Assignment 2 - Emirps

Rahil Agrawal z5165505
Aditya Karia z5163287

COMP2111 18s1

1 Task 1 - Specification Statement

A prime number is a positive integer that is only divisible by 1 and itself. Therefore, we can say that a number r is prime if it is not divisible by any number between 2 and $r-1$ inclusive.

Therefore, we can define a primality check function as follows:

$$isPrime(r) = \begin{cases} \text{true} & \neg\exists k \in 2..(r-1)\,(r \ \mathbf{mod}\ k = 0) \\ \text{false} & \exists k \in 2..(r-1)\,(r \ \mathbf{mod}\ k = 0) \end{cases}$$

The $reverse(r, s)$ function can be used to store the reverse of a number r in a variable called s.

Having defined a primality check function $isPrime(r, a)$ and a function to store the reverse of a number r in s, we define an *emirp*.

An *emirp* is a prime number whose reversal is also prime, but which is not a palindromic prime.

Therefore, if $EMIRP(r, n)$ states that r is the n^{th} *emirp*,where n is a positive integer, then:

$$\text{EMIRP}(r, n) = \begin{cases} \text{true} & isPrime(r) \wedge reverse(r, s) \wedge isPrime(s) \wedge r \neq s \\ \text{false} & otherwise \end{cases}$$

We want to find the n^{th} *emirp*. Having defined the limitations on n ($n > 0$) and described what it means for a r to be the n^{th}, we can specify our program with by:
$\mathbf{proc}\ \text{EMIRP}(\mathbf{value}\ n, \mathbf{result}\ r) \cdot {}_{\llcorner} n, r, x : \left[\ n > 0, \text{emirp}(r, n)\ \right]_{\lrcorner(1)}$

2 Task 2 - Derivation

proc EMIRP(**value** n, **result** r) $\cdot \llcorner n, r, x : [\ n > 0, \text{emirp}(r, n)\ ]\lrcorner_{(1)}$

(1) $\sqsubseteq$ ⟨**c-frame**⟩

$\llcorner r, x : [\ n > 0, \text{emirp}(r, n)\ ]\lrcorner_{(2)}$

(2) $\sqsubseteq$ ⟨**i-loc**⟩

$\llcorner i, r, x : [\ n > 0, \text{emirp}(r, n)\ ]\lrcorner_{(3)}$

(3) $\sqsubseteq$ ⟨**seq**⟩

$\llcorner i, x, r : [\ n > 0, i = 1 \wedge x = 13 \wedge n > 0\ ]\lrcorner_{(4)};$

$\llcorner i, x : [\ i = 1 \wedge x = 13 \wedge n > 0, \text{emirp}(r, n)\ ]\lrcorner_{(5)}$

(4) $\sqsubseteq$ ⟨**c-frame**⟩

$i, x : [\ n > 0, i = 1 \wedge x = 13 \wedge n > 0\ ]$

$\sqsubseteq$ ⟨**ass - (1)**⟩

i := 1

x := 13

(5) $\sqsubseteq$ ⟨**seq**⟩

$\llcorner i, r, x : [\ i = 1 \wedge x = 13 \wedge n > 0, Inv\ ]\lrcorner_{(6)};$

$\llcorner i, r, x : [\ Inv, Inv \wedge i = n\ ]\lrcorner_{(7)};$

$\llcorner i, r, x : [\ Inv \wedge i = n, \text{emirp}(r, n)\ ]\lrcorner_{(8)}$

(6) $\sqsubseteq$ ⟨**w-pre, c-frame - (2)**⟩

$r, x : [\ Inv[^{13}/_{r}], Inv\ ]$

$\sqsubseteq$ ⟨**ass - (3)**⟩

r := 13

(7) $\sqsubseteq$ ⟨**while**⟩

while i $\neq$ n do

$\llcorner i, r, x : [\ Inv \wedge i \neq n, Inv\ ]\lrcorner_{(9)}$

od;

(8) $\sqsubseteq$ ⟨**w-pre, c-frame - (4)** ⟩

$\llcorner r, x : [\ \text{emirp}(r, n), \text{emirp}(r, n)\ ]\lrcorner_{(10)}$

$\sqsubseteq$ ⟨**skip - (5)**⟩

skip

(9) $\sqsubseteq$ ⟨**seq**⟩

$\llcorner r, x : [\ Inv \wedge i \neq n, Inv[^{x+1}/_{x}]\ ]\lrcorner_{(10)};$

$\llcorner r, x : [\ Inv[^{x+1}/_{x}], Inv\ ]\lrcorner_{(11)}$

$(10) \sqsubseteq \quad \langle \textbf{ass - (6)} \rangle$

$\mathbf{x := x + 1}$

$(11) \sqsubseteq \quad \langle \textbf{i-loc, seq} \rangle$

$\llcorner a, i, r, x : [\ Inv[^{x+1}/_x], Inv[^{x+1}/_x] \land a = 1\ ]\lrcorner_{(12)};$

