Jamie Sweeney - 213728s

Question 4

Use `n` to denote λ-term for n.

size(`n`) = 7n + 2

Question 5

- T - Not linear
- F - Not linear.
- Z - Linear.
- S - Not linear.
- pair - Linear.
- fst - Not linear.
- snd - Not linear.

Question 6

Base case:

M is a variable, x.
Then M[x := N] = N.
Since N is linear so too is M[x:= N].

Inductive step:

M is an application AB.
(AB)[x := N] = (A[x := N])(B[x := N])
By induction hypothesis A[x := N] and B[x := N] are linear
Since A[x := N] and B[x := N] are linear so too is (AB)[x := N]

Inductive step:

M is an abstraction λy.A
M[x := N] = λy.(A[x:=N])

By induction hypothesis A[x:=N] is linear if N and A are linear
Since A[x:=N] is linear so too is M[x := N]

Question 7

Base case:

M is a variable x.

size(M) = 1
Then M[x := N] = N
We have size(M[x := N]) = size(N)
And size(M) + size (N) = size(N) + 1
Since size(N) < size(N) + 1, we have size(M[x := N]) < size(M) + size(N)

Inductive step:
M is an abstraction λy.A
size(M) = size(A) + 1

M[x := N] = λy.(A[x:=N])
size(M[x := N]) = size(A[x:=N]) +1

By our induction hypothesis, size(A[x := N]) < size(A) + size(N)
So, size(A[x := N]) +1 < size(A) + size(N) + 1

Substitute LHS with size(M[x := N]) to get,
size(M[x := N]) < size(A) + size(N) + 1

Substitute with size(M) to get,
size(M[x := N]) < size(M) + size(N)

So hypothesis holds.

Inductive step:
M is an application AB
size(M) = size(A) + size(B)

M[x := N] = (A[x := N])(B[x := N])
size(M[x := N]) = size(A[x:=N]) + size(B[x:=N])

By our induction hypothesis, size(A[x := N]) < size(A) + size(N)
And size(B[x := N]) < size(B) + size(N)
So, size((A[x := N])(B[x := N])) < size(A) + size(B) + 2*size(N)

Substitute LHS with size(M[x := N]) to get,
size(M[x := N]) < size(A) + size(B) + 2*size(N)

Substitute with size(M) to get,
size(M[x := N]) < size(M) + 2*size(N)

Therefore
size(M[x := N]) < size(M) + size(N)

So hypothesis holds.

Question 8

Question 9

For any β reduction we have:
$(\lambda x.M)N \rightarrow_\beta M[x := N]$

Size(LHS) = size($(\lambda x.M)N$) = 1+size(M) + size(N)

Size(RHS) = size(M[x := N])
We know that size(M[x := N]) < size(M) + size(N)
So,
Size(RHS) < size(M) + size(N)

Since size(M) + size(N) < 1+size(M) + size(N)
size(RHS) < Size(LHS)

So for any linear M, if $M \rightarrow_\beta N$.
size(RHS) < Size(LHS)
size(N) < Size(M)

Question 10

Linear λ-calculus seems quite limited in the scope of computations that are available. From Q4 we see that only Z and pair are linear. I would expect that the terms expressible in linear λ-calculus would be those with no 'branches' in computation or conditional terms.