Type Systems

Lecture 10: Classical Logic and Continuation-Passing Style

Neel Krishnaswami
University of Cambridge

Proof (and Refutation) Terms

Propositions	A	$::=$	$\top \mid A \wedge B \mid \bot \mid A \vee B \mid \neg A$
True contexts	Γ	$::=$	$\cdot \mid \Gamma, x : A$
False contexts	Δ	$::=$	$\cdot \mid \Delta, u : A$
Values	e	$::=$	$\langle\rangle \mid \langle e, e'\rangle \mid \mathsf{L}\, e \mid \mathsf{R}\, e \mid \mathsf{not}(k)$
		$\mid$	$\mu u : A.\, c$
Continuations	k	$::=$	$[] \mid [k, k'] \mid \mathsf{fst}\, k \mid \mathsf{snd}\, k \mid \mathsf{not}(e)$
		$\mid$	$\mu x : A.\, c$
Contradictions	c	$::=$	$\langle e \mid_A k\rangle$

Expressions — Proof Terms

$$\frac{x : A \in \Gamma}{\Gamma; \Delta \vdash x : A \text{ true}} \textsc{Hyp}$$

(No rule for $\bot$)

$$\frac{}{\Gamma; \Delta \vdash \langle\rangle : \top \text{ true}} \top\text{P}$$

$$\frac{\Gamma; \Delta \vdash e : A \text{ true} \qquad \Gamma; \Delta \vdash e' : B \text{ true}}{\Gamma; \Delta \vdash \left\langle e, e' \right\rangle : A \wedge B \text{ true}} \wedge\text{P}$$

$$\frac{\Gamma; \Delta \vdash e : A \text{ true}}{\Gamma; \Delta \vdash \mathsf{L}\, e : A \vee B \text{ true}} \vee\text{P}_1 \qquad \frac{\Gamma; \Delta \vdash e : B \text{ true}}{\Gamma; \Delta \vdash \mathsf{R}\, e : A \vee B \text{ true}} \vee\text{P}_2$$

$$\frac{\Gamma; \Delta \vdash k : A \text{ false}}{\Gamma; \Delta \vdash \text{not}(k) : \neg A \text{ true}} \neg\text{P}$$

$$\frac{x : A \in \Delta}{\Gamma; \Delta \vdash x : A \text{ false}} \textsc{Hyp}$$

(No rule for $\top$)

$$\frac{}{\Gamma; \Delta \vdash [] : \bot \text{ false}} \bot R$$

$$\frac{\Gamma; \Delta \vdash k : A \text{ false} \qquad \Gamma; \Delta \vdash k' : B \text{ false}}{\Gamma; \Delta \vdash \left[k, k'\right] : A \vee B \text{ false}} \vee R$$

$$\frac{\Gamma; \Delta \vdash k : A \text{ false}}{\Gamma; \Delta \vdash \text{fst}\, k : A \wedge B \text{ false}} \wedge R_1 \qquad \frac{\Gamma; \Delta \vdash k : B \text{ false}}{\Gamma; \Delta \vdash \text{snd}\, k : A \wedge B \text{ false}} \wedge R_2$$

$$\frac{\Gamma; \Delta \vdash e : A \text{ true}}{\Gamma; \Delta \vdash \text{not}(e) : \neg A \text{ false}} \neg R$$

Contradictions

$$\frac{\Gamma; \Delta \vdash e : A \text{ true} \qquad \Gamma; \Delta \vdash k : A \text{ false}}{\Gamma; \Delta \vdash \langle e \mid_A k \rangle \text{ contr}} \textsc{Contr}$$

$$\frac{\Gamma; \Delta, u : A \vdash c \text{ contr}}{\Gamma; \Delta \vdash \mu u : A.\, c : A \text{ true}} \qquad \frac{\Gamma, x : A; \Delta \vdash c \text{ contr}}{\Gamma; \Delta \vdash \mu x : A.\, c : A \text{ false}}$$

