Practice Final
(December 10, 2017)

This exam is closed-book, notes, and technology.

Please carefully unbind the exam if it is bound.
Write only on the pages provided; scratch pages are provided at the end of the exam.
When you are done, ensure that all exam pages are present and rebind the exam with a paperclip.

Please do not open the test until the instructor says time has begun.
Please stop writing once the instructor has called time.
Failure to stop writing will result in a zero on the exam.

Remember you are here to learn.
Relax and think of this as yet another learning experience.

Good luck, have fun!

Your Name: ________________________________
Problem 1: Once

(a) Consider the following OCaml program:

```ocaml
let foo (x:int) (y:int) (z:int) : int =
  let a = x + y + z in
  if a > x && a > y && a > z then
    a
  else if a > x && a > y then
    z
  else
    x + y
```

Fill in the complete step-by-step evaluation of the expression `foo 1 (-2) 3` in the space below. Please write down the complete expression at every step of your derivation. Note that you should provide the program that is the immediate result of taking a step of evaluation from the previous, given program. When evaluating the guards of conditionals you can evaluate the entire guard to either `true` or `false` in a single step.

```ocaml
foo 1 (-2) 3

---> (* FILL ME IN (i) *)

---> if 2 > 1 && 2 > (-2) && 2 > 3 in

2

else if 2 > 1 && 2 > (-2) then

3

else

1 + (-2)

---> (* FILL ME IN (ii) *)

---> if false in

2

else if 2 > 1 && 2 > (-2) then

3

else

1 + (-2)

---> (* FILL ME IN (iii) *)

---> if true then

3

else

1 + (-2)

---> (* FILL ME IN (iii) *)
```
(b) Consider the following OCaml program:

```ocaml
let rec foo (l1:'a list) (l2:'a list) : 'a list =
  match (l1, l2) with
  | (x :: l1', y :: l2') -> x :: y :: bar l1' l2'
  | _ -> []

and bar (l1:'a list) (l2:'a list) : 'a list =
  match (l1, l2) with
  | (x :: l1', y :: l2') -> y :: x :: foo l1' l2'
```

Give the complete step-by-step evaluation of the following expression: `foo [1; 2] [3; 4]`. You may elide the branches of `match` expressions in your derivation to save writing time, e.g., `match [1; 2; 3] with ....`
Problem 2: Even Flow

(a) Assume the existence of three atomic propositions with the following truth valuations:

\[ A = \text{true} \quad B = \text{false} \quad C = \text{true} \]

What truth value does each of the given expressions in propositional logic evaluate to?

- \((A \lor B \lor C) \land (\neg A \lor \neg B \lor \neg C) \land (\neg A \lor B \lor \neg C)\)
- \(B \rightarrow C \rightarrow (A \land B)\)

(b) Show that the following expression in propositional logic:

\[ ((A \lor B) \land (A \rightarrow C) \land (B \rightarrow C)) \rightarrow C \]

is a tautology by giving a complete truth table for the expression. Your truth table should include an entry for every sub-expression of each proposition.

(b) Define the following logic predicates:

\[ P(d, y) = \text{d is a rainy day on year y} \]
\[ Q(d, p) = \text{x is p's birthday} \]

Translate the following formal proposition into an English sentence:

\[ \exists p. \forall d. Q(d, p) \rightarrow P(d, 2017). \]

(c) Translate the following informal proposition into a formal proposition using \(P\) and \(Q\) as defined above:

“Every person has a year where it is rainy on their birthday.”
Problem 3: Alive

Consider sets drawn from the universe \( U = \{ A, B, C, D, E, F \} \).

(a) Consider the following concrete sets drawn from \( U \):

\[
A = \{ A, C, E \} \\
B = \{ B, D, E \} \\
C = \{ D, F \}
\]

Write down the following description of sets using set literal notation:

\[
\overline{A \cup C} - B \\
\overline{B \cap C} \\
B \times A \\
\mathcal{P}(\mathcal{P}(C))
\]

(b) Let \( U \) be the set of students at Grinnell and consider the following predicates over \( U \)

\[
\text{friends}(x, y) = \text{true} \text{ iff } x \text{ and } y \text{ are friends} \\
\text{standing}(x, i) = \text{true} \text{ iff } x \text{ is an } i\text{-th year}
\]

Write down set comprehensions for each of the following informal set descriptions.

The set of all Grinnelleans that are third years with at least one friend at the college.

The set of all triples of Grinnelleans that are mutually friends, ordered by their standing (lowest standing first).
Problem 4: Why Go

Consider the following set $U = \{1, 2, 3, 4, 5\}$.

