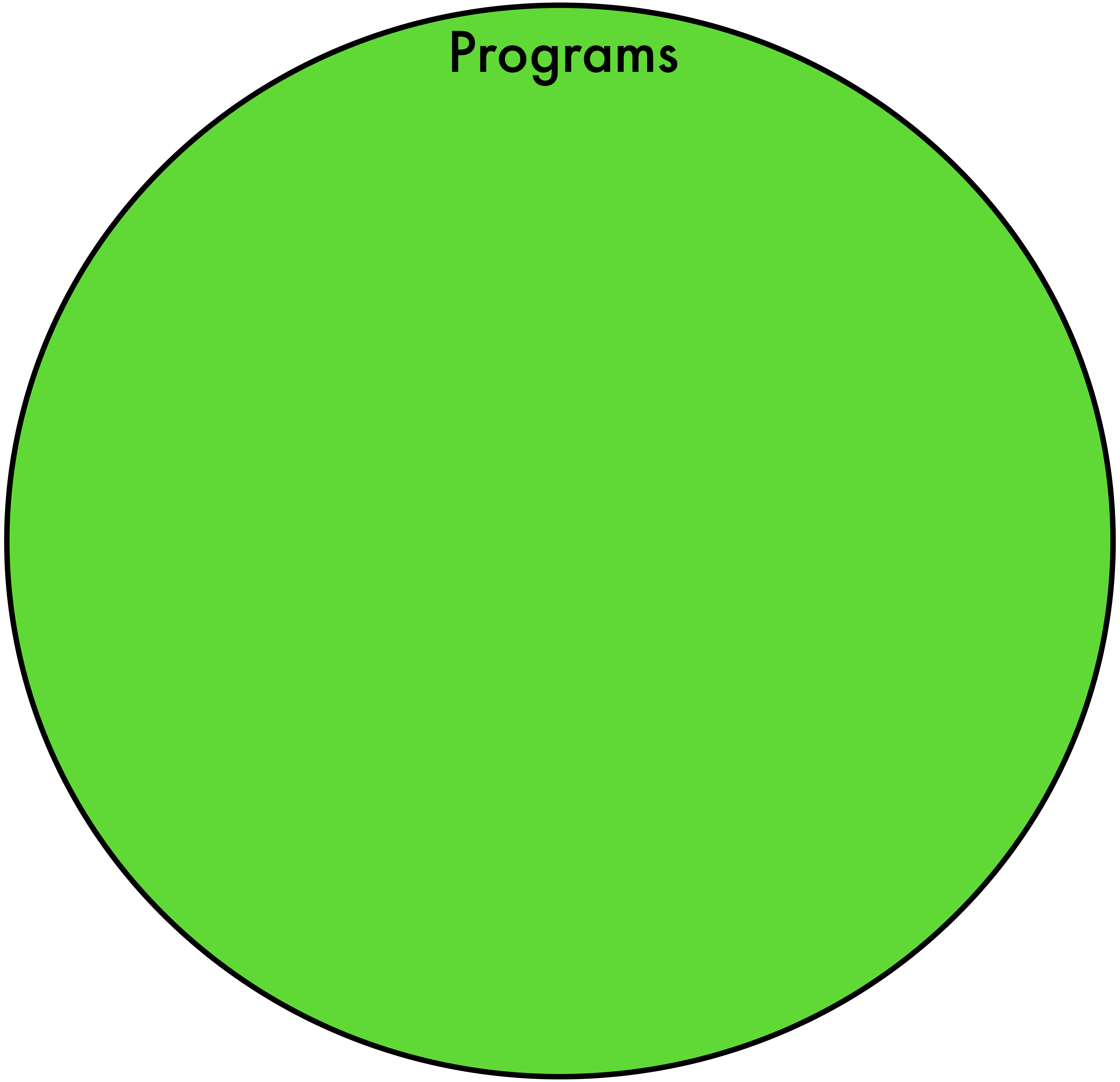
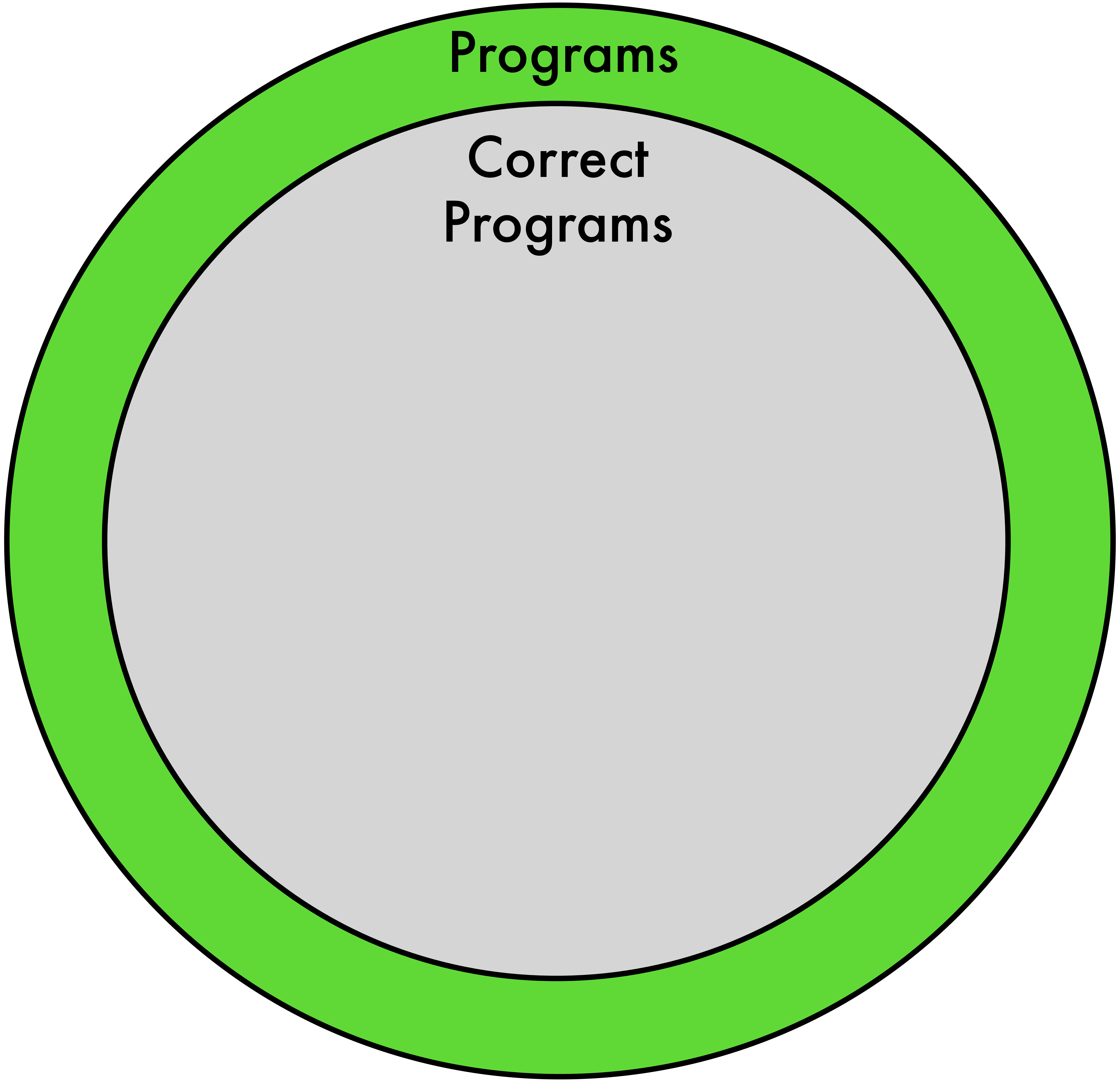
Type Safety