Operational Semantics

$$
\begin{array}{lcl}
\langle \langle e_1, e_2 \rangle \mid_{A \wedge B} \mathsf{fst}\, k \rangle & \mapsto & \langle e_1 \mid_A k \rangle \\
\langle \langle e_1, e_2 \rangle \mid_{A \wedge B} \mathsf{snd}\, k \rangle & \mapsto & \langle e_2 \mid_B k \rangle \\
\langle \mathsf{L}\, e \mid_{A \vee B} [k_1, k_2] \rangle & \mapsto & \langle e \mid_A k_1 \rangle \\
\langle \mathsf{R}\, e \mid_{A \vee B} [k_1, k_2] \rangle & \mapsto & \langle e \mid_B k_2 \rangle \\
\langle \mathsf{not}(k) \mid_{\neg A} \mathsf{not}(e) \rangle & \mapsto & \langle e \mid_A k \rangle \\
\hline
\langle \mu u : A.\, c \mid_A k \rangle & \mapsto & [k/u]c \\
\langle e \mid_A \mu x : A.\, c \rangle & \mapsto & [e/x]c
\end{array}
$$

Type Safety?

Preservation If $\cdot; \cdot \vdash c$ contr and $c \rightsquigarrow c'$ then $\cdot; \cdot \vdash c'$ contr.

Proof By *case analysis* on evaluation derivations!

(We don't even need induction!)

Type Preservation

$$\langle\langle e_1, e_2\rangle \mid_{A \wedge B} \mathsf{fst}\, k\rangle \rightsquigarrow \langle e_1 \mid_A k\rangle$$ Assumption

$$\dfrac{\overbrace{\Gamma; \Delta \vdash \langle e_1, e_2\rangle : A \wedge B \text{ true}}^{(1)} \qquad \overbrace{\Gamma; \Delta \vdash \mathsf{fst}\, k : A \wedge B \text{ false}}^{(2)}}{\Gamma; \Delta \vdash \langle\langle e_1, e_2\rangle \mid_{A \wedge B} \mathsf{fst}\, k\rangle \text{ contr}}$$ Assumption

$$\dfrac{\overbrace{\Gamma; \Delta \vdash e_1 : A \text{ true}}^{(3)} \qquad \Gamma; \Delta \vdash e_2 : B \text{ true}}{\Gamma; \Delta \vdash \langle e_1, e_2\rangle : A \wedge B \text{ true}} \wedge P$$ Analysis of (1)

$$\dfrac{\overbrace{\Gamma; \Delta \vdash k : A \text{ false}}^{(4)}}{\Gamma; \Delta \vdash \mathsf{fst}\, k : A \wedge B \text{ false}} \wedge R_1$$ Analysis of (2)

$\cdot; \cdot \vdash \langle e_1 \mid_A k\rangle$ contr By rule on (3), (

Progress?

Progress? If $\cdot;\cdot \vdash c$ contr then $c \rightsquigarrow c'$ (or c final).

Proof:

1. A closed term c is a contradiction
2. Hopefully, there aren't any contradictions!
3. So this theorem is vacuous (assuming classical logic is consistent)

Making Progress Less Vacuous

Propositions	A	$::=$	$\ldots$	$\mid$ ans
Values	e	$::=$	$\ldots$	$\mid$ halt
Continuations	k	$::=$	$\ldots$	$\mid$ done

$$\frac{}{\Gamma; \Delta \vdash \mathsf{halt} : \mathsf{ans}\ \mathsf{true}} \qquad \frac{}{\Gamma; \Delta \vdash \mathsf{done} : \mathsf{ans}\ \mathsf{false}}$$

Progress

Progress If $\cdot; \cdot \vdash c$ contr then $c \leadsto c'$ or $c = \langle \text{halt} \mid_{\text{ans}} \text{done} \rangle$.