$\llcorner a, i, r, x : [\ Inv[^{x+1}/_x] \land a = 1, Inv\ ]\lrcorner_{(13)}$

$(12) \sqsubseteq \quad \langle \textbf{c-frame} \rangle$

$a, x : [\ Inv[^{x+1}/_x], Inv[^{x+1}/_x] \land a = 1\ ]$

$\sqsubseteq \quad \langle \textbf{ass - (7)} \rangle$

$\mathbf{a := 1}$

$(13) \sqsubseteq \quad \langle \textbf{seq} \rangle$

$$\llcorner a, i, r, x : \left[\begin{array}{l} Inv[^{x+1}/_x] \land a = 1, (a = 1\ \land \neg\exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0)) \\ \lor(a = 0 \land \exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0))) \end{array} \right]\lrcorner_{(14)};$$

$$\llcorner a, i, r, x : \left[\begin{array}{l} (a = 1\ \land \neg\exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0)) \\ \lor(a = 0 \land \exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0))), Inv \end{array} \right]\lrcorner_{(15)}$$

$(14) \sqsubseteq \quad \langle \textbf{w-pre - (8)} \rangle$

$a, x : [\ a = 1 \land x > 0, post(14)\ ]$

$\sqsubseteq \quad \langle \textbf{proc} \rangle$

$\mathbf{isPrime(x, a)}$

$(15) \sqsubseteq \quad \langle \textbf{if} \rangle$

$\mathbf{if\ a = 1}$

$\mathbf{then}\ \llcorner a, i, r, x : [a = 1 \land pre(15), post(15)]\lrcorner_{(16)}$

$\mathbf{else}\ \llcorner p, x : [a \neq 1 \land pre(15), post(15)]\lrcorner_{(17)}$

$\mathbf{fi}$

$(16) \sqsubseteq \quad \langle \textbf{i-loc} \rangle$

$a, i, r, s, x : [\ pre(16), post(16)\ ]$

$\sqsubseteq \quad \langle \textbf{seq} \rangle$

$\llcorner s, x : [\ pre(16), s = 0 \land x > 0\ ]\lrcorner_{(18)};$

$\llcorner a, i, r, s, x : [\ x > 0 \land s = 0, post(16)\ ]\lrcorner_{(19)}$

$(17) \sqsubseteq \quad \langle \textbf{c-frame, w-pre- (9)} \rangle$

$i, r : [\ Inv, Inv\ ]$

$\sqsubseteq \quad \langle \textbf{skip - (10)} \rangle$

$\mathbf{skip}$

$(18) \sqsubseteq \quad \langle \textbf{ass - (11)} \rangle$

$\mathbf{s := 0}$

(19) $\sqsubseteq$ $\langle$**seq**$\rangle$

$\llcorner s, x : \left[\ pre(19), s = \sum_{i=0}^{c(x)} (S_i 10^i)\ \right]_{\lrcorner(20)};$

$\llcorner a, i, r, s, x : \left[\ s = \sum_{i=0}^{c(x)} (S_i 10^i), post(19)\ \right]_{\lrcorner(21)}$

(20) $\sqsubseteq$ $\langle$**i-con, c-frame, w-pre - (12)**$\rangle$

$\textbf{con } S : [10]^* \cdot s : \left[\ x = \sum_{i=0}^{c(x)} (S_i 10^{(c(x)-i)}) \wedge x > 0, s = \sum_{i=0}^{c(x)} (S_i 10^i)\ \right]$

$\sqsubseteq$ $\langle$**proc**$\rangle$

reversen(x, s)

(21) $\sqsubseteq$ $\langle$**i-loc, seq**$\rangle$

$\llcorner a, i, r, s, b, x : [\ pre(21), pre(21) \wedge b = 1\ ]_{\lrcorner(22)};$

$\llcorner a, i, r, s, b, x : [\ pre(21) \wedge b = 1, post(21)\ ]_{\lrcorner(23)}$

(22) $\sqsubseteq$ $\langle$**c-frame, ass - (13)**$\rangle$

b := 1

(23) $\sqsubseteq$ $\langle$**seq**$\rangle$

$\llcorner a, i, r, s, b, x : \left[\begin{array}{l} pre(21) \wedge b = 1, (b = 1\ \wedge \neg \exists k \in 2..(s-1)\,(s \textbf{ mod } k = 0)) \\ \vee (b = 0 \wedge \exists k \in 2..(s-1)\,(s \textbf{ mod } k = 0)) \end{array}\right]_{\lrcorner(24)};$

$\llcorner a, i, r, s, b, x : \left[\begin{array}{l} (b = 1\ \wedge \neg \exists k \in 2..(s-1)\,(s \textbf{ mod } k = 0)) \\ \vee (b = 0 \wedge \exists k \in 2..(s-1)\,(s \textbf{ mod } k = 0)), post(21) \end{array}\right]_{\lrcorner(25)}$

(24) $\sqsubseteq$ $\langle$**w-pre - (14)**$\rangle$

$a, i, r, s, b, x : \left[\begin{array}{l} s > 0 \wedge b = 1, (b = 1\ \wedge \neg \exists k \in 2..(s-1)\,(s \textbf{ mod } k = 0)) \\ \vee (b = 0 \wedge \exists k \in 2..(s-1)\,(s \textbf{ mod } k = 0)) \end{array}\right]$

