

---

# HW 2 – Evaluating the application of a function

CS 421 – Fall 2013

Revision 1.0

**Assigned** September 3, 2013

**Due** September 10, 2013, 19:59 pm

**Extension** 48 hours (20% penalty)

---

## 1 Change Log

1.0 Initial Release.

## 2 Turn-In Procedure

Using your favorite tool(s), you should put your solution in a file named `hw2.pdf` (the same name as this file has on the website). If you have problems generating a pdf, please seek help from the course staff. Your answers to the following questions are to be submitted electronically via the handin script as though an MP. This assignment is named `hw2`.

## 3 Objectives and Background

The purpose of this HW is to test your understanding of

- the order of evaluation of expressions in OCaml,
- the scope of variables, and the state of environments used during evaluation, and
- how an application of a function is evaluated.

Another purpose of HWs is to provide you with experience answering non-programming written questions of the kind you may experience on the midterms and final.

## 4 Problems

1. (25 pts) Below is a fragment of OCaml code, with various program points indicated by numbers with comments.

```
let x = 8;;
let y = 2;;
let mul = fun x -> fun y -> x * y;;
let mul_x = fun y -> mul x y;;
let z = mul_x y;;
(* 1 *)
let square_z = fun x -> mul z z;;
let y = square_z 0;;
```

**(★ 2 ★)**

```
let g x = x + if mul_x x = mul x x then mul_x x else -x;;
```

**(★ 3 ★)**

```
g y;;
```

You need to do:

1. (8 pts) for each of program points 1, 2, and 3, please describe the environment in effect after evaluation has reached that point,
2. (17 pts) show step by step how the application of `g y` would be evaluated.

You may assume that the evaluation begins in an empty environment, and that the environment is cumulative thereafter. The program points are supposed to indicate points at which all complete preceding declarations (including local ones) have been fully evaluated.