(a) Give a non-empty relation over $U$ that obeys each of the following properties:

- Symmetric, non-reflexive, non-transitive:
  
- Reflexive, transitive, but non-symmetric:

(b) Consider the following relation $R$ over $U$, $R = \{(1, 3), (2, 1)\}$. In the space below, provide the additional pairs necessary to make $R$ an equivalence relation.

(c) Consider the following relation $R$ over $U$, $R = \{(1, 2), (2, 3), (3, 4), (4, 5), (5, 5), (5, 1)\}$. Does the relation fulfill each of the given properties? If so, you can simply write “yes”. If not, give a one-sentence description as to why the property does not hold.

Left-total: ____________________________________________________________

Right-total: __________________________________________________________

Left-unique: __________________________________________________________

Right-unique: __________________________________________________________

Function: ____________________________________________________________
Problem 5: Black

Consider the problem of filling high school basketball teams. A single basketball team must have five people. Give combinatorial descriptions for each of the required values.

(a) Suppose you have 10 possible students to choose from. What are the total number of ways to fill a single team?

(b) Suppose you have 20 possible students to choose from. What are the total number of ways to fill two teams?

(c) Now suppose you are filling teams of mixed grades. Such a team must have at least 1 player that is in each of 9th, 10th, 11th, and 12th grades. The last slot can be filled by any player. Furthermore, suppose that you have 5 9th graders, 8 10th graders, 4 11th graders, and 6 12th graders to choose from. What are the total number of ways to fill a single team?

(d) Finally, suppose that you are filling teams based on a rating system. Suppose the pool of players contains 10 1 point players, 5 2 point players, and 3 3 point players. As a coach, you are allotted 5 points to spend on recruiting players (keeping in mind the requirement that a team must contains five players). What are the total number of ways to fill a single team?

(e) Suppose the pool of players consist of 10 point guards, 15 shooting guards, 5 small forwards, 8 power forwards, and 3 centers. Give a short description of the value described by the following combinatorial description. Be explicit when the description cares about ordering or not.

\[ \binom{10}{2} \cdot \binom{5}{1} \cdot \binom{8}{1} \cdot \binom{3}{1} \cdot \binom{1}{1}. \]
**Problem 6: Jeremy**

In the game of roulette, players bet on the outcome of a ball randomly landing in 38 distinct slots labeled with the numbers 1–36, 0, and 00. The numbers in the range 1–36 are furthermore colored as follows:

\[
\begin{align*}
1, 3, 5, 7, 9, 12, 14, 16, 19, 21, 23, 25, 27, 30, 32, 34, 36 &= \text{red} \\
2, 4, 6, 8, 10, 11, 13, 15, 17, 20, 22, 24, 26, 28, 29, 31, 33, 35 &= \text{black}
\end{align*}
\]

Give combinational descriptions (i.e., unsimplified formulae) for each of the required values.

(a) What is the probability of winning a “00” bet (where the ball lands on 00)?

(b) What is the probability of winning an “odd” bet (where the ball lands on an odd number)?

(c) What is the probability of winning either a “1st dozen” bet (where the ball lands on a number in the range 1–12) or a red bet (where the ball lands on a red number)?

(d) What is the probability of winning both an “even” bet (where the ball lands on an even number, 0 and 00 do not count as even) and a “3rd dozen” bet (where the ball lands on a 25–36)?

(e) What is the probability of winning a “black” bet (where the ball lands on a black number) provided that we would also win either an “odd” bet or a “19-to-36” bet (where the ball lands on a 19–36)?

(f) Let $X$ be a random variable describing the pay-off of a single set of bets in dollars. What is the expected value of placing both a “black” bet and a “1st dozen bet” bet if the pay-off for the black bet is 1 dollar and the pay-off for the “1st dozen bet” is 2 dollars?
Problem 7: Oceans

Give a natural deduction style proof of the following proposition in propositional logic:

$$\neg A \rightarrow (A \lor B) \rightarrow (B \rightarrow C) \rightarrow C.$$

On the opposite side of this page, you will find the natural deduction rules for propositional logic for your reference.
### Natural Deduction Rules for Propositional Logic