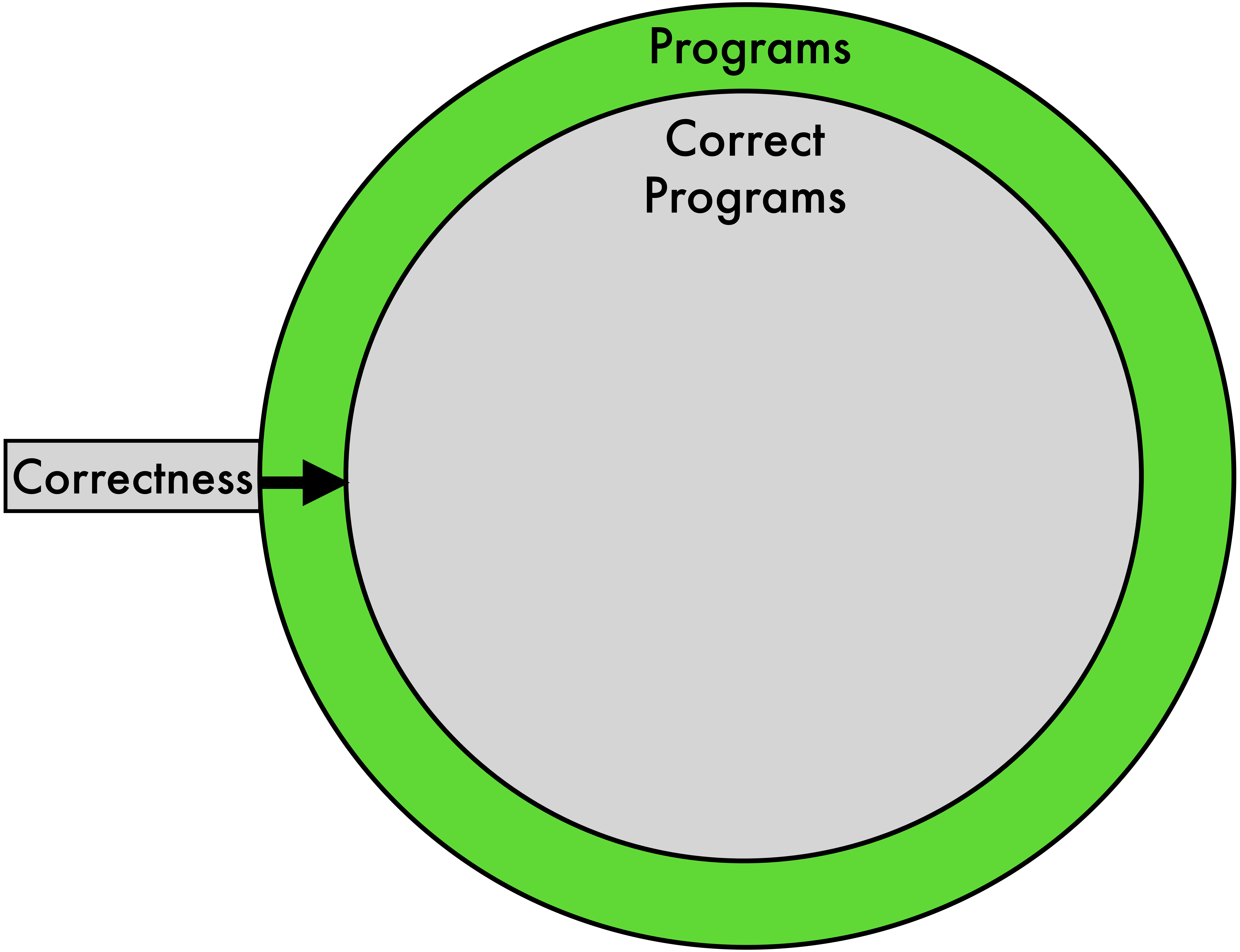
Why do we care about typing?

