Two big questions:

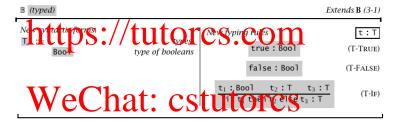
- What can we say about a term without running it? (Static Analysis)
 Can we tell prom will get truck Cithou Cuming it? (Types)

A type is a means of classifying terms. We will want these to "play well" with the **reduction relation**.

WeChat: cstutorcs

Typing Rules for Booleans

Acorparamenatics Prepire extio Es xenemain Help inference rules.



Typing If

Note the form of the rule T-If.

Assignment Project Exam Help

Otherwise, the expression has no type

• Otherwise, the expression has no type

A term which can be typed is called **typable**, or **well-typed**. A term which can't be typed typed in typation com.

Another way to say it: the type relation is **not total** on terms.

WeChat: cstutorcs

Typing If

Note the form of the rule T-If.

Assignment Project Exam Help

Otherwise, the expression has no type

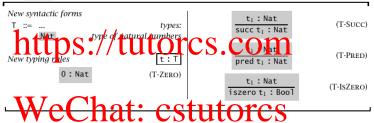
A term which can be typed is called **typable**, or **well-typed**. A term which can't be typed in typed for the typed is called **typable**, or **well-typed**. A term which

Another way to say it: the type relation is not total on terms.

WeChat: cstutorcs

The following evaluates to a value, but is untypeable:

if true then false else 0 (1)



Definition of the Typing Relation

Assping nation of the property of the Sping rules given in the last two figures.

- A terinftit pretyped unter to B. Commit t: T
- When talking about types, we will often make statements like:
- If a term of the form succ t_1 has any type at all, then it has type Nat.

There is a format flow in the SST, of typing information.

Inversion of the Typing Relation

The following inversion rules are immediately derivable from our typing rules:

Assignment Project Exam Help

$$true: R \implies R = Bool \tag{2}$$

if
$$t_1$$
 then t_2 else $t_3: R \implies t_1: Bool \land t_2: R \land t_3: R$ (4)

$$We Chair \stackrel{0:R}{\underset{R}{\rightleftharpoons}} \stackrel{R}{\underset{R}{\rightleftharpoons}} \stackrel{Nat}{\underset{Nat}{\rightleftharpoons}} \stackrel{(5)}{\underset{Nat}{\rightleftharpoons}} \stackrel{(5)}{\underset{(6)}{\rightleftharpoons}}$$

$$pred t_1: R \implies R = Nat \wedge t_1: Nat$$
 (7)

iszero
$$t_1: R \implies R = Bool \land t_1: Nat$$
 (8)

Consider the term if iszero 0 then 0 else pred 0 Let's draw (by bard) of typing derivation for it.

WeChat: cstutorcs

Uniqueness of Types

Assignment Project Exam Help

Each term t has at most one type. That is, if t is well-typed, then its type is unique. Additionally, there is only one derivation of this type, based on our inference rule to t . / tutores.com

• Proof is by structural induction on t, and uses inversion.

Note that induction over typing derivations is also a valid means to prove certain protection Chat: CSTUTORCS

Assignment Project Exam Help The most important property of any type system: safety.

ne most important property of any type system: **safety**

- Slogan: Well-typed terms can't go wrong
- i.e., interpresel that tores seem

We break safety down into two pieces:

Wechat: Cstutorcs

Progress + Preservation

Awailay of the Indian State of Typed Arithmetic Expressions Awailay of the Indian State of Typed Arithmetic Expressions Propressions Propression Propressi

THEOREM [Preservation of Typed Arithmetic Expressions]
If a well-typed term takes a step of evaluation, then the resulting term is also well-typed.

- Taken together, we can say that any well-typed term will eventually evaluate to a well-typed value without getting stuck.
- We can argue this inductively over evaluation derivations.

The **canonical forms** of a type are the values which have that type.