Proof By induction on typing derivations

The Price of Progress

$$\dfrac{\dfrac{\dfrac{\Gamma; \Delta, A \vdash \mathsf{ans}\ \ \mathsf{true} \qquad \Gamma; \Delta, A \vdash \mathsf{ans}\ \mathsf{false}}{\Gamma; \Delta, A \vdash \mathsf{contr}}}{\Gamma; \Delta \vdash A\ \ \mathsf{true}} \qquad \dfrac{\dfrac{\dfrac{\Gamma, A; \Delta \vdash \mathsf{ans}\ \ \mathsf{true} \qquad \Gamma, A; \Delta \vdash \mathsf{ans}\ \mathsf{false}}{\Gamma, A; \Delta \vdash \mathsf{contr}}}{\Gamma; \Delta \vdash A\ \mathsf{false}}}{\Gamma; \Delta \vdash \neg A\ \ \mathsf{true}}}{\Gamma; \Delta \vdash A \land \neg A\ \ \mathsf{true}}$$

- As a term:

$$\langle \mu u : A.\, \langle \mathsf{halt} \mid \mathsf{done} \rangle, \mathsf{not}(\mu x : A.\, \langle \mathsf{halt} \mid \mathsf{done} \rangle) \rangle$$

- Adding a halt configuration makes classical logic inconsistent – $A \land \neg A$ is derivable

Embedding Classical Logic into Intuitionistic Logic

- Intuitionistic logic has a clean computational reading
- Classical logic *almost* has a clean computational reading
- Q: Is there any way to equip classical logic with computational meaning?
- A: Embed classical logic *into* intuitionistic logic

The Double Negation Translation

- Fix an intuitionistic proposition p
- Define "quasi-negation" $\sim X$ as $X \to p$
- Now, we can define a translation on types as follows:

$$
\begin{array}{lcl}
(\neg A)^\circ & = & \sim A^\circ \\
\top^\circ & = & 1 \\
(A \land B)^\circ & = & A^\circ \times B^\circ \\
\bot^\circ & = & p \\
(A \lor B)^\circ & = & \sim\sim(A^\circ + B^\circ)
\end{array}
$$

In general, $\neg\neg X \to X$ is not derivable constructively. However, the following *is* derivable:

Lemma For all X, there is a function tne : $(\sim\sim\sim X) \to \sim X$

$$
\dfrac{\dfrac{\dots \qquad \dfrac{\dfrac{\dots \vdash q : X \to p \qquad \dots \vdash x : X}{k : \sim\sim\sim X, x : X, q : \sim X \vdash q\,x : p}}{k : \sim\sim\sim X, x : X \vdash \lambda q.\, q\,x : \sim\sim X}}{k : \sim\sim\sim X, x : X \vdash k\,(\lambda q.\, q\,x) : p}}{\dfrac{k : \sim\sim\sim X \vdash \lambda x.\, k\,(\lambda q.\, q\,a) : \sim X}{\cdot \vdash \underbrace{\lambda k.\, \lambda a.\, k\,(\lambda q.\, q\,a)}_{\text{tne}} : (\sim\sim\sim X) \to \sim X}}
$$

Intuitionistic Double Negation Elimination

Lemma For all A, there is a term dne_A such that

$$\cdot \vdash \mathsf{dne}_A : \sim\sim A^\circ \to A^\circ$$

Proof By induction on A.

$$\begin{array}{lcl}
\mathsf{dne}_\top & = & \lambda x.\, \langle\rangle \\
\mathsf{dne}_{A\wedge B} & = & \lambda p.\, \left\langle \begin{array}{l} \mathsf{dne}_A\,(\lambda k.\, q\,(\lambda p.\, k\,(\mathsf{fst}\, p))), \\ \mathsf{dne}_B\,(\lambda k.\, q\,(\lambda p.\, k\,(\mathsf{snd}\, p))) \end{array} \right\rangle \\
\mathsf{dne}_\bot & = & \lambda q.\, q\,(\lambda x.\, x) \\
\mathsf{dne}_{A\vee B} & = & \lambda q : \sim\sim \underbrace{\sim\sim(A^\circ \vee B^\circ)}_{(A\vee B)^\circ}.\, \mathsf{tne}\, q \\
\mathsf{dne}_{\neg A} & = & \lambda q : \sim\sim \underbrace{(\sim A^\circ)}_{(\neg A)^\circ}.\, \mathsf{tne}\, q
\end{array}$$

