Universiteit van Amsterdam

Theoretical Excercises

PGLEc MPP

February 24, 2017

Student:
Maico Timmerman
10542590

Lecturer:
Alban Ponse

Course:
Theoretische aspecten van de Programmatuur

1 Creating the formula $(x1 + 1) \cdot (x2 + 1)$

```
setZero;

x=y; succ;

x=y; succ;

//Set the x1 to 2
x1=y;

x=y; succ;

//Set the x2 to 3
x2=y;
```

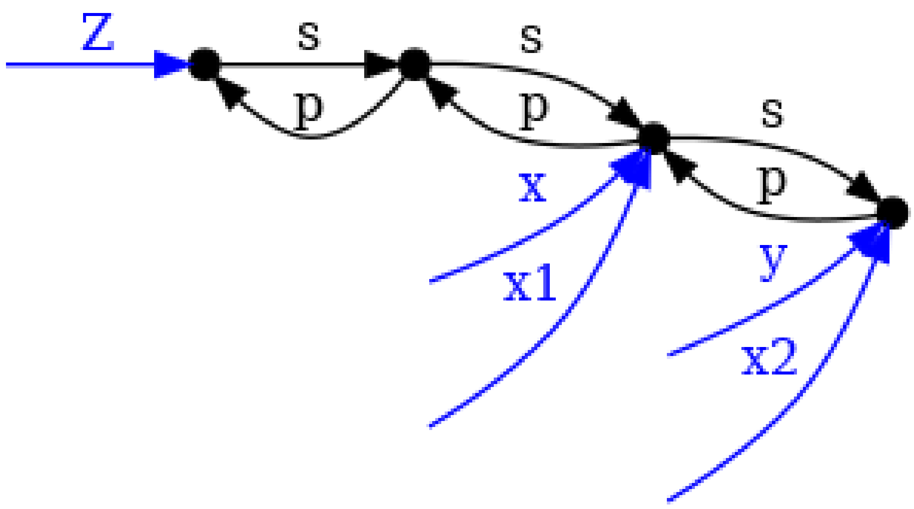

Figure 1: Initial state $x1 = 2$ and $x2 = 3$ before the algorithm.

```
// (x1 + 1)
x=x1; succ; x1=y;

// (x2 + 1)
x=x2; succ; x2=y;
```

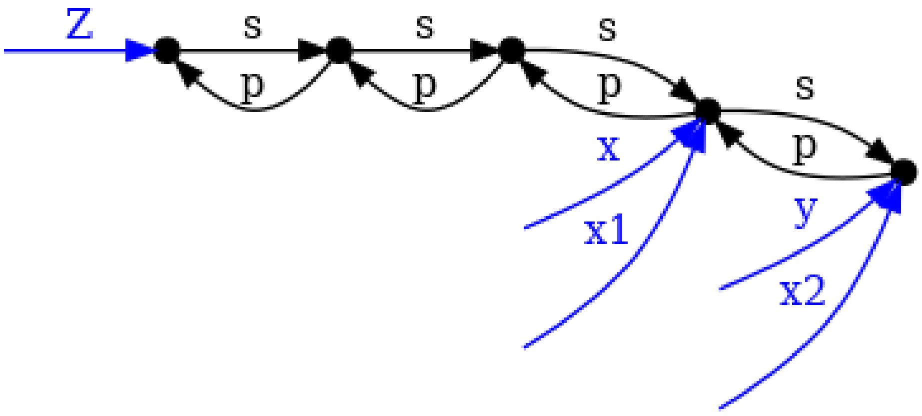

Figure 2: State after calling succ on both $x1$ and $x2$.

```
// init for multiply;
One = Z.s;
x3 = x1;
x4 = x2;
y = x2;

// Start multiply loop;
L1;
-x3==One{;
    //decrement the down-wards counter by 1;
    x3 = x3.p;

    // Set variables of ready for the addition;
    x1 = x4;
    x2 = y;

    add;

    // Loop until x3 is equal to 1;
    ##L1;
}{;
};
```

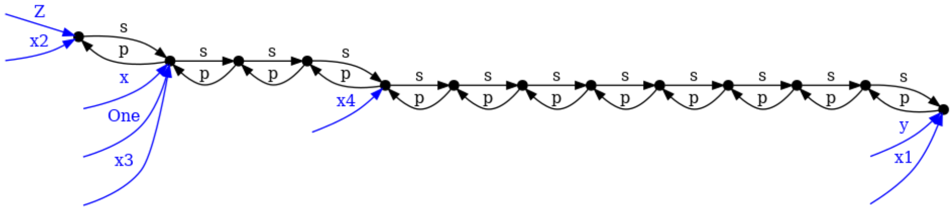

Figure 3: Final state of $(x1 + 1) \cdot (x2 + 1)$ with $x1 = 2$ and $x2 = 3$

The program will first call the successor function (`succ`) on both $x1$ and $x2$, which means can never be 0.
Implementation of the algorithm is as following:

$$\forall m, n \in \mathbb{N} : \begin{cases} m \cdot n = m \cdot 1 & = m \\ m \cdot (n+1) & = m \cdot n + m \end{cases}$$

The addition (`add`) in the program is the one provided, defined as:

$$\forall m, n \in \mathbb{N} : \begin{cases} m + n = m + 0 & = m \\ m + n & = (m+1) + (n-1) \end{cases}$$