■ If v is a value of type Bool, then v is either true or false

- ② If v is a value of type Nat, then v is a numeric value.
 - That is, v is either 0, or succ nv, where nv is also a numeric value.

Canonical Form of Bool. Nat

Avissignment Project Exam Help

By analysis of all values forms: true, false, 0, and succ nv.

- For true and false, get Bool from inversion.
 For 0 get part in teledrores.com
- For succ *nv* inversion gives that term must have type *Nat*, not *Bool*.

If v is a valve of the latter (Cs) the to fice nv is a value of type Nat.

Argument is very similar to above.

THEOREM: [Progress]

Suppose t:T. Then t is either a value, or else there is some t' such that

 $t \rightarrow t'$ https://tutorcs.com

By Induction on typing derivations:

• T-True T-False and T-Zero, all apply if t is a value. **WeChat: CSTUTORCS**

By inversion:

https://tutorcsrcom t3: T

- By the induction hypothesis, t_1 is either a value, or there is some t_1' such that $t_1 \rightarrow t_1'$ be Castlate of the Sononical forms lemma. In these cases either E-IfTrue or E-IfFalse apply to t respectively.
 - ★ If $t_1 \rightarrow t'_1$, then E-If is applicable to t.

Proof of Progress III

- T-Succ. Inversion gives $t = \text{succ } t_1 \wedge t_1 : Nat$
- ▶ IH: either t_1 value, or $\exists t_1'$ such that $t_1 \to t_2'$ ASSIGN IN THE E-Succ is applicable. (compact the length of the control of the co
 - T-Pred. Inversion gives $t = \text{pred } t_1 \wedge t_1 : Nat$
 - I the value, t mut be fumeric value of the canonical forms lemma.
 - If $t_1 = 0$. E-PredZero applies to t.
 - If $t_1 = succ\ t_2$, E-PredSucc applies to t.
 - ***** If $t_1 \rightarrow t_1'$, the congruency rule E-Pred applies to t.
 - T-IsZevaluese meat $\stackrel{*}{=}$ iszertit to the second t_1 is either a value, or $\exists t_1'$ such that $t_1 \rightarrow t_1'$
 - - ★ If t_1 is a value, must be NV by canonical from lemma.
 - If $t_1 = 0$, E-IsZeroZero applies to t.
 - If $t_1 = \text{succ } t_2$, E-IsZeroSucc applies to t.
 - ★ If $t_1 \rightarrow t'_1$, the congruency rule E-IsZero applies to t.

Proof of Preservation I

$\underset{t:\ T\land\ t\to\ t}{\text{HEOREM}} [\text{Preservation of Typed Arithmetic Expressions}] \\ \text{He}]$

Induction on typing derivations; if last step was: T-True: the typing derivations; if last step was:

T-False, T-Zero: same.

T-Succ: t = succ $t_1 \wedge T = Nat \wedge t_1 : Nat$

- only only estrutores
- Plus t_1 : Nat implies t'_1 : Nat.
- From $t' = succ \ t'_1$ and $t'_1 : Nat$, typing says t' : Nat

Proof of Preservation II

T-If: $t = \text{if } t_1 \text{ then } t_2 \text{ else } t_3, t_1 : Bool \land t_2 : T \land t_3 : T$ ANS SI BANKON BENEFIT LAND HELP • E-IfTrue: $t_1 = \text{true and } t' = t_2 \implies t' : T$.

- E-IfFalse: $t_2 = \text{false} \text{ and } t' = t_3 \Longrightarrow t' : T$.
- E-If: https://tuteorese.com/1 then t2 else t3.
 - ► IH: $t_1: T \land t_1 \rightarrow t_1' \implies t_1': T$.

 * $t_1: Bool$ (via typing relation case analysis)

 * Thus $t_1: Bool$ by IH
- ▶ As t'_1 : Bool, t_2 : T and t_3 : T, typing gives if t'_1 then t_2 else t_3 : T. (and so on; T-Pred does require more care)