Double Negation Elimination for $\bot$

$$\dfrac{\dfrac{}{q : (p \to p) \to p \vdash q : (p \to p) \to p} \qquad \dfrac{\dfrac{}{q : (p \to p) \to p, x : p \vdash x : p}}{q : (p \to p) \to p \vdash \lambda x : p.\, x : p}}{\dfrac{q : (p \to p) \to p \vdash q\,(\lambda x : p.\, x) : p}{\dfrac{\cdot \vdash \lambda q : (p \to p) \to p.\, q\,(\lambda x : p.\, x) : ((p \to p) \to p) \to p}{\dfrac{\cdot \vdash \lambda q : \sim\sim p.\, q\,(\lambda x : p.\, x) : \sim\sim p \to p}{\cdot \vdash \lambda q : \sim\sim\bot^\circ.\, q\,(\lambda x : p.\, x) : \sim\sim\bot^\circ \to \bot^\circ}}}}$$

Translating Derivations

Theorem Classical terms embed into intutionistic terms:

1. If $\Gamma; \Delta \vdash e : A$ true then $\Gamma^\circ, \sim\Delta \vdash e^\circ : A^\circ$.
2. If $\Gamma; \Delta \vdash k : A$ false then $\Gamma^\circ, \sim\Delta \vdash k^\circ : \sim A^\circ$.
3. If $\Gamma; \Delta \vdash c$ contr then $\Gamma^\circ, \sim\Delta \vdash c^\circ : p$.

Proof By induction on derivations – but first, we have to define the translation!

Translating Value Contexts:

$$
\begin{array}{lcl}
(\cdot)^\circ & = & \cdot \\
(\Gamma, x : A)^\circ & = & \Gamma^\circ, x : A^\circ
\end{array}
$$

Translating Continuation Contexts:

$$
\begin{array}{lcl}
\sim(\cdot) & = & \cdot \\
\sim(\Gamma, x : A) & = & \sim\Gamma, x : \sim A^\circ
\end{array}
$$

Translating Contradictions

$$\frac{\Gamma; \Delta \vdash e : A \text{ true} \qquad \Gamma; \Delta \vdash k : A \text{ false}}{\Gamma; \Delta \vdash \langle e \mid_A k \rangle \text{ contr}} \textsc{Contr}$$

Define:

$$\langle e \mid_A k \rangle \quad = \quad k^\circ \, e^\circ$$

Translating (Most) Expressions

$$
\begin{array}{lcl}
x^\circ & = & x \\
\langle\rangle^\circ & = & \langle\rangle \\
\langle e_1, e_2\rangle^\circ & = & \langle e_1{}^\circ, e_2{}^\circ\rangle \\
(\mathsf{L}\, e)^\circ & = & \lambda k : {\sim}(A^\circ + B^\circ).\, k\, (\mathsf{L}\, e^\circ) \\
(\mathsf{R}\, e)^\circ & = & \lambda k : {\sim}(A^\circ + B^\circ).\, k\, (\mathsf{R}\, e^\circ) \\
(\mathrm{not}(k))^\circ & = & k^\circ
\end{array}
$$

Translating (Most) Continuations

$$
\begin{array}{lcl}
x^\circ & = & x \\
[]^\circ & = & \lambda x : \mathrm{ans}.\, x \\
[k_1, k_2]^\circ & = & \lambda k : \sim\sim(A^\circ + B^\circ). \\
& & \quad k\ (\lambda i : A^\circ + B^\circ. \\
& & \qquad \mathrm{case}(i, \mathrm{L}\, x \to k_1^\circ\, x, \mathrm{R}\, y \to k_2^\circ\, y)) \\
(\mathrm{fst}\, k)^\circ & = & \lambda p : (A^\circ \times B^\circ).\, k^\circ\, (\mathrm{fst}\, p) \\
(\mathrm{snd}\, k)^\circ & = & \lambda p : (A^\circ \times B^\circ).\, k^\circ\, (\mathrm{snd}\, p) \\
(\mathrm{not}(e))^\circ & = & \lambda k : \underbrace{\sim A^\circ}_{(\neg A)^\circ}.\, k\, e^\circ
\end{array}
$$