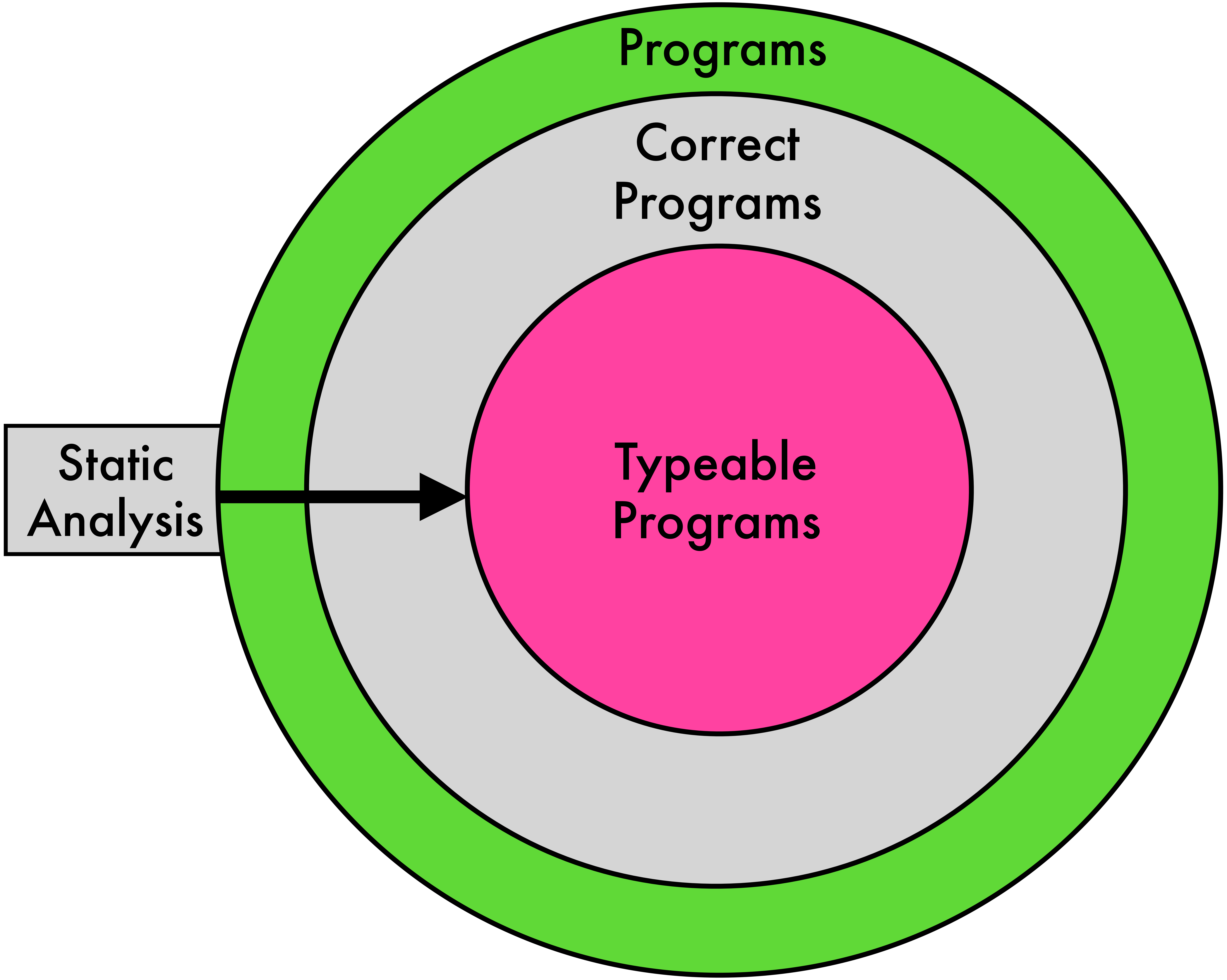
Why do we care about typing?

