Computability Assignment
Year 2012/13 - Number 5

Please keep this file anonymous: do not write your name inside this file.
More information about assignments at http://disi.unitn.it/∼zunino/teaching/computability/assignments
Please do not submit a file containing only the answers; edit this file, instead, filling the answer sections.

1 Question

(I am re-proposing this exercise since only a few students solved it. This exercise is rather important, since it involves a reasoning which frequently appeared in past exam questions. While we shall see more examples of these concepts in class, it would be useful to start exercising on that. If you have already sumbitted an answer, skip this and do *not* resubmit your answer please.)

Let $\mathcal{F}$ be the set of partial functions $\{f \in (\mathbb{N} \rightsquigarrow \mathbb{N}) | \forall x \in \mathbb{N}.\ f(2 \cdot x) = x\}$.

- Define two distinct partial functions f_1, f_2 which belong to $\mathcal{F}$. (I.e, provide two such examples.)
- Define two distinct partial functions g_1, g_2 which do *not* belong to $\mathcal{F}$. (I.e, provide two such examples.)
- Define a partial function $f \in \mathcal{F}$, and consider the set of its *finite* restrictions $\mathcal{G} = \{g \in (\mathbb{N} \rightsquigarrow \mathbb{N}) | g \subseteq f \ \wedge\ \mathsf{dom}(g) \text{ finite}\}$.
  - Define two distinct partial functions h_1, h_2 which belong to $\mathcal{G}$. (I.e, provide two such examples.)
  - Prove whether $\mathcal{F} \cap \mathcal{G} = \emptyset$.

1.1 Answer

Already done in the previous assignment.

2 Question

Consider the following function:

$$f(n) = \sum_{i=0}^{n} i^2 + i$$

Write a FOR loop implementing f, then translate it in the λ-calculus as program F_1.

Then, write a recursive Java-like function implementing f, and translate it in the λ-calculus as program F_2.

2.1 Answer

The FOR loop can be implemented in this way:

$sum = 0$
$FOR\, i = 0\, TO\, n\, DO$
$\quad sum = sum + i \cdot i + i$

By that we can define a λ-calculus program that can do this operations. I defined a support function called G and the program function called F_1. The idea is to create a pair of elements where the first is the solution of the function (our sum) while the second is the i. The support function just update the pair (so it do the real sum operation) while F_1 call it by passing a pair with initial values 0, that "initialize" the sum, and 1, for the i.

$G = \lambda p.Cons((Sum\,(Fst\,p)\,(Sum\,(Mul\,(Sndp)\,(Sndp))\,(Sndp)))\,(Succ(Sndp)))$
$F1 = \lambda n.Fst(n\, G(Cons\, \ulcorner 0 \urcorner\, \ulcorner 1 \urcorner))$

If we want to use a recursive function we can express the program as:

$int\, F2(int\, x)$
$\quad if(x == 0)$
$\quad\quad return\, 0;$
$\quad else\, return\, F2(x-1) + x \cdot x + x;$

Basically, at each step it adds the calculated value with the x to what the function returns with $x-1$. So like before we can define a support function and then the real function called $F2$:

$G = \lambda g.\lambda x.IsZero\, x\, \ulcorner 0 \urcorner\, (Sum\, (g\, g\, (Pred\, x))\, (Sum\, (Mul\, x\, x)\, x))$
$F2 = G\, G$

3 Question

Consider the following function:

$$f(n) = \begin{cases} x^2 + y & \text{if } n = \mathsf{pair}(\mathsf{inL}(x), y) \\ x + 4 \cdot y & \text{if } n = \mathsf{pair}(\mathsf{inR}(x), y) \end{cases}$$

Convince yourself that f is defined for all naturals n, i.e. it is total.

Write a λ-term implementing function f, exploiting the programs $Pair, Proj1, Proj2, InL, InR, Case, \ldots$ we saw in class (also defined in the notes).

3.1 Answer

The idea is to use n and the programs saw in class in order to do all the required operations. If we use the $Proj1$ function we can determine the value of x while with the $Proj2$ function we find the y value. Once we have access to the x value we can check if it is even or odd by using the $Case$ function. Finally we calculate the required values.

$F = \lambda n.Case\ (Proj1\ n)\ (Sum\ (Mul\ (Proj1\ n)\ (Proj1\ n))\ (Proj2\ n))\ (Sum\ (Proj1\ n)\ (Mul\ \ulcorner 4 \urcorner\ (Proj2\ n)))$

(RZ: $Proj1\ n$ is not x, but is $inL/inR(x)$)