Translating Proof by Contradiction

$$\frac{\Gamma; \Delta, u : A \vdash c \text{ contr}}{\Gamma; \Delta \vdash \mu u : A.\, c : A \text{ true}}$$

1	$\Gamma^\circ, \sim(\Delta, u : A) \vdash c^\circ : p$	Assumption
2	$\Gamma^\circ, \sim\Delta, u : \sim A^\circ \vdash c^\circ : p$	Def. of $\sim$ on contexts
3	$\Gamma^\circ, \sim\Delta \vdash \lambda u : \sim A^\circ.\, c^\circ : \sim A^\circ \to p$	$\to$I
4	$\Gamma^\circ, \sim\Delta \vdash \lambda u : \sim A^\circ.\, c^\circ : \sim\sim A^\circ$	Def. of $\sim$ on types
5	$\Gamma^\circ, \sim\Delta \vdash \text{dne}_A(\lambda u : u : \sim A^\circ.\, c^\circ) : A^\circ$	$\to$E

So we define

$$(\mu u : A.\, c)^\circ = \text{dne}_A(\lambda u : \sim A^\circ.\, c^\circ)$$

Translating Refutation by Contradiction

$$\frac{\Gamma, x : A; \Delta \vdash c \text{ contr}}{\Gamma; \Delta \vdash \mu x : A.\, c : A \text{ false}}$$

1. We assume $\Gamma, x : A^\circ, \sim\Delta \vdash c^\circ : p$
2. So $\Gamma^\circ, x : A^\circ, \sim\Delta \vdash c^\circ : p$
3. So $\Gamma^\circ, \sim\Delta \vdash \lambda x : A^\circ.\, c^\circ : A^\circ \to p$
4. So $\Gamma^\circ, \sim\Delta \vdash \lambda x : A^\circ.\, c^\circ : \sim A^\circ$

So we define

$$(\mu x : A.\, c)^\circ = \lambda x : A^\circ.\, c^\circ$$

Consequences

- We now have a proof that every classical proof has a corresponding intuitionistic proof
- So classical logic is a *subsystem* of intuitionistic logic
- Because intuitionistic logic is consistent, so is classical logic
- Classical logic can inherit operational semantics from intuitionistic logic!

Many Different Embeddings

- Many different translations of classical logic were discovered many times
  - Gerhard Gentzen and Kurt Gödel
  - Andrey Kolmogorov
  - Valery Glivenko
  - Sigekatu Kuroda
- The key property is to show that $\sim\sim A^\circ \rightarrow A^\circ$ holds.

The Gödel-Gentzen Translation

Now, we can define a translation on types as follows:

$$
\begin{array}{lcl}
\neg A^\circ & = & \sim A^\circ \\
\top^\circ & = & 1 \\
(A \wedge B)^\circ & = & A^\circ \times B^\circ \\
\bot^\circ & = & p \\
(A \vee B)^\circ & = & \sim(\sim A^\circ \times \sim B^\circ)
\end{array}
$$

- This uses a different de Morgan duality for disjunction

The Kolmogorov Translation

Now, we can define another translation on types as follows:

$$
\begin{array}{lcl}
\neg A^\bullet & = & \sim\sim\sim A^\bullet \\
A \supset B^\bullet & = & \sim\sim(A^\bullet \to B^\bullet) \\
\top^\bullet & = & \sim\sim 1 \\
(A \wedge B)^\bullet & = & \sim\sim(A^\bullet \times B^\bullet) \\
\bot^\bullet & = & \sim\sim\bot \\
(A \vee B)^\bullet & = & \sim\sim(A^\bullet + B^\bullet)
\end{array}
$$

- Uniformly stick a double-negation in front of each connective.
- Deriving $\sim\sim A^\bullet \to A^\bullet$ is particularly easy:
  - The tne term will always work!

