Matthew Kelly
Dr. George Rudolph
CSIS618
18 March 2009
Homework 9

Question 9.1a

The final value of the array is {5,2,4}. Because the array is passed by value, the values in the original memory location are retained.

Question 9.1b

The final value of the array is {5,4,2} because the value of the references are changed when an array is passed by reference.

Question 9.1c

The final value of the array is {5,4,2}, like the pass by reference question.

Question 9.3

Fibonacci sequence in OCaml:

```
let rec fib n =
match n with
 0 -> 0
 |1 -> 1
 |_ -> (fib (n-1) + fib (n-2));;
```

Question 9.3a

With the above code, the value of the parameter each time the fib function is called is the current Fibonacci value being determined. For example, if fib(5) is being determined, the value of the parameter of the “top” stack frame, assuming a stack grows up and the top frame is the one most recently called, is 5 first then 4 when the next recursive call is done, 3 when the next recursive call is done, etc.

Question 9.3b

13 stack activation records are called with fibonacci(13), one for each function call.

Question 9.3c

Tail recursion can be used here to use less stack activation records. This can be accomplished by adding an additional function to handle the calculation and have that function count up instead of down causing the same activation to be repeatedly used.

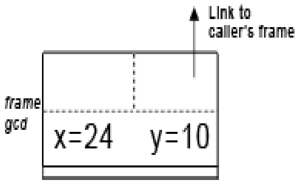
```
let rec fib2 currentNum nextNum numberLeft =
 match numberLeft with
 0 -> currentNum
 |_ -> (fib2 (nextNum) (currentNum + nextNum) (numberLeft - 1));;
```

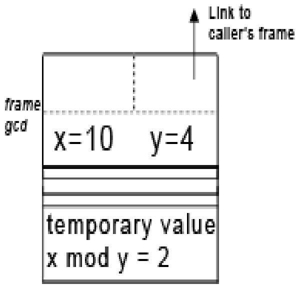
```
let rec fib n = fib2 0 1 n;;
```

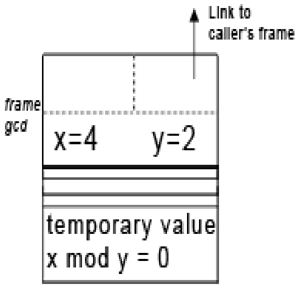
Question 9.6

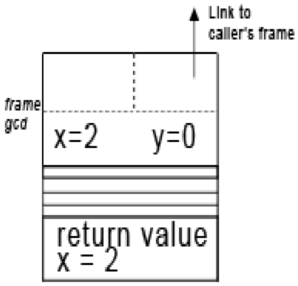
The gcd function is OCaml can be written as:

```
let rec gcd (x:int) (y:int) =
 match y with
 0 -> x
 | _ -> gcd y (x mod y);;
```


Link to
caller's frame
frame
gcd
x=24
y=10


Link to
caller's frame
frame
gcd
x=10
y=4
temporary value
x mod y = 2


Link to
caller's frame
frame
gcd
x=4
y=2
temporary value
x mod y = 0


Link to
caller's frame
frame
gcd
x=2
y=0
return value
x = 2