CIS 425 Assignment 2

Zhexin Jia

April 21, 2017

1. Example of a variable occurs both bound and free.

$$\lambda x.(\lambda y.xy)y$$

In above lambda expression, the variable y occurs both bound and free. The first y is the binding occurrence, the second y is bound because it is in the scope of λy, the third y is free because it is not in the scope of λy.

2. Give the free variables and bound variables of there terms:
note: bound variables are in **bold font** .
(a). $\lambda x.(\lambda y.\mathbf{xy})y$
Bound variable: x, y
Free variable: y.
The second x is in the scope of λx, the second y is in the scope of λy. So they are bound variables.
The third y is neither in the scope of λy or λx, it is free variable.

(b). $\lambda k.\mathbf{k}(\lambda f.h\mathbf{f})(qf)$
Bound variable: k, f
Free variable: h, q, f
The variable k is in the scope of λk, the second f is in the scope of λf, so they are bound variables.
The third f and q and h are neither in the scope of λf nor λk, so they are free variables.

3. Give the result of performing the following substitutions:
a.$[(\lambda y.xy)/x](x(\lambda x.yx))$

$= [(\lambda y.xy)/x](x(\lambda z.yz))$
$= ((\lambda y.xy)(\lambda z.yz))$
$= x(\lambda z.yz)$
The "x" in $\lambda x.yx$ is a bounded variable to λx(inside expression), so I rename it to "z" before doing the outside expression substitution.

b. $[(\lambda x.xy)/x](\lambda y.x(\lambda x.x))$
$= [(\lambda x.xy)/x](\lambda f.x(\lambda z.z))$
$= (\lambda f.(\lambda x.xy)(\lambda z.z))$
$= (\lambda f.((\lambda z.z)y))$
$= (\lambda f.y)$

4. Verify following by applying β-axiom:
a. $SKII = I$, where S is $\lambda x.\lambda y.\lambda z.(xz)(yz), K$ is $\lambda x.\lambda y.x$ and I is $\lambda x.x$
SKII $= (\lambda x.\lambda y.\lambda z.(xz)(yz))(\lambda x.\lambda y.x)II$
$= (\lambda y.\lambda z.((\lambda x.\lambda y.x)z)(yz))II$
$= (\lambda y.\lambda z.(\lambda y.z)(yz))II$
$= (\lambda y.\lambda z.(\lambda y.z)(yz))(\lambda x.x)I$
$= (\lambda z.(\lambda y.z)((\lambda x.x)z))I$
$= (\lambda z.(\lambda y.z)(z))I$
$= (\lambda z.(\lambda y.z)(z))(\lambda x.x)$
$= (\lambda y.(\lambda x.x))(\lambda x.x)$
$= (\lambda x.x)$
= I

b.$(\lambda x.xx)II = I$
$(\lambda x.xx)II = (II)I$
$= (\lambda x.xI)I$
$= II$
$= (\lambda x.x)(\lambda x.x)$
$= (\lambda x.x)$
$= I$

5. $Succ = \lambda u.\lambda x.\lambda y.x(uxy)$, show $Succ(\lambda x.\lambda y.x^n y) = (\lambda x.\lambda y.x^{n+1}y)$
$Succ(\lambda x.\lambda y.x^n y) = (\lambda u.\lambda x.\lambda y.x(uxy))(\lambda x.\lambda y.x^n y)$
$= (\lambda u.\lambda z.\lambda f.z(uzf))(\lambda x.\lambda y.x^n y)$
$= (\lambda z.\lambda f.z((\lambda x.\lambda y.x^n y)zf))$

$= (\lambda z.\lambda f.z((\lambda y.z^n y)f))$
$= (\lambda z.\lambda f.z(z^n f))$
$= (\lambda z.\lambda f.z^{n+1} f)$
$= (\lambda x.\lambda y.x^{n+1} y)$

6. Definition of f into lambda calculus: $\lambda g.g(g)$
$f(f) = (\lambda g.g(g))(\lambda g.g(g))$
$= (\lambda g.g(g))(\lambda g.g(g))$
Reducing f(f) will loop forever and never stop.