$\sqsubseteq$ $\langle$**proc**$\rangle$

isPrime(s, b)

(25) $\sqsubseteq$ $\langle$**if**$\rangle$

if b = 1 $\wedge$ **s** $\neq$ **x**

then $\llcorner i, x : [b = 1 \wedge s \neq x \wedge pre(25), post(25)]_{\lrcorner(26)}$

else $\llcorner i, r, a, s, b, x : [(b \neq 1 \vee s = x) \wedge pre(25), post(25)]_{\lrcorner(27)}$

fi;

(26) $\sqsubseteq$ $\langle$**c-frame, w-pre- (15)**$\rangle$

$a, i, r, s, b, x : [\ Inv[^{i+1}/_i][^x/_r], Inv\ ]$

$\sqsubseteq$ $\langle$**ass - (16)**$\rangle$

i := i + 1

r := x

(27) $\sqsubseteq$ $\langle$**c-frame, w-pre- (17)**$\rangle$

$a, i, r, s, b, x : [\ Inv, Inv\ ]$

$\sqsubseteq$ $\langle$**skip - (18)**$\rangle$

skip

proc $\text{isPrime}(\textbf{value}\ r, \textbf{result}\ a)\cdot$

$\llcorner r, a : \left[\begin{array}{l} a = 1 \land r > 0, (a = 1\ \land \neg\exists k \in 2..(r-1)\,(r\ \textbf{mod}\ k = 0)) \\ \lor(a = 0 \land \exists k \in 2..(r-1)\,(r\ \textbf{mod}\ k = 0) \end{array} \right]_{\lrcorner(1)}$

(1) $\sqsubseteq$ $\langle$**seq, i-loc**$\rangle$

$\llcorner r, a, j : [\ a = 1 \land r > 0, a = 1 \land r > 0 \land j = 2\ ]_{\lrcorner(2)};$

$\llcorner r, a, j : \left[\begin{array}{l} a = 1 \land r > 0 \land j = 2, (a = 1\ \land \neg\exists k \in 2..(r-1)\,(r\ \textbf{mod}\ k = 0)) \\ \lor(a = 0 \land \exists k \in 2..(r-1)\,(r\ \textbf{mod}\ k = 0) \end{array} \right]_{\lrcorner(3)}$

(2) $\sqsubseteq$ $\langle$**ass - (19)**$\rangle$

$\mathbf{j := 2}$

(3) $\sqsubseteq$ $\langle$**seq**$\rangle$

$\llcorner r, a, j : [\ a = 1 \land r > 0 \land j = 2, Inv_2\ ]_{\lrcorner(4)};$

$\llcorner r, a, j : [\ Inv_2, Inv_2 \land j = r\ ]_{\lrcorner(5)};$

$\llcorner r, a, j : [\ Inv_2 \land j = r, post(3)\ ]_{\lrcorner(6)}$

(4) $\sqsubseteq$ $\langle$**w-pre - (20)**$\rangle$

$r, a, j : [\ Inv_2, Inv_2\ ]$

$\sqsubseteq$ $\langle$**skip - (21)**$\rangle$

skip

(6) $\sqsubseteq$ $\langle$**w-pre - (22)**$\rangle$

$\llcorner r, a, j : [\ post(3), post(3)\ ]_{\lrcorner(7)}$

$\sqsubseteq$ $\langle$**skip - (23)**$\rangle$

skip

(5) $\sqsubseteq$ $\langle$**while**$\rangle$

while $\mathbf{j \neq r}$ **do**

$\llcorner r, j : [\ Inv_2 \land j \neq r, Inv_2\ ]_{\lrcorner(8)}$

od;

(8) $\sqsubseteq$ $\langle$**seq**$\rangle$

$\llcorner r, j : [\ pre(8), Inv_2[^{j+1}/_j]\ ]_{\lrcorner(9)};$

$\llcorner r, j : [\ Inv_2[^{j+1}/_j], Inv_2\ ]_{\lrcorner(10)}$

$(9) \sqsubseteq \quad \langle \mathbf{if} \rangle$

$\quad \mathbf{if\ r\ mod\ j = 0}$

$\quad \mathbf{then}\ \llcorner a : [r\ \mathbf{mod}\ j = 0 \wedge pre(9), post(9)] \lrcorner_{(11)}$

$\quad \mathbf{else}\ \llcorner a : [r\ \mathbf{mod}\ j \neq 0 \wedge pre(9), post(9)] \lrcorner_{(12)}$

