The self-referential problem

Holo

Tuesday 16th May, 2023

Type-Token Distinction

Is 'Red' red?

Is 'Red' red?

Is 'Red' red?

Is 'Red' red?
Is 'Red' red?
Is 'Red' red?

Is 'Red' red?

Is 'Red' red?

Is 'Red' red?

Is 'Blue' red?

Is 'Red' red?
Is 'Red' red?
Is 'Red' red?
Is 'Blue' red?

Token (linguistics)

A **token** is an instance of word

Is 'Red' red?
Is 'Red' red?
Is 'Red' red?
Is 'Blue' red?

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."

The sentence has 3 types

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."
The sentence has 3 types: "a"

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."
The sentence has 3 types: "a", "is"

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."
The sentence has 3 types: "a", "is"and "rose"

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."
The sentence has 3 types: "a", "is"and "rose"
The sentence contains 3 instances of the type "a"

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."
The sentence has 3 types: "a", "is"and "rose"
The sentence contains 3 instances of the type "a", 2 instances of the type "is"

Token (linguistics)

A **token** is an instance of word

Type (linguistics)

A **type** is an abstract idea that a *token* refers to

"A rose is a rose is a rose."
The sentence has 3 types: "a", "is"and "rose"
The sentence contains 3 instances of the type "a", 2 instances of the type "is"and 3 instances of the type "rose".

Is 'Red' red?

Is 'Red' red?

The token 'Red' is not red

Is 'Red' red?

The token 'Red' is not red, the type is *also not red*.

Is 'Red' red?

The token 'Red' is not red, the type is *also not red*.

Is 'Red' red?

Is 'Red' red?

The token 'Red' is not red, the type is *also not red*.

Is 'Red' red?

The token *is* red

Is 'Red' red?

The token 'Red' is not red, the type is *also not red*.

Is 'Red' red?

The token *is* red, this token refers to the same type as before, so it is also not red.

Is 'Red' red?
The token 'Red' is not red, the type is *also not red*.
Is 'Red' red?
The token *is* red, this token refers to the same type as before, so it is also not red.

When asking "is 'A' B?" and the question doesn't make sense, the answer is "no".

Is 'Red' red?
The token 'Red' is not red, the type is *also not red*.
Is 'Red' red?
The token *is* red, this token refers to the same type as before, so it is also not red.

When asking "is 'A' B?" and the question doesn't make sense, the answer is "no".so the type of 'Red', which doesn't have any colour what so ever, is **not** red.

Examples

Examples

The '(token) palindrome' is not a palindrome.

Examples

The '(token) palindrome' is not a palindrome.
The '(token) civic' is a palindrome.

Examples

The '(token) palindrome' is not a palindrome.
The '(token) civic' is a palindrome.
The '(type) cat' is not "three letters long"

Examples

The '(token) palindrome' is not a palindrome.
The '(token) civic' is a palindrome.
The '(type) cat' is not "three letters long"
The '(token) writable' is writable.

Examples

The '(token) palindrome' is not a palindrome.
The '(token) civic' is a palindrome.
The '(type) cat' is not "three letters long"
The '(token) writable' is writable.
The '(type) writable' is not writable.

Examples

The '(token) palindrome' is not a palindrome.
The '(token) civic' is a palindrome.
The '(type) cat' is not "three letters long"
The '(token) writable' is writable.
The '(type) writable' is not writable.

From now on, unless specify otherwise, when asking "is 'A' B?", we will talk about the *type* of 'A' (Newhard variation).

Partitioning the English language

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".

Every word in the English language **is not** "red".

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".

Every word in the English language **is not** "red".
Every word in the English language **is** "word"*.

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".

Every word in the English language **is not** "red".
Every word in the English language **is** "word"*.
Some words in the English language are "nouns" and some are not.

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".

Every word in the English language **is not** "red".
Every word in the English language **is** "word"*.
Some words in the English language are "nouns" and some are not.
Some words in the English language are "offensive" and some are not.

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".