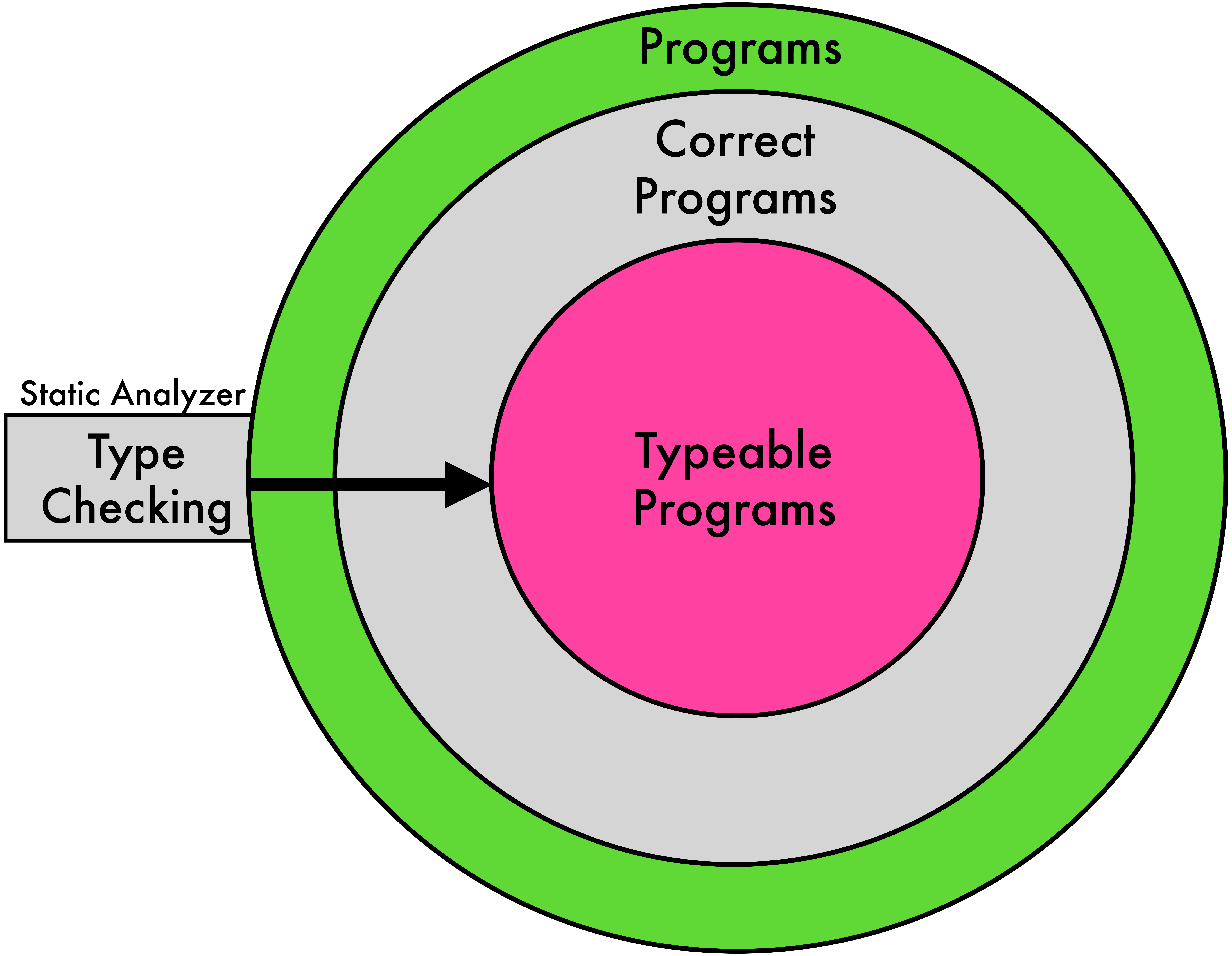
What's the point?