$\quad \mathbf{fi};$

$(10) \sqsubseteq \quad \langle \mathbf{ass\ \text{-}\ (24)} \rangle$

$\quad \mathbf{j := j + 1}$

$(11) \sqsubseteq \quad \langle \mathbf{w\text{-}pre\ \text{-}\ (25)} \rangle$

$\quad r, a, j : \left[\ Inv_2[^{j+1}/_j][^0/_a], post(11)\ \right]$

$\sqsubseteq \quad \langle \mathbf{ass\ \text{-}\ (26)} \rangle$

$\quad \mathbf{a := 0}$

$(12) \sqsubseteq \quad \langle \mathbf{w\text{-}pre\ \text{-}\ (27)} \rangle$

$\quad r, a, j : \left[\ Inv_2[^{j+1}/_j], post(11)\ \right]$

$\sqsubseteq \quad \langle \mathbf{skip\ \text{-}\ (28)} \rangle$

$\quad \mathbf{skip}$

We gather the code for the procedure body of emirp:

EMIRP($\mathbf{r}, \mathbf{n}$) :

$\quad var\ i := 1;$

$\quad var\ x := 13;$

$\quad r := 13;$

$\quad \mathbf{while}\ j \neq r\ \mathbf{do}$

$\quad\quad x := x + 1;$

$\quad\quad var\ a := 1;$

$\quad\quad isPrime(x, a);$

$\quad\quad \mathbf{if}\ a = 1\ \mathbf{then}$

$\quad\quad\quad var\ s := 0;$

$\quad\quad\quad reversen(x, s);$

$\quad\quad\quad var\ b := 1;$

$\quad\quad\quad isPrime(s, b);$

$\quad\quad\quad \mathbf{if}\ b = 1 \wedge s \neq x\ \mathbf{then}$

$\quad\quad\quad\quad i := i + 1;$

$\quad\quad\quad\quad r := x;$

$\quad \mathbf{od};$

Also, we gather the code for the procedure body of isPrime:

```
isPrime(r,j) :
    var j := 2;
    while j ≠ r do
        if (r mod j) = 0 then
            a := 0;
        j := j + 1;
    od;
```

We have derived our code. However we need to prove **some** refinements.

2.1 Implication 1: (4) $\sqsubseteq i := 1$

To prove: $i = i_0 \wedge n > 0 \Rightarrow (i = 1 \wedge x = 13 \wedge n > 0)[^1/_i][^{13}/_x]$

Proof:

$LHS = i = i_0 \wedge n > 0$

$\Rightarrow \langle$1=1 $\wedge$ 13=13 is vacuously true$\rangle$

$1 = 1 \wedge 13 = 13 \wedge i = i_0 \wedge n > 0$

$\Rightarrow \langle A \wedge B \wedge C \wedge D \Rightarrow A \wedge B \wedge C\rangle$

$1 = 1 \wedge 13 = 13 \wedge n > 0$

$\Rightarrow \langle 1 = 1 \Rightarrow (i = 1)[^1/_i], 13 = 13 \Rightarrow (x = 13)[^{13}/_x]\rangle$

$(i = 1 \wedge x = 13 \wedge n > 0)[^1/_i][^{13}/_x]$

$\Rightarrow \langle$clearly$\rangle$

RHS

2.2 Implication 2: $(6) \sqsubseteq r, x : [Inv[^{13}/_x], Inv]$

To prove w-pre we need to prove: $pre \Rightarrow pre'$

To prove: $i = 1 \land n > 0 \land x = 13 \Rightarrow Inv[^{13}/_r]$

Proof:

$LHS = i = 1 \land n > 0 \land x = 13$

$\Rightarrow \langle A \land B \land C \Rightarrow A \land B \rangle$

$i = 1 \land x = 13$

$\Rightarrow$ ⟨We know that 13 is the 1st emirp from our definition of emirp, also 13≥r in this case⟩

$i = 1 \land \text{emirp}(13, 1) \land x = 13 \land 13 \geq r$

$\Rightarrow$ ⟨This is our Inv with 13 substituted for x⟩

$Inv[^{13}/_x]$

$\Rightarrow$ ⟨clearly⟩

RHS

2.3 Implication 3: $r, x : [Inv[^{13}/_r], Inv] \sqsubseteq r := 13$

To prove: $r = r_0 \land Inv[^{13}/_r] \Rightarrow Inv[^{13}/_r]$

Proof:

$LHS = r = r_0 \land Inv[^{13}/_r]$

$\Rightarrow \langle A \land B \Rightarrow A \rangle$

$Inv[^{13}/_r]$

$\Rightarrow$ ⟨clearly⟩

RHS

2.4 Implication 4: $Inv \wedge i = n \sqsubseteq$ **emirp**(r, n)

To prove:$Inv \wedge i = n \Rightarrow \text{emirp}(r, n)$

Proof:

$LHS = Inv \wedge i = n$

$\Rightarrow$ ⟨Expanding the Invariant⟩

$\text{emirp}(r, i) \wedge x \geq r \wedge i = n$

$\Rightarrow$ ⟨Combining conjuncts⟩

$\text{emirp}(r, n) \wedge x \geq r$

$\Rightarrow \langle A \wedge B \Rightarrow A \rangle$

$\text{emirp}(r, n)$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.5 Implication 5: (10) $\sqsubseteq$ *skip*

To prove skip, we need to prove $pre \Rightarrow post[^{r_0}/_r]$

To prove: $\text{emirp}(r, n) \Rightarrow \text{emirp}(r, n)[^{r_0}/_r]$

Proof:

