (mod 11) (from 21 and 22)

24. p(i+1) is true (from 23)

So, p(i)implies(i+1) from (7, 9, 16, 17, 24)

So, p(i) is true for all i $\in \mathbb{Z}^{+}$and $\mathrm{i} \leq \mathrm{q}$.

Lemma 7: If the input of DivisibleBy11 is n, which is not 0, and n has q digits in decimal system, DivisibleBy11 execute the while loop q times.

Proof: write n in decimal system, we have n =m[q]…m[2]m[1]. After executed the while loop i times, we have k= m[q]…m[i+1]. So, after executed the while loop q-1 times, we have k = m[q]$\neq 0$. Then, execute the while loop one more time makes k=0 for the first time. So, DivisibleBy11 execute the while loop q times.

Proof of b:

Case 1: DivisibleBy11(0) returns true, and 0 is divisible by 11, so DivisibleBy11 is correct when input is 0.

Case 2: If the input of DivisibleBy11 is n, which is a positive number, and n has q digits in decimal system, DivisibleBy11 execute the while loop q times.

n=$\mathrm{m}[\mathrm{q}] \ldots \mathrm{m}[2]\mathrm{m}[1]$ So, from lemma 5 and lemma 7, $(\mathrm{s} \equiv \mathrm{n}(\bmod 11)) OR(-s \equiv \mathrm{n}(\bmod 11))$. s=0 if and only if n is divisible by 11. So DivisibleBy11 is correct.

Case 3: If the input of DivisibleBy11 is n, which is a negative number, and n has q digits in decimal system, DivisibleBy11 execute the while loop q times.

n=$-\mathrm{m}[\mathrm{q}] \ldots \mathrm{m}[2]\mathrm{m}[1]$ So, from lemma 6 and lemma 7, $(-s \equiv -\mathrm{n}(\mathrm{mod}\ 11)) OR (-s \equiv \mathrm{n}(\mathrm{mod}\ 11))$. s=0 if and only if n is divisible by 11. So DivisibleBy11 is correct.

So, from case 1, case 2, and case 3, we have DivisibleBy11 is totally correct.