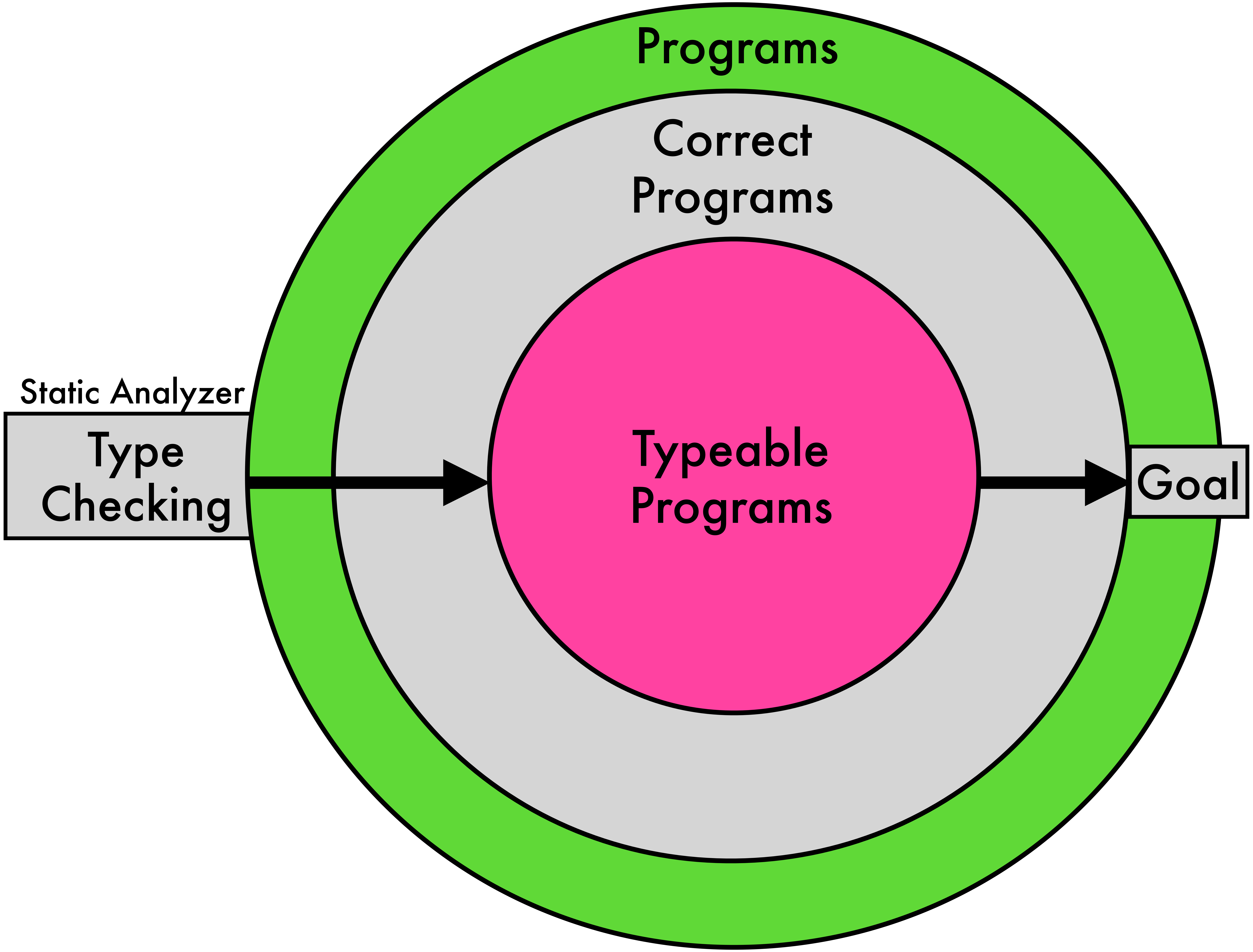

Programs


Programs
Correct
Programs


Programs
Correct
Programs
Correctness


Programs
Correct Programs
Typeable Programs
Static Analysis


Programs
Correct
Programs
Static Analyzer
Type
Checking
Typeable
Programs


Programs
Correct
Programs
Typeable
Programs
Static Analyzer
Type
Checking
Goal

When is a program correct?

Minimal Requirements of Correctness

1. The program should never get stuck during evaluation.
2. The type of a program should never change during evaluation.

Type Soundness or Type Safety

1. The program should never get stuck during evaluation.
2. The type of a program should never change during evaluation.

Type Safe Languages

C# is considered type safe with a substantial subset being proven to be.

Haskell is type safe as well.

OCaml is type safe too.