Implementing Classical Logic Axiomatically

- The proof theory of classical logic is elegant
- It is also very awkward to use:
  - Binding only arises from proof by contradiction
  - Difficult to write nested computations
  - Continuations/stacks are always explicit
- Functional languages make the stack implicit
- Can we make the continuations implicit?

The Typed Lambda Calculus with Continuations

$$
\begin{array}{llll}
\text{Types} & X & ::= & 1 \mid X \times Y \mid 0 \mid X + Y \mid X \to Y \mid \neg X \\
\text{Terms} & e & ::= & x \mid \langle\rangle \mid \langle e, e \rangle \mid \mathsf{fst}\, e \mid \mathsf{snd}\, e \\
 & & \mid & \mathsf{abort} \mid \mathsf{L}\, e \mid \mathsf{R}\, e \mid \mathsf{case}(e, \mathsf{L}\, x \to e', \mathsf{R}\, y \to e'') \\
 & & \mid & \lambda x : X.\, e \mid e\, e' \\
 & & \mid & \mathsf{throw}(e, e') \mid \mathsf{letcont}\, x.\, e \\
\text{Contexts} & \Gamma & ::= & \cdot \mid \Gamma, x : X
\end{array}
$$

Units and Pairs

$$\frac{}{\Gamma \vdash \langle\rangle : 1}\ 1\mathsf{I}$$

$$\frac{\Gamma \vdash e : X \qquad \Gamma \vdash e' : Y}{\Gamma \vdash \langle e, e' \rangle : X \times Y}\ \times\mathsf{I}$$

$$\frac{\Gamma \vdash e : X \times Y}{\Gamma \vdash \mathsf{fst}\, e : X}\ \times\mathsf{E}_1 \qquad \frac{\Gamma \vdash e : X \times Y}{\Gamma \vdash \mathsf{snd}\, e : Y}\ \times\mathsf{E}_1$$

Functions and Variables

$$\frac{X : X \in \Gamma}{\Gamma \vdash x : X}\ \textsc{Hyp} \qquad\qquad \frac{\Gamma, x : X \vdash e : Y}{\Gamma \vdash \lambda x : X.\, e : X \to Y}\ {\to}\mathrm{I}$$

$$\frac{\Gamma \vdash e : X \to Y \qquad \Gamma \vdash e' : X}{\Gamma \vdash e\, e' : Y}\ {\to}\mathrm{E}$$

Sums and the Empty Type

$$\frac{\Gamma \vdash e : X}{\Gamma \vdash \mathsf{L}\, e : X + Y} \; +\mathsf{I}_1 \qquad\qquad \frac{\Gamma \vdash e : Y}{\Gamma \vdash \mathsf{R}\, e : X + Y} \; +\mathsf{I}_2$$

$$\frac{\Gamma \vdash e : X + Y \qquad \Gamma, x : X \vdash e' : Z \qquad \Gamma, y : Y \vdash e'' : Z}{\Gamma \vdash \mathsf{case}(e, \mathsf{L}\, x \to e', \mathsf{R}\, y \to e'') : Z} \; +\mathsf{E}$$

(no intro for 0)

$$\frac{\Gamma \vdash e : 0}{\Gamma \vdash \mathsf{abort}\, e : Z} \; 0\mathsf{E}$$

Continuation Typing

$$\frac{\Gamma, u : \neg X \vdash e : X}{\Gamma \vdash \mathrm{letcont}\, u : X.\ e : X}\ \textsc{Cont} \qquad \frac{\Gamma \vdash e : \neg X \qquad \Gamma \vdash e' : X}{\Gamma \vdash \mathrm{throw}_Y(e, e') : Y}\ \textsc{Throw}$$

Examples

Double-negation elimination:

$$\begin{array}{lcl} \mathsf{dne}_X & : & \neg\neg X \to X \\ \mathsf{dne}_X & \triangleq & \lambda k : \neg\neg X.\, \mathsf{letcont}\, u : \neg X.\, \mathsf{throw}(k, u) \end{array}$$

