CS350 Assignment 4

Rishabh Kumar Chaudhary

1 Answer 1

- return a will be either Just(a) or Nothing.

  1. Suppose return a is nothing. Then, $return(a) >>= f$ will be $nothing$.
  2. Suppose return a is Just(a). Then, $return(a) >>= f$ will be fa.

  Therefore, $return(a) >>= f$ will always be fa.

- 1. Suppose m is Nothing. Then $m >>= return$ will return Nothing.
  2. Suppose m is Just(x). Then $m >>= return$ will be $returnx$ which will again become Just(x).

  Therefore, $m >>= return$ will always be m.

- 1. Suppose m is nothing. Then $m >>= f$ will be nothing. Then, $Nothing >>= g$ will return Nothing. Therefore, final value will be Nothing.
  2. Suppose m is Just(a). Then $m >>= f$ will be fa. Then we do a $fa >>= g$. If we see $m >>= (\backslash x- > fx >>= g)$, it will give a as argument to the anonymous function. Therefore, it will become $fa >>= g$.

  Therefore, both are equivalent.

2 Answer 2

- return a will be [a]. $[a] >>= f$ will become (concat $ map f [a]). This is equal to (concat [f a]) where f will return a list. Let f a = xs where xs is a list. Then, (concat [f a]) becomes concat [xs] which becomes xs. Therefore, $[a] >>= f = xs = fa$.

- $m >>= return$ is equivalent to (concat $ map return m). Suppose $m = [m,_1, m_2 \dots m_n]$. We know that return $\mathrm{m_i}$ = $[\mathrm{m_i}]$. Therefore, given function becomes $(concat[[m_1], [m_2], \dots [m_n]])$ which is equal to $[m_1, m_2, \dots m_n]$. Hence, $m >>= return$ is m.

- $m >>= f >>= g$ is equal to (concat \$ map f m)>>=g. Suppose $m = [m_{,1}, m_2 \ldots m_n]$ and $fm_i = [f_i]$. The given expression becomes, (concat $[[f_1],[f_2]\ldots[f_n]]$)>>= g. Suppose the concat produces a list ln, then we get the expression $ln >>= g$. Now, $m >>= (\backslash x- > fx >>= g)$ is equal to concat \$ map $(\backslash x- > fx >>= g)$ m. This becomes concat \$ $[[f_1] >>= g, [f_2] >>= g, \ldots [f_n] >>= g]$. This is equal to concat \$ $[gf_1, gf_2, \ldots gf_n]$ which becomes concat \$ map g ln which is equal to $ln >>= g$. Hence, proved.