$LHS = \text{emirp}(r, n)$

$\Rightarrow$ ⟨Since r_0 is the value of r in the precondition, $r = r_0$ in the precondition⟩

$\text{emirp}(r_0, n)$

$\Rightarrow$ ⟨clearly⟩

RHS

2.6 Implication 6: $[Inv \wedge i \neq n, Inv[^{x+1}/_x]] \sqsubseteq x := x + 1$

To prove:$[Inv \wedge i \neq n, Inv[^{x+1}/_x]] \sqsubseteq x := x + 1$

Proof:

$LHS = [Inv \wedge i \neq n, Inv[^{x+1}/_x]]$

$\Rightarrow$ ⟨Expanding Inv and performing substitution⟩

$[\text{emirp}(r, i) \wedge x \geq r \wedge i \neq n, \text{emirp}(r, i) \wedge x + 1 \geq r]$

We know that $x \geq r \Rightarrow x + 1 \geq r$

Since we have not found the n^{th} emirp yet and x is not an emirp, we increment x

$\Rightarrow$ ⟨Therefore, our program can be refined by $x := x + 1$⟩

RHS

2.7 Implication 7: $Inv[^{x+1}/_x], Inv[^{x+1}/_x] \wedge a = 1 \sqsubseteq a := 1$

To prove: $a = a_0 \wedge Inv[^{x+1}/_x] \Rightarrow (a = 1 \wedge Inv[^{x+1}/_x])[^1/_a]$

Proof:

$LHS = a = a_0 \wedge Inv[^{x+1}/_x]$

$\Rightarrow$ ⟨1=1 is vacuously true⟩

$1 = 1 \wedge a = a_0 \wedge Inv[^{x+1}/_x]$

$\Rightarrow$ ⟨$A \wedge B \wedge C \Rightarrow A \wedge B$⟩

$1 = 1 \wedge Inv[^{x+1}/_x]$

$\Rightarrow$ ⟨$1 = 1 \Rightarrow (a = 1 \wedge Inv[^x/_{x+1}])[^1/_i]$ (Since, Inv does not involve a)⟩

$(a = 1 \wedge Inv[^{x+1}/_x])[^1/_a]$

$\Rightarrow$ ⟨clearly⟩

RHS

2.8 Implication 8:

$[Inv[^{x+1}/_x] \wedge a = 1, post(14)] \sqsubseteq [a = 1 \wedge x > 0, post(14)]$

To prove: $Inv[^{x+1}/_x] \wedge a = 1 \Rightarrow a = 1 \wedge x > 0$

Proof:

$LHS = Inv[^{x+1}/_x] \wedge a = 1$

$\Rightarrow$ ⟨Expanding Inv and performing substitution⟩

emirp$(r, n) \wedge x + 1 >= r \wedge a = 1$

$\Rightarrow$ ⟨Since x and r starts at 13 and we are incrementing x, $x > 0$ ⟩

$x > 0 \wedge a = 1$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.9 Implication 9: $[a \neq 1 \wedge pre(15), post(15)] \sqsubseteq [[Inv, Inv]]$

To prove:$a \neq 1 \wedge pre(15) \Rightarrow Inv$

Proof:

$LHS = a \neq 1 \wedge pre(15)$

$\Rightarrow$ ⟨$a \neq 1 \Rightarrow$ x is not prime $\Rightarrow$ we have not found a new Emirp⟩

emirp$(r.n)$

$\Rightarrow$ ⟨$x \geq r$ because x only ever increases and started with x=r=13⟩

emirp$(r, n) \wedge x \geq r$

$\Rightarrow$ ⟨By definition of Inv⟩

Inv

$\Rightarrow$ ⟨Clearly⟩

RHS

2.10 Implication 10: $[Inv, Inv] \sqsubseteq skip$

To prove:$Inv \Rightarrow Inv[^{r_0}/_r]$

Proof:

$LHS = Inv$

$\Rightarrow$ $\langle$Since r_0 is the value of r in the pre-condition, thus in precondition, $r_0 = r\rangle$

$Inv[^{r_0}/_r]$

$\Rightarrow$ $\langle$Clearly$\rangle$

RHS

2.11 Implication 11: $(18) \sqsubseteq s := 0$

To prove: $s = s_0 \wedge (a = 1 \ \wedge \neg\exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0))\vee$
$((a = 0 \wedge \exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0))) \Rightarrow (s = 0 \wedge x > 0)[^0/_s]$

Proof:

$LHS = s = s_0 \wedge ((a = 1 \ \wedge \neg\exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0))\vee$
$(a = 0 \wedge \exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0)))$

$\Rightarrow$ $\langle$0=0 is vacuously true and x is a positive integer$\rangle$

$0 = 0 \wedge s = s_0 \wedge ((a = 1 \ \wedge \neg\exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0))\vee$
$(a = 0 \wedge \exists k \in 2..(x-1)\,(x \textbf{ mod } k = 0)) \wedge x > 0$