Type Safety can be hard to get

Java has been known to violate type safety.

For example, the user-definable classloaders bug in the JVM in the 90's.

How do we prove type safety?

Step 1: Progress

If $\varnothing \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

How do we prove type safety?

Step 2: Preservation

If $\varnothing \vdash e : T$ and $e \mapsto e'$, then $\varnothing \vdash e' : T$.

Step 1: Progress

$\boxed{\text{If } \varnothing \vdash e : T \text{, then either } e \text{ val or there is a } e' \text{ such that } e \mapsto e'.}$

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1 : \text{Num}$ and $\varnothing \vdash e_2 : \text{Num}$, then $\varnothing \vdash \text{plus}(e_1, e_2) : \text{Num}$.

In this case $e = \text{plus}(e_1, e_2)$.

It is never the case that $\text{plus}(e_1, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(e_1 \text{val})$ and e_2 val or $\neg(e_2 \text{val})$. Suppose the left disjuncts are true.

Step 1: Progress

$\boxed{\text{If } \varnothing \vdash e : T\text{, then either } e \text{ val or there is a } e' \text{ such that } e \mapsto e'.}$

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1 : \text{Num}$ and $\varnothing \vdash e_2 : \text{Num}$, then $\varnothing \vdash \text{plus}(e_1, e_2) : \text{Num}$.

In this case $e = \text{plus}(e_1, e_2)$.

It is never the case that $\text{plus}(e_1, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that $\underline{e_1 \text{ val}}$ or $\neg(e_1 \text{val})$ and $\underline{e_2 \text{ val}}$ or $\neg(e_2 \text{val})$. Suppose the left disjuncts are true.

Step 1: Progress

If $\varnothing \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\underline{\varnothing \vdash e_1 : \text{Num}}$ and $\underline{\varnothing \vdash e_2 : \text{Num}}$, then $\varnothing \vdash \text{plus}(e_1, e_2) : \text{Num}$.

In this case $e = \text{plus}(e_1, e_2)$.

It is never the case that $\text{plus}(e_1, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that $\underline{e_1 \text{ val}}$ or $\neg(e_1 \text{val})$ and $\underline{e_2 \text{ val}}$ or $\neg(e_2 \text{val})$. Suppose the left disjuncts are true.

Step 1a: Canonical Forms

If $\varnothing \vdash e : T$ and e val, then:

1. If $T = \text{Num}$, then $e = \text{num}[n]$ for some number n.
2. If $T = \text{Str}$, then $e = \text{str}[s]$ for some string s.

Step 1: Progress

$\boxed{\text{If } \varnothing \vdash e : T\text{, then either } e \text{ val or there is a } e' \text{ such that } e \mapsto e'.}$

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\underline{\varnothing \vdash e_1 : \text{Num}}$ and $\underline{\varnothing \vdash e_2 : \text{Num}}$, then $\varnothing \vdash \text{plus}(e_1, e_2) : \text{Num}$.

In this case $e = \text{plus}(e_1, e_2)$.

It is never the case that $\text{plus}(e_1, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that $\underline{e_1 \text{ val}}$ or $\neg(e_1 \text{val})$ and $\underline{e_2 \text{ val}}$ or $\neg(e_2 \text{val})$. Suppose the left disjuncts are true. By the canonical forms lemma there are numbers n_1 and n_2 such that $e_1 = \text{num}[n_1]$ and $e_2 = \text{num}[n_2]$. So choose $e' = \text{num}[n_1 + n_2]$, and by the rule PlusVal, $e \mapsto e'$.

Step 1: Progress

$\boxed{\text{If } \varnothing \vdash e : T\text{, then either } e \text{ val or there is a } e' \text{ such that } e \mapsto e'.}$

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1$: Num and $\varnothing \vdash e_2$: Num, then $\varnothing \vdash$ plus(e_1, e_2) : Num.

In this case $e =$ plus(e_1, e_2).

It is never the case that plus(e_1, e_2) val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(e_1$val$)$ and e_2 val or $\neg(e_2$val$)$. Suppose the first left disjunct is true and the second right disjunct is true.

Step 1: Progress

If $\varnothing \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1$: Num and $\varnothing \vdash e_2$: Num, then $\varnothing \vdash$ plus(e_1, e_2) : Num.

In this case $e =$ plus(e_1, e_2).

It is never the case that plus(e_1, e_2) val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(e_1$val$)$ and e_2 val or $\neg(e_2$val$)$. Suppose the first left disjunct is true and the second right disjunct is true. By the IH, either e_2 val or there is a e_2' such that $e_2 \mapsto e_2'$. But, we know $\neg(e_2$ val$)$, and hence, $e_2 \mapsto e_2'$. Thus, choose $e' =$ plus$(e_1; e_2')$ and we know $e \mapsto e'$ by using the rule Plus2.