7. (a) Explain how to use map and reduce to compute the sum of first five squares, in one line.
reduce(function(x,y){return (x+y)}, map(function(x){return (x*x)}, [1,2,3,4,5]));
First use map function to create a new list contains all the squared values, then use reduce function to add them all together.

(b) Explain how to use map and reduce to count the number of possitve numbers in an array of numbers.

```
function check(x){
    if (x>0) return 1;
    return 0;
}
var array = [1,2,3,4,5,-1,-2,-3];
reduce(function(x,y){return (x+y)}, map(check, array))
```

Create a function check(x), it returns 1 if $x > 0$; otherwise returns 0. At first, run map(check, array) to get a new array contains 1 and 0 only, each 1 represent 1 positive number. Then run the reduce function with two parameters, the first parameter is the sum function and the second parameter is the new array we got from running the map function.

(c) Explain how to use map and/or reduce to "flatten" an array of arrays of numbers, such as [[1,2],[3,4],[5,6],[7,8,9]], to an array of numbers.

```
function append(array1, array2){
    return array1.concat(array2);
}
var arrayOfArrays = [[1, 2], [3, 4], [5, 6], [7, 8, 9]];
```

reduce(append, arrayOfArrays);
Create a function append(array1, array2), it use built-in JavaScript concatenation function to return the concatenation of array1 and array2. Then run the reduce function with paramters "append" and array of arraies.

8. (a) What is the value of g(f) in the first code example?
15. g(f) = f(7) = (x+y-2) = (10+7)-2 = 15;

(b) The call g(f) in the first code example causes the expression (x+y)-2 to be evaluated. What are the values of x and y that are used to produce the value you gave in part(a)?
x = 10, y = 7

(c)the function g(h) returns h(x), and x is the local variable 7, so the function g(h) always return h(7) no mattar what number we assigned to the global x; x = 7 and x = 10 are two different variables. x = 7 is the local variable, x = 10 is the global(free) variable
We are looking for the result of g(f), as explained above, g(f) always returns f(7). So 7 is the parameter of f(y) function, the bound variable y is assigned to 7.

(d)free variable x is first assigned to 5, then we re-assigned x to 10 just before we run the g(f) function, so the value of free variable x is 10 before we executing all the functions. g(f) returns f(7), f(7) returns (x+y)-2, x is free variable here, so we use 10 as the value of x.

(e) What's the value of g(f) in the second code example?
10. (x+y) - 2 = (5+y) - 2 = (5 + 7) - 2 = 10;

(f) The call g(f) in the second code example cause the expression (x+y)-2 to be evaluated. What are the values of x and y that are used to produce the value you gave in part(e)?
x = 5, y = 7;

(g) Explain how the value of y is set in the sequence of calls that occur before (x+y)-2 is evaluated.

Bottom functions are inside of the top functions, so the execuation sequences are from top to down. which menas x=5 is executed first, then f(y), then g(h), then x=10 and g(f);
After we assigned x = 5, we execute f(y) although we don't know y's value, the return value of f(y) becomes (5+y) -2, y is unknow at this time.
Then we execute the inner function g(h), it assigned 7 to h(x) function, the return value of g(h) is h(7), y is still unknow at this time.
At last function, we assigned x = 10, and execute g(f); Now we know g has the paramater f, so we replace h(7) to f(7), because y is the bound variable of f(y), so we assigned y = 7 at this step. The value of y is 7 at this time.

(h) Explain why x has the value you gave in part(f) when (x+y)-2 is evaluated.
Same as above, the sequence of execution is: first step x = 5; second step f(y); third step g(h); fourth step x= 10 and g(f).
We first assign x = 5, then execute f(y), f(y) now become (5+y)-2; because x is already known and substituted into the function.
when x = 10 is assigned and g(f) is being executed, the equation (x+y)-2 already become (5+y)-2, the x value is no longer needed for the equation. So the value of x remains 5.