$\Rightarrow$ $\langle A \wedge B \wedge C \Rightarrow A\rangle$

$0 = 0 \wedge x > 0$

$\Rightarrow$ $\langle 0 = 0 \Rightarrow (s = 0)[^0/_s] \wedge x > 0$ does not involve $s\rangle$

$(s = 0 \wedge x > 0)[^0/_s]$

$\Rightarrow$ $\langle$clearly$\rangle$

RHS

2.12 Implication 12: $[pre(19), s = \sum_{i=0}^{c(x)}(S_i 10^i)] \sqsubseteq$

$$[x = \sum_{i=0}^{c(x)}(S_i 10^{(c(x)-i)}) \wedge x > 0, s = \sum_{i=0}^{c(x)}(S_i 10^i)]$$

To prove: $s = 0 \wedge x > 0 \Rightarrow x = \sum_{i=0}^{c(x)}(S_i 10^{(c(x)-i)}) \wedge x > 0$

Proof:

$LHS = s = 0 \wedge x > 0$

$\Rightarrow \langle A \wedge B \Rightarrow A \rangle$

$x > 0$

$\Rightarrow \langle x \in N \wedge x > 0 \Rightarrow x$ can be represented as a sum of $S_i 10^{c(n)-i}\rangle$

(S_i is a digit from the sequence of digits)

$x = \sum_{i=0}^{c(x)}(S_i 10^{(c(x)-i)}) \wedge x > 0$

$\Rightarrow \langle$Clearly$\rangle$

RHS

2.13 Implication 13: $[pre(21), pre(21) \wedge b = 1] \sqsubseteq b := 1$

To prove:$b = b_0 \wedge pre(21) \Rightarrow (pre(21) \wedge b = 1)[^1/_b]$

Proof:

$LHS = b = b_0 \wedge pre(21)$

$\Rightarrow \langle$1=1 is vacously true$\rangle$

$1 = 1 \wedge b = b_0 \wedge pre(21)$

$\Rightarrow \langle A \wedge B \wedge C \Rightarrow A \wedge B \rangle$

$1 = 1 \wedge pre(21)$

$\Rightarrow \langle 1 = 1 \Rightarrow (b = 1 \wedge pre(21))[^1/_b]$ since b does not appear in $pre(21)\rangle$

$(b = 1 \wedge pre(21))[^1/_b]$

$\Rightarrow \langle$Clearly$\rangle$

RHS

2.14 Implication 14:

$$[s = \sum_{i=0}^{c(x)}(S_i 10^i) \wedge b = 1, post(24) \sqsubseteq [s > 0 \wedge b = 1, post(24)]$$

To prove:$s = \sum_{i=0}^{c(x)}(S_i 10^i) \wedge b = 1 \Rightarrow s > 0 \wedge b = 1$

Proof:

$LHS = s = \sum_{i=0}^{c(x)}(S_i 10^i) \wedge b = 1$

$\Rightarrow \langle s = \sum_{i=0}^{c(x)}(S_i 10^i) \Rightarrow s > 0 \rangle$

$s > 0 \wedge b = 1$

$\Rightarrow \langle \text{clearly} \rangle$

RHS

2.15 Implication 15: $(26) \sqsubseteq [Inv[^{i+1}/_i][^x/_r], Inv]$

To prove: $b = 1 \wedge s \neq x \wedge (a = 1 \ \wedge \neg \exists k \in 2..(x-1)\,(x \ \mathbf{mod} \ k = 0))$
$\vee (a = 0 \wedge \exists k \in 2..(x-1)\,(x \ \mathbf{mod} \ k = 0)) \Rightarrow Inv[^{i+1}/_i][^x/_r]$

Proof:

$LHS = b = 1 \wedge s \neq x \wedge (a = 1 \ \wedge \neg \exists k \in 2..(x-1)\,(x \ \mathbf{mod} \ k = 0))$
$\vee (a = 0 \wedge \exists k \in 2..(x-1)\,(x \ \mathbf{mod} \ k = 0))$

$\Rightarrow \langle A \wedge B \wedge C \Rightarrow A \wedge B \rangle$

$b = 1 \wedge s \neq x$

$\Rightarrow \langle \text{since } b = 1 \wedge s \neq x \Rightarrow x \text{ is an emirp} \Rightarrow r = x = (i+1)^{th} \text{ emirp. } \rangle$

$[\text{emirp}(r, i) \wedge x \geq r][^{i+1}/_i][^x/_r]$

$\Rightarrow \langle \text{By the definition of Inv and substitution} \rangle$

$Inv[^{i+1}/_i][^x/_r]$

$\Rightarrow \langle \text{clearly} \rangle$

RHS

2.16 Implication 16: $[Inv[^{i+1}/_i][^x/_r], Inv] \sqsubseteq i := i+1; r := x$

To prove:$i = i_0 \wedge r = r_0 \wedge Inv[^{i+1}/_i][^x/_r] \Rightarrow Inv[^{i+1}/_i][^x/_r]$

Proof:

