IN

main

const 1,2,3,4 → cons ×4 = [1,2,3,4]

const 10,20,30,40 → cons ×4 = [10,20,30,40]

callsnd sumVec

retsnd

callsnd print_final

sumVec (recursive template)

xs (input) → isNil xs → steer

ys (input) → isNil ys → steer

[]   *(x:xt)*   *(y:yt)*   *[]*

return []   head xs   tail xs   head ys   tail ys   return []

= x   = xt   = y   = yt

add (x + y)

callsnd sumVec → retsnd = *rest*

cons (x+y) : *rest*

ret sumVec

recursive callsnd

print_final

xs (input)

super #4 (GHC print)

OUT

super-instruction (GHC-compiled)   dataflow node (TALM-interpreted)