

# "MLing Sets"

---

## ML Programming Lab

---

## Concepts of Programming Languages

---

## CSCI 305, Spring 2018

---

## Due Date: March 09, 2018 at Midnight

---

## Table of Contents

---

- [Troubleshooting](#)
- [ML](#)
  - [Warmup](#)
    - [Lab Questions](#)
  - [Sets datatype](#)
    - [isMember function](#)
    - [list2Set function](#)
    - [Lab Questions](#)
    - [Union function](#)
    - [Intersect function](#)
    - [Lab Questions](#)
    - [Lab Questions](#)
- [Submission](#)
- [Grading](#)

## Troubleshooting

---

This lab requires an independent study of the ML language. You are encouraged to use any web tutorials and resources to learn ML beyond those in the book and provided by me (i.e. you will need to find them).

- **Given the size of the class, I will not be able to debug your code for you**
- **Please do not send panicked emails requesting I fix your bug for you.**
- **Allow yourself plenty of time, and use patience, perseverance, and the internet to debug your code.**

I will gladly answer **clarifying questions about the goals and instructions of the Lab assignment**. What this means is that if you come to my office hours or email me with questions concerning the following items or their derivatives, the provided stock answer applies. When it comes to these assignments, think of me more as your client/customer who is acquiring your product. **Before coming to me with a question regarding the assignment, ask whether you would ask a client/customer that question.**

- Can you take a look at this code and tell me what I'm doing wrong? No.
- Can you help me debug this program? No.
- Can you help me get XXX running on my computer? No.
- Can you tell me why my code isn't working? No.

## ML

---

For this lab, you will use ML. You may use any flavor of ML of your choice. I recommend the Standard New Jersey flavor for Linux and MacOS, and PolyML for Windows.

## Warmup

---

Begin by entering this function in ML. The lines that begin with a semicolon are comment lines that you will fill in.

```
fun f [] = [] (* a *)  
  | f (x::xs) = (x + 1) :: (f xs) (* b *)
```

## Lab Questions

1. Run this function as `f [3, 1, 4, 1, 5, 9]` . What output do you get?
2. What does this function `f` do?
3. Give a comment that explains the line following `(a)` .
4. Give a comment that explains the line following `(b)` .

## Sets datatype

Construct a datatype used to represent sets, called `set` , it should have two different types: `Set` and `Empty` . Where `Set` is of type `'element * 'element set` , meaning that it hold some polymorphic type `'element` and a sets of `'element` .

## isMember function

Write a function `isMember` that determines if an element `e` is part of the set, `set` . This function will return `true` if `e` is a member of the set, `set` , and `false` otherwise.

```
fun isMember e set = ; (* complete this function definition *)
```

## list2Set function

Write a function `list2Set` that convertes a list into a set ensuring that the properties of a well formed set, i.e., that is has no duplicates, are maintained. You may find it useful to make use of your `isMember` function in your `list2Set` function.

For example:

```
list2Set [1, 3, 2]; => Set (1, Set (3, Set (2, Empty)))
list2Set ["a", "b", "c"]; => Set ("a", Set ("b", Set ("c", Empty)))
list2Set []; => Empty
list2Set [6, 2, 2]; => Set (6, Set(2, Empty))
list2Set ["x", "y", "z", "x"]; => Set ("x", Set ("y", Set ("z", Empty)))
```

```
fun isSet set = ; (* complete this function definition *)
```

## Lab Questions

Your answers **must** reflect the output of your code. No credit will be given to answers if you have not submitted the respective correct function implementation.

1. What output do you get for the call: `isMember "one" ["1", "2", "3", "4"]`

2. Does your `isMember` function use head or tail recursion?
3. What output do you get for the call

```
list2Set ["it", "was", "the", "best", "of", "times,", "it", "was", "the", "worst",
```



1. Research `tail recursion`. Describe, in a few short sentences, why it can be beneficial to write tail recursing functions.

## Union function

Write a function `union` that takes the set union of set `set1` and set `set2` and returns a set representing the mathematical union of the two sets. You may use the functions you defined previously ( `list2Set` and `isMember` ), if useful, in addition to any of the common ML functions mentioned in class. Comment your function.

```
fun union set1 set2 = ; (* complete this function definition *)
```

## Intersect function

Write a function `intersect` that takes the set intersection of set `set1` and set `set2` and returns a set representing the mathematical intersection of the two sets. Comment your function.

```
fun intersect set1 set2 = ; (* Complete this function definition *)
```

## Lab Questions

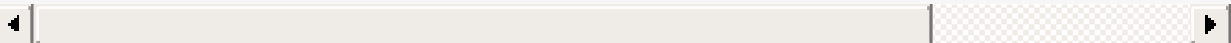
Your answers **must** reflect the output of your code for these functions. No credit will be given to answers if you have not submitted the respective correct function implementation in ML.

1. What output do you get for the call:

```
union (list2Set ["green", "eggs", "and"]) (list2Set ["ham"]);
```

1. What output do you get for the call:

```
intersect (list2Set ["stewed", "tomatoes", "and", "macaroni"]) (list2Set ["macaroni
```



## Lab Questions

The following questions are for feedback and evaluation purposes. Points are awarded for any sincere answer.

1. Name something you like about ML. Explain.
2. Name something you dislike about ML. Explain.
3. Did you enjoy this lab? Which aspects did you like and/or dislike?
4. Approximately how many hours did you spend on this lab?
5. Do you think you will use ML again? For which type(s) of project(s)?

## Submission

---

Each student will complete and submit this assignment individually. Do not consult with others. However, you are encouraged to use the internet to learn ML **but not to research the questions asked in this lab.**

Comment your program heavily. Intelligent comments and a clean, readable formatting of your code account for 20% of your grade.

Save the final version of your program as `[lastname]_[firstname].ml_lab.ml` Type your lab questions in plain text as `[lastname]_[firstname].ml_lab.questions.txt` . Include your name in the text file.

Submit your files to the ML Lab dropbox folder on BrightSpace. Do not archive your files but instead use two attachments. Submit your files before the due date as late submissions will not be accepted.

## Grading

---

Total Assignment Points: 100, Total Grade Points: 7.5

The rubric for this assignment is as follows:

- 5 points each for questions 1 - 16
  - Questions 1, 6, 8, 10, and 11 will be graded all or nothing
  - Questions 11 - 16, will be given full credit for reasonable answers
  - The remaining questions will be graded with partial credit
- 10 points for working code, with no errors
- 10 points for documented code