$LHS = i = i_0 \wedge r = r_0 \wedge Inv[^{i+1}/_i][^x/_r]$

$\Rightarrow \langle A \wedge B \wedge C \Rightarrow A \rangle$

$Inv[^{i+1}/_i][^x/_r]$

$\Rightarrow \langle \text{Cleary} \rangle$

RHS

2.17 Implication 17: $(27) \sqsubseteq [Inv, Inv]$

To prove: $(b \neq 1 \vee s = x) \wedge (a = 1 \ \wedge \neg \exists k \in 2..(x-1)\,(x \ \mathbf{mod}\ k = 0))$
$\vee (a = 0 \wedge \exists k \in 2..(x-1)\,(x \ \mathbf{mod}\ k = 0)) \Rightarrow Inv$

Proof:

$LHS = (b \neq 1 \vee s = x) \wedge (a = 1 \ \wedge \neg \exists k \in 2..(x-1)\,(x \ \mathbf{mod}\ k = 0))$
$\vee (a = 0 \wedge \exists k \in 2..(x-1)\,(x \ \mathbf{mod}\ k = 0))$

$\Rightarrow \langle A \wedge B \wedge C \Rightarrow A \wedge B \rangle$

$b \neq 1 \vee s = x$

$\Rightarrow \langle$since $b \neq 1 \wedge s = x \Rightarrow x$ is not an emirp $\Rightarrow r$ is still the i^{th} emirp$\rangle$

emirp$(r, i) \wedge x \geq r$

$\Rightarrow \langle$By the defintion of Inv$\rangle$

Inv

$\Rightarrow \langle$Clearly$\rangle$

RHS

2.18 Implication 18: $[Inv, Inv] \sqsubseteq skip$

To prove:$Inv \Rightarrow Inv[^{r_0}/_r]$

Proof:

$LHS = Inv$

$\Rightarrow$ ⟨Since r_0 is the value of r in the pre-condition, thus in precondition, $r_0 = r$⟩

$Inv[^{r_0}/_r]$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.19 Implication 19: $[a = 1 \wedge r > 0, a = 1 \wedge r > 0 \wedge j = 2] \sqsubseteq j := 2$

To prove:$j = j_0 \wedge a = 1 \wedge r > 0 \Rightarrow (a = 1 \wedge r > 0 \wedge j = 2)[^2/_j]$

Proof:

$LHS = j = j_0 \wedge a = 1 \wedge r > 0$

$\Rightarrow$ ⟨2=2 is vacously true⟩

$j = j_0 \wedge a = 1 \wedge r > 0 \wedge 2 = 2$

$\Rightarrow$ ⟨2=2 $\Rightarrow (j = 2 \wedge a = 1 \wedge r > 0)[^2/_j]$⟩

$(j = 2 \wedge a = 1 \wedge r > 0)[^2/_j]$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.20 Implication 20: $[a = 1 \wedge r > 0 \wedge j = 2, Inv_2] \sqsubseteq Inv_2, Inv_2$

To prove:$a = 1 \wedge r > 0 \wedge j = 2 \Rightarrow Inv_2$

Proof:

$$
\begin{aligned}
& LHS = a = 1 \wedge r > 0 \wedge j = 2 \\
\Rightarrow\ & \langle A \wedge B \wedge C \Rightarrow A \rangle \\
& j = 2 \\
\Rightarrow\ & \langle \exists k \in 2..(2-1)\,(\phi) \text{ is vacously true} \rangle \\
& (a = 1\ \wedge \neg \exists k \in 2..(2-1)\,(j \textbf{ mod } k = 0) \\
& \vee (a = 0 \wedge \exists k \in 2..(2-1)\,(j \textbf{ mod } k = 0)) \wedge j = 2 \\
\Rightarrow\ & \langle \text{combining conjuncts, putting j instead of 2} \rangle \\
& (a = 1\ \wedge \neg \exists k \in 2..(j-1)\,(j \textbf{ mod } k = 0) \\
& \vee (a = 0 \wedge \exists k \in 2..(j-1)\,(j \textbf{ mod } k = 0)) \\
\Rightarrow\ & \langle \text{By definition of } Inv_2 \rangle \\
& Inv_2 \\
\Rightarrow\ & \langle \text{Clearly} \rangle \\
& RHS
\end{aligned}
$$

2.21 Implication 21: $[Inv_2, Inv_2] \sqsubseteq skip$

To prove:$Inv_2 \Rightarrow Inv_2[^{r_0}/_r]$

Proof:

$$
\begin{aligned}
& LHS = Inv_2 \\
\Rightarrow\ & \langle \text{Since } r_0 \text{ is the value of } r \text{ in the pre-condition, thus in precondition, } r_0 = r \rangle \\
& Inv_2[^{r_0}/_r] \\
\Rightarrow\ & \langle \text{Clearly} \rangle \\
& RHS
\end{aligned}
$$

2.22 Implication 22: $[Inv_2 \wedge j = r, post(13)] \sqsubseteq [post(13), post(13)]$

To prove:$Inv_2 \wedge j = r \Rightarrow post(3)$

Proof:

$LHS = (a = 1 \ \wedge \neg \exists k \in 2..(j-1)\,(j \ \mathbf{mod}\ k = 0)$

$\vee\, (a = 0 \wedge \exists k \in 2..(j-1)\,(j \ \mathbf{mod}\ k = 0)) \wedge j = r$