Step 1: Progress

If $\emptyset \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\emptyset \vdash e : T$.

Case (Plus): Suppose if $\emptyset \vdash e_1$: Num and $\emptyset \vdash e_2$: Num, then $\emptyset \vdash \text{plus}(e_1, e_2)$: Num.

In this case $e = \text{plus}(e_1, e_2)$.

It is never the case that $\text{plus}(e_1, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(\underline{e_1 \text{val}})$ and $\underline{e_2 \text{ val}}$ or $\neg(e_2\text{val})$. Suppose the first right disjunct is true and the second left disjunct is true.

Step 1: Progress

If $\emptyset \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\emptyset \vdash e : T$.

Case (Plus): Suppose if $\dfrac{\emptyset \vdash e_1 : \text{Num} \quad \emptyset \vdash e_2 : \text{Num}}{\emptyset \vdash \text{plus}(e_1, e_2) : \text{Num}}$, then

In this case $e = \text{plus}(e_1, e_2)$.

It is never the case that $\text{plus}(e_1, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\underline{\neg(e_1 \text{val})}$ and $\underline{e_2 \text{ val}}$ or $\neg(e_2 \text{val})$. Suppose the first right disjunct is true and the second left disjunct is true. By the IH, either e_1 val or there is a e_1' such that $e_1 \mapsto e_1'$. But, we know $\neg(e_1 \text{ val})$, and hence, $e_1 \mapsto e_1'$. Thus, choose $e' = \text{plus}(e_1'; e_2)$ and we know $e \mapsto e'$ by using the rule Plus1.

Step 1: Progress

If $\varnothing \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1$: Num and $\varnothing \vdash e_2$: Num, then $\varnothing \vdash$ plus(e_1, e_2) : Num.

In this case $e =$ plus(e_1, e_2).

It is never the case that plus(e_1, e_2) val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(e_1$val$)$ and e_2 val or $\neg(e_2$val$)$. Suppose the first right disjunct is true and the second right disjunct is true. By the IH, either e_1 val or there is a e_1' such that $e_1 \mapsto e_1'$. But, we know $\neg(e_1$ val$)$, and hence, $e_1 \mapsto e_1'$. Thus, choose $e' =$ plus$(e_1'; e_2)$ and we know $e \mapsto e'$ by using the rule Plus1.

Step 1: Progress

If $\varnothing \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1 : T_1$ and $x : T_1 \vdash e_2 : T_2$, then $\varnothing \vdash \text{let}(e_1; x \,.\, e_2) : T_2$.

In this case $e = \text{let}(e_1; x \,.\, e_2)$.

It is never the case that $\text{let}(e_1; x \,.\, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(e_1 \text{val})$.

Step 1: Progress

$\boxed{\text{If } \varnothing \vdash e : T\text{, then either } e \text{ val or there is a } e' \text{ such that } e \mapsto e'.}$

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1 : T_1$ and $x : T_1 \vdash e_2 : T_2$, then $\varnothing \vdash \text{let}(e_1; x . e_2) : T_2$.

In this case $e = \text{let}(e_1; x . e_2)$.

It is never the case that $\text{let}(e_1; x . e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that $\underline{e_1 \text{ val}}$ or $\neg(e_1 \text{val})$. Suppose the left disjunct is true.

Step 1: Progress

> If $\varnothing \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1 : T_1$ and $x : T_1 \vdash e_2 : T_2$, then $\varnothing \vdash \text{let}(e_1; x \,.\, e_2) : T_2$.

In this case $e = \text{let}(e_1; x \,.\, e_2)$.

It is never the case that $\text{let}(e_1; x \,.\, e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(e_1 \text{val})$. Suppose the left disjunct is true. In this case, choose $e' = [e_1/x]e_2$ and we know $e \mapsto e'$ by the LetVal rule.

Step 1: Progress

If $\varnothing \vdash e : T$, then either e val or there is a e' such that $e \mapsto e'$.

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\varnothing \vdash e_1 : T_1$ and $x : T_1 \vdash e_2 : T_2$, then $\varnothing \vdash \mathsf{let}(e_1; x\,.\,e_2) : T_2$.

In this case $e = \mathsf{let}(e_1; x\,.\,e_2)$.

It is never the case that $\mathsf{let}(e_1; x\,.\,e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(\underline{e_1 \text{val}})$. Suppose the right disjunct is true.

Step 1: Progress

$\boxed{\text{If } \varnothing \vdash e : T\text{, then either } e \text{ val or there is a } e' \text{ such that } e \mapsto e'.}$

Proof. By induction over the derivation of $\varnothing \vdash e : T$.

Case (Plus): Suppose if $\underline{\varnothing \vdash e_1 : T_1}$ and $x : T_1 \vdash e_2 : T_2$, then $\varnothing \vdash \mathsf{let}(e_1; x\,.\,e_2) : T_2$.

