# Assignment 3 - Emirps

Rahil Agrawal z5165505
Aditya Karia z5163287

COMP2111 18s1

## 1 Task 1 -Syntactic Data Type Dict

Declare dict data type here

## 2 Task 2 - Refinement to DictA

Refine to dictA here.

## 3 Task 3 - Derivation

$$
\begin{array}{ll}
& \textbf{proc } \mathit{FunctionName}(\textbf{value } n, \textbf{result } r) \cdot \llcorner n, r, x : [\ \mathit{Pre}, \mathit{Post}\ ]\lrcorner_{(1)} \\
(1) \sqsubseteq & \quad \langle \textbf{Rule} \rangle \\
& \llcorner r, x : [\ \mathit{NewPre}, \mathit{NewPost}\ ]\lrcorner_{(2)} \\
(2) \sqsubseteq & \quad \langle \textbf{seq} \rangle \\
& \llcorner i, x, r : [\ \mathit{Pre}, \mathit{Blah}\ ]\lrcorner_{(3)}; \\
& \llcorner i, x : [\ \mathit{Blah}, \mathit{Post}\ ]\lrcorner_{(4)} \\
\sqsubseteq & \quad \langle \textbf{ass - (1)} \rangle \\
& \mathbf{i := 1} \\
& \mathbf{x := 13} \\
\sqsubseteq & \quad \langle \textbf{seq} \rangle \\
& \llcorner s, x : [\ \mathit{pre}(16), s = 0 \wedge x > 0\ ]\lrcorner_{(18)}; \\
& \llcorner a, i, r, s, x : [\ x > 0 \wedge s = 0, \mathit{post}(16)\ ]\lrcorner_{(19)} \\
(5) \sqsubseteq & \quad \langle \textbf{while} \rangle \\
& \textbf{while j} \neq \textbf{r do} \\
& \qquad \llcorner r, j : [\ \mathit{Inv}_2 \wedge j \neq r, \mathit{Inv}_2\ ]\lrcorner_{(8)} \\
& \textbf{od}; \\
(9) \sqsubseteq & \quad \langle \textbf{if} \rangle \\
& \textbf{if } \text{Gaurd} \\
& \textbf{then } \llcorner a : [\text{Gaurd} \wedge \mathit{pre}(9), \mathit{post}(9)]\lrcorner_{(11)} \\
& \textbf{else } \llcorner a : [\text{Not Gaurd} \wedge \mathit{pre}(9), \mathit{post}(9)]\lrcorner_{(12)} \\
& \textbf{fi};
\end{array}
$$

We gather the code for the procedure body of blah:

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

```
var i := 1;
var x := 13;
r := 13;
while j ≠ r do
    x := x + 1;
    var a := 1;
    isPrime(x, a);
    if a = 1 then
        var s := 0;
        reversen(x, s);
        var b := 1;
        isPrime(s, b);
        if b = 1 ∧ s ≠ x then
            i := i + 1;
            r := x;
od;
```

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

$\mathbf{isPrime}(\mathbf{r}, \mathbf{j})$ :

```
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.

## 3.1 Implication 1: $[BLAH, BLAH] \sqsubseteq BLAH$

To prove:$BLAH \Rightarrow BLAH$

Proof:

$LHS = BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$RHS$

## 3.2 Implication 2: $[BLAH, BLAH] \sqsubseteq BLAH$

To prove:$BLAH \Rightarrow BLAH$

Proof:

$LHS = BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$BLAH$

$\Rightarrow$ ⟨BLAH⟩

$RHS$

# 4 Task 4 - C Code

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