$\Rightarrow$ ⟨Combining conjuncts⟩

$(a = 1 \ \wedge \neg \exists k \in 2..(r-1)\,(r \ \mathbf{mod}\ k = 0)$

$\vee\, (a = 0 \wedge \exists k \in 2..(r-1)\,(r \ \mathbf{mod}\ k = 0))$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.23 Implication 23: $[post(3), post(3)] \sqsubseteq skip$

To prove:$post(3) \Rightarrow post(3)[^{r_0}/_r]$

Proof:

$LHS = post(3)$

$\Rightarrow$ ⟨Since r_0 is the value of r in the pre-condition, thus in precondition, $r_0 = r$⟩

$post(3)[^{r_0}/_r]$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.24 Implication 24: $[Inv_2[^{j+1}/_j], Inv_2] \sqsubseteq j := j + 1$

To prove:$j = j_0 \wedge Inv_2[^{j+1}/_j] \Rightarrow Inv_2[^{j+1}/_j]$

Proof:

$LHS = j = j_0 \wedge Inv_2[^{j+1}/_j]$

$\Rightarrow$ ⟨$A \wedge B \Rightarrow A$⟩

$Inv_2[^{j+1}/_j]$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.25 Implication 25: $(11) \sqsubseteq [Inv_2[^{j+1}/_j][^0/_a], post(11)]$

To prove: $r \textbf{ mod } j = 0 \land Inv_2 \land j \neq r \Rightarrow Inv_2[^{j+1}/_j][^0/_a]$

Proof:

$LHS = r \textbf{ mod } j = 0 \land Inv_2 \land j \neq r$

$\Rightarrow \langle A \land B \land C \Rightarrow A \land B \rangle$

$r \textbf{ mod } j = 0 \land Inv_2$

$\Rightarrow \langle r \textbf{ mod } j = 0 \Rightarrow r$ is not a prime $\Rightarrow a := 0$ and we increment j$\rangle$

$Inv_2[^{j+1}/_j][^0/_a]$

$\Rightarrow$ ⟨clearly⟩

RHS

2.26 Implication 26: $[Inv_2[^{j+1}/_j][^0/_a] \sqsubseteq post(11)$

To prove:$a = a_0 \land Inv_2[^{j+1}/_j][^0/_a] \Rightarrow post(11)$

Proof:

$LHS = a = a_0 \land Inv_2[^{j+1}/_j][^0/_a]$

$\Rightarrow$ ⟨Removing conjuncts⟩

$Inv_2[^{j+1}/_j][^0/_a]$

$\Rightarrow$ ⟨Clearly⟩

RHS

2.27 Implication 27: $(12) \sqsubseteq [Inv_2[^{j+1}/_j], post(11)]$

To prove: $r \textbf{ mod } j \neq 0 \land Inv_2 \land j \neq r \Rightarrow Inv_2[^{j+1}/_j]$

Proof:

$LHS = r \textbf{ mod } j \neq 0 \land Inv_2 \land j \neq r$

$\Rightarrow \langle A \land B \land C \Rightarrow A \land B \rangle$

$r \textbf{ mod } j \neq 0 \land Inv_2$

$\Rightarrow \langle r \textbf{ mod } j \neq 0 \Rightarrow r$ may be prime and therefore we only increment j$\rangle$

$Inv_2[^{j+1}/_j]$

$\Rightarrow \langle$clearly$\rangle$

RHS

2.28 Implication 28: $[Inv_2, post(11)] \sqsubseteq skip$

To prove:$Inv_2[^{j+1}/_j] \Rightarrow post(11)[^{r_0}/_r]$

Proof:

$LHS = Inv_2[^{j+1}/_j]$

$\Rightarrow \langle$Since r_0 is the value of r in the pre-condition, thus in precondition, $r_0 = r\rangle$

$Inv_2[^{j+1}/_j][^{r_0}/_r]$

$\Rightarrow \langle post(11) = Inv_2[^{j+1}/_j]\rangle$

RHS

3 Task 3 - C Code

```
#include <stdio.h>
#include "reverse.h"

unsigned long emirp(unsigned long n);
void isPrime(unsigned long r, int *a);

int main (int argc, char* argv[]){
        unsigned long n;
        if(scanf("%lu", &n)==1)
           printf("%lu\n",emirp(n));
```

```
}

unsigned long emirp(unsigned long n) {
    int i = 1;
    unsigned long x = 13;
    unsigned long r = 13;
    while (i != n) {
        x = x + 1;
        int a = 1;
        isPrime(x, &a);
        if (a == 1) {
            unsigned long s = 0;
            reversen(x, &s);
            int b = 1;
            isPrime(s, &b);
            if (b == 1 && s != x)
                i = i + 1;
                r = x;
        }
    }
        return r;
}

void isPrime(unsigned long r, int *a) {
    unsigned long j = 2;
    while (j != r) {
        if (r % j == 0)
            *a = 0;
        j = j + 1;
    }
}
```