In this case $e = \mathsf{let}(e_1; x\,.\,e_2)$.

It is never the case that $\mathsf{let}(e_1; x\,.\,e_2)$ val. Thus, it suffices to show that there is an e' with $e \mapsto e'$. It is the case that e_1 val or $\neg(\underline{e_1 \text{val}})$. Suppose the right disjunct is true. By the IH, either e_1 val or there is a e_1' such that $e_1 \mapsto e_1'$. But, we know $\neg(e_1 \text{ val})$, thus $e_1 \mapsto e_1'$. So choose $e' = \mathsf{let}(e_1'; x\,.\,e_2)$ and we know $e \mapsto e'$ by the Let1 rule.

Step 2: Preservation

If $\varnothing \vdash e : T$ and $e \mapsto e'$, then $\varnothing \vdash e' : T$

Proof. By induction over the derivation of $e \mapsto e'$.

Case (PlusVal): Suppose $\text{plus}(\text{num}[n_1]; \text{num}[n_2]) \mapsto \text{num}[n_1 + n_2]$.

In this case $e = \text{plus}(\text{num}[n_1]; \text{num}[n_2])$, $e' = \text{num}[n_1 + n_2]$, and $T = \text{Num}$.

This case easily holds by applying the Num rule.

Step 2: Preservation

If $\varnothing \vdash e : T$ and $e \mapsto e'$, then $\varnothing \vdash e' : T$

Proof. By induction over the derivation of $e \mapsto e'$.

Case (Plus1): Suppose $e_1 \mapsto e_1'$ and $\text{plus}(e_1; e_2) \mapsto \text{plus}(e_1'; e_2)$.

In this case $e = \text{plus}(e_1; e_2)$, $e' = \text{plus}(e_1'; e_2)$, and $T = \text{Num}$.

Step 2: Preservation

$\boxed{\text{If } \varnothing \vdash e : T \text{ and } e \mapsto e', \text{ then } \varnothing \vdash e' : T}$

Proof. By induction over the derivation of $e \mapsto e'$.

Case (Plus1): Suppose $\underline{e_1 \mapsto e_1'}$ and $\text{plus}(e_1; e_2) \mapsto \text{plus}(e_1'; e_2)$.

In this case $e = \text{plus}(e_1; e_2)$, $e' = \text{plus}(e_1'; e_2)$, and $T = \text{Num}$.

Step 2: Preservation

If $\varnothing \vdash e : T$ and $e \mapsto e'$, then $\varnothing \vdash e' : T$

Proof. By induction over the derivation of $e \mapsto e'$.

Case (Plus1): Suppose $\underline{e_1 \mapsto e_1'}$ and $\text{plus}(e_1; e_2) \mapsto \text{plus}(e_1'; e_2)$.

In this case $e = \text{plus}(e_1; e_2)$, $e' = \text{plus}(e_1'; e_2)$, and $T = \text{Num}$. By inversion (previously proved), we know that $\varnothing \vdash e_1 : \text{Num}$ and $\varnothing \vdash e_2 : \text{Num}$. By the IH, we know that $\varnothing \vdash e_1' : \text{Num}$, and our result follows by using the Plus rule.

Step 2: Preservation

If $\varnothing \vdash e : T$ and $e \mapsto e'$, then $\varnothing \vdash e' : T$

Proof. By induction over the derivation of $e \mapsto e'$.

Case (LetVal): Suppose $e_1$val and let$(e_1; x\,.\,e_2) \mapsto [e_1/x]e_2$.

In this case $e =$ let$(e_1; x\,.\,e_2)$, $e' = [e_1/x]e_2$. By inversion (previously proved), we know that $\varnothing \vdash e_1 : T'$ and $x : T' \vdash e_2 : T$ for some type T'. It suffices to show that $\varnothing \vdash [e_1/x]e_2 : T$.

Step 2: Substitution for typing

$$\boxed{\text{If } \Gamma \vdash e_1 : T_1 \text{ and } \Gamma, x : T_1 \vdash e_2 : T_2\text{, then } \Gamma \vdash [e_1/x]e_2 : T_2.}$$

Step 2: Preservation

If $\varnothing \vdash e : T$ and $e \mapsto e'$, then $\varnothing \vdash e' : T$

Proof. By induction over the derivation of $e \mapsto e'$.

Case (LetVal): Suppose $e_1$val and let$(e_1; x \,.\, e_2) \mapsto [e_1/x]e_2$.

In this case $e =$ let$(e_1; x \,.\, e_2)$, $e' = [e_1/x]e_2$. By inversion (previously proved), we know that $\varnothing \vdash e_1 : T'$ and $x : T' \vdash e_2 : T$ for some type T'. It suffices to show that $\varnothing \vdash [e_1/x]e_2 : T$. Our result follows directly from substitution for typing.