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Premise 2</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \land )-intro</td>
<td>( p_1 )</td>
<td>( p_2 )</td>
<td>( p_1 \land p_2 )</td>
</tr>
<tr>
<td>( \land )-elim-left</td>
<td>( p_1 \land p_2 )</td>
<td></td>
<td>( p_1 )</td>
</tr>
<tr>
<td>( \land )-elim-right</td>
<td>( p_1 \land p_2 )</td>
<td></td>
<td>( p_2 )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Premise 2</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lor )-intro-left</td>
<td>( p_1 )</td>
<td></td>
<td>( p_1 \lor p_2 )</td>
</tr>
<tr>
<td>( \lor )-intro-right</td>
<td>( p_2 )</td>
<td></td>
<td>( p_1 \lor p_2 )</td>
</tr>
<tr>
<td>( \lor )-elim</td>
<td>( p_1 \lor p_2 )</td>
<td></td>
<td>( p_1 \rightarrow q \rightarrow p_2 \rightarrow q )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Premise 2</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rightarrow )-intro</td>
<td>( [ x : p ] )</td>
<td></td>
<td>( p \rightarrow q )</td>
</tr>
<tr>
<td>( \rightarrow )-elim</td>
<td>( q \rightarrow q \rightarrow p )</td>
<td></td>
<td>( q )</td>
</tr>
<tr>
<td>Assumption</td>
<td>( [ x : p ] )</td>
<td></td>
<td>( p \rightarrow q = p \rightarrow q \land q \rightarrow p )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \top )-intro</td>
<td></td>
<td>( \top )</td>
</tr>
<tr>
<td>( \bot )-elim</td>
<td></td>
<td>( \bot )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \bot )-elim</td>
<td>( \bot )</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \top )-intro</td>
<td>( \top )</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIDDLE</td>
<td>( p \land \lnot p )</td>
<td>( \bot )</td>
</tr>
<tr>
<td>CONFLICT</td>
<td>( p \lor \lnot p )</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Premise 1</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lnot )-intro</td>
<td>( [ x : \lnot p ] )</td>
<td></td>
</tr>
<tr>
<td>( \bot )-elim</td>
<td></td>
<td>( \top )</td>
</tr>
<tr>
<td>( \bot )-elim</td>
<td></td>
<td>( p )</td>
</tr>
</tbody>
</table>
Problem 8: Garden

Formally prove the following facts about sets:

(a) $A - B = A \cap \overline{B}$

(b) $A - (A \cap (B \cup \overline{B})) = \emptyset$
Problem 9: Porch

Consider the following OCaml function:

```ocaml
let rec append (l1:'a list) (l2:'a list) = match l1 with
  | [] -> l2
  | x :: l1' -> x :: append l1' l2
```

Prove the following claims over this function:

**Claim 1 (Appending Nil).** \( \forall l. \text{append} \ [\] \ l = l \).

**Claim 2 (Append is Associative).** \( \forall l1, l2, l3. \text{append} \ l1 (\text{append} \ l2 \ l3) = \text{append} \ (\text{append} \ l1 \ l2) \ l3 \).
Problem 10: Deep

A tree is a recursively defined structure defined to be either:

- A leaf, or
- A node with two children that are two trees.

Intuitively, we can think of a tree as a list with two tails instead of just one.

The head of the tree is called its root, and we define the height of the tree to be the length of the longest path from the root to a leaf of the tree. The level of an individual node is the length of the path from the root to it.

For example, consider the following tree:

```
    A
   / \
  B   E
 / \ / \
C D F G
```

A is the root of the tree at level 0. B and E are at level 1 of the tree whereas C and D are at level 2. The overall height of the tree is 2 (i.e., the maximal level of any node in the tree).

(a) A tree is considered complete if the final level of the tree is completely filled out. The example tree above is not considered complete, but the following tree is complete.

```
    A
   / \
  B   E
 / \ / \
C D F G
```

Derive a formula for the number of nodes at level $i$ of a complete tree with height at least $i$. (Hint: to do this, write out examples of complete trees starting with height 0 and note the pattern that results.)
(b) Prove that the formula you derived in part (a) correct by induction on the height of the tree $h$. In your proof, make explicit (i) the base case and recursive case of your proof and (ii) the induction hypothesis obtained in the recursive case of the proof.

(c) Denote the number of nodes at level $i$ of a tree of height $h$ to be $C(i)$. Then the total number of nodes of a complete tree of height $h$ is given by:

$$\sum_{i=0}^{h} C(i).$$

Derive an explicit formula for this summation by a combinatorial argument based on the formula for $C(i)$ that you derived in part (a). (Hint: consider what the summation is counting and use the basic principles of counting to come up with an alternative characterization of what is being counted. Because both these formulae count the same thing, you can equate the two, completing the proof.)
(Scratch paper!)
(Another scratch paper!)