The Excluded Middle:

$$\begin{array}{l} t : X \vee \neg X \\ t \triangleq \mathsf{letcont}\, u : \neg(X \vee \neg X). \\ \qquad \mathsf{throw}(u, \mathsf{R}\,(\mathsf{letcont}\, q : \neg\neg X. \\ \qquad\qquad\qquad \mathsf{throw}(u, \mathsf{L}\,(\mathsf{dne}_X\, q))) \end{array}$$

Continuation-Passing Style (CPS) Translation

Type translation:

$$
\begin{array}{lcl}
\neg X^\bullet & = & \sim\sim\sim X^\bullet \\
X \to Y^\bullet & = & \sim\sim(X^\bullet \to Y^\bullet) \\
1^\bullet & = & \sim\sim 1 \\
(X \times Y)^\bullet & = & \sim\sim(X^\bullet \times Y^\bullet) \\
0^\bullet & = & \sim\sim 0 \\
(X + Y)^\bullet & = & \sim\sim(X^\bullet + Y^\bullet)
\end{array}
$$

Translating contexts:

$$
\begin{array}{lcl}
(\cdot)^\bullet & = & \cdot \\
(\Gamma, x : A)^\bullet & = & \Gamma^\bullet, x : A^\bullet
\end{array}
$$

The CPS Translation Theorem

Theorem If $\Gamma \vdash e : X$ then $\Gamma^\bullet \vdash e^\bullet : X^\bullet$.

Proof: By induction on derivations – we "just" need to define $e^\bullet$.

The CPS Translation

$$
\begin{array}{lcl}
x^\bullet & = & \lambda k.\, x\, k \\
\langle\rangle^\bullet & = & \lambda k.\, k\ \langle\rangle \\
\langle e_1, e_2\rangle^\bullet & = & \lambda k.\, e_1^\bullet\, (\lambda x.\, e_2{}^\bullet\, (\lambda y.\, k\, (x, y))) \\
(\mathsf{fst}\, e)^\bullet & = & \lambda k.\, e^\bullet\, (\lambda p.\, k\, (\mathsf{fst}\, p)) \\
(\mathsf{snd}\, e)^\bullet & = & \lambda k.\, e^\bullet\, (\lambda p.\, k\, (\mathsf{snd}\, p)) \\
(\mathsf{L}\, e)^\bullet & = & \lambda k.\, e^\bullet\, (\lambda x.\, k\, (\mathsf{L}\, x)) \\
(\mathsf{R}\, e)^\bullet & = & \lambda k.\, e^\bullet\, (\lambda y.\, k\, (\mathsf{R}\, y)) \\
\mathsf{case}(e, \mathsf{L}\, x \to e_1, \mathsf{R}\, y \to e_2)^\bullet & = & \lambda k.\, e^\bullet\, (\lambda v.\, \mathsf{case}(v, \\
 & & \qquad\qquad \mathsf{L}\, x \to e_1^\bullet\, k \\
 & & \qquad\qquad \mathsf{R}\, y \to e_2^\bullet\, k) \\
(\lambda x : X.\, e)^\bullet & = & \lambda k.\, k\, (\lambda x : X^\bullet.\, e^\bullet) \\
(e_1\, e_2)^\bullet & = & \lambda k.\, e_1^\bullet\, (\lambda f.\, e_2^\bullet\, (\lambda x.\ k\, (f\, x)))
\end{array}
$$

The CPS Translation for Continuations

$$
\begin{array}{rcl}
(\mathsf{letcont}\, u : \neg X.\, e)^\bullet & = & \lambda k.\, [(\lambda q.\, q\, k)/u](e^\bullet) \\
\mathsf{throw}(e_1, e_2)^\bullet & = & \mathsf{tne}(e_1^\bullet)\, e_2^\bullet
\end{array}
$$

- The rest of the CPS translation is bookkeeping to enable these two clauses to work!

Questions

1. Give the embedding (ie, the e° and k° translations) of classical into intuitionistic logic for the Gödel-Gentzen translation. You just need to give the embeddings for sums, since that is the only case different from lecture.
2. Using the intuitionistic calculus extended with continuations, give a typed term proving *Peirce's law*:

$$((X \to Y) \to X) \to X$$