Every word in the English language **is not** "red".
Every word in the English language **is** "word"*.
Some words in the English language are "nouns" and some are not.
Some words in the English language are "offensive" and some are not.

* we can also talk about "sentences" and not only words, in which case not everything will be a "word"

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".
Alternative form: Every word is either **B** or not

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".
Alternative form: Every word is either **B** or not

"Every word is either **red** or not"

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".
Alternative form: Every word is either **B** or not

"Every word is either **red** or not"
"Every word is either a **word** or not"

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".
Alternative form: Every word is either **B** or not

"Every word is either **red** or not"
"Every word is either a **word** or not"
"Every word is either a **noun** or not"

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".
Alternative form: Every word is either **B** or not

"Every word is either **red** or not"
"Every word is either a **word** or not"
"Every word is either a **noun** or not"
"Every word is either **offensive** or not"

The Partition Claim

Given a property "B" we can check every word in the English language and put it in one of 2 buckets: "is B", "is not B".
Alternative form: Every word is either **B** or not

"Every word is either **red** or not"
"Every word is either a **word** or not"
"Every word is either a **noun** or not"
"Every word is either **offensive** or not"

Grelling–Nelson paradox

"**The Partition Claim**" is false

Autological words and Heterological words

Autological

A word is **autological** if it describes itself

Autological

A word is **autological** if it describes itself

"word" is an autological word. "noun" is an autological word.

Autological

A word is **autological** if it describes itself

"word" is an autological word. "noun" is an autological word.

Heterological

A word is **heterological** if it does not describes itself. That is, a word heterological if it is not autological

Autological

A word is **autological** if it describes itself

"word" is an autological word. "noun" is an autological word.

Heterological

A word is **heterological** if it does not describes itself. That is, a word heterological if it is not autological

"red" is heterological, "offensive" is heterological.

The Partition Claim for Autological and Heterological words

Every word is either autological or heterological.

The Partition Claim for Autological and Heterological words

Every word is either autological or heterological.

We will show that the above statement is **false**, that is, there are words that are neither autological nor heterological.

The Partition Claim for Autological and Heterological words

Every word is either autological or heterological.

We will show that the above statement is **false**, that is, there are words that are neither autological nor heterological.
Similarly we will find a word that *can be* **both** autological and heterological.

Grelling–Nelson paradox

Also known as Weyl's paradox and Grelling's paradox

Heterological is not heterological

Assume **"Heterological" is heterological.**

Heterological is not heterological

Assume **"Heterological" is heterological.**
Remember that heterological means "not autological"

Heterological is not heterological

Assume **"Heterological" is heterological.**
Remember that heterological means "not autological"
"Heterological" is not autological.

Heterological is not heterological

Assume **"Heterological" is heterological.**
Remember that heterological means "not autological"
"Heterological" is not autological.
But "autological" means "describes itself", so the above is actually

Heterological is not heterological

Assume **"Heterological" is heterological.**
Remember that heterological means "not autological"
"Heterological" is not autological.
But "autological" means "describes itself", so the above is actually
"Heterological" is not heterological.

Heterological is not heterological

Assume **"Heterological" is heterological.**
Remember that heterological means "not autological"
"Heterological" is not autological.
But "autological" means "describes itself", so the above is actually
"Heterological" is not heterological.
But a word can't be **both** heterological and not heterological, so "heterological" cannot be heterological.

Heterological is not autological

Assume **"Heterological" is autological.**

Heterological is not autological

Assume **"Heterological" is autological.**
Remember that autological means "describes itself", so we have

Heterological is not autological

Assume **"Heterological" is autological.**

Remember that autological means "describes itself", so we have

"Heterological" is heterological.

Heterological is not autological

Assume **"Heterological" is autological.**
Remember that autological means "describes itself", so we have
"Heterological" is heterological.
But we already saw that this is impossible, so heterological can't be autological.

Autological can be both autological and heterological

Unlike heterological, where it can be neither heterological nor autological, autological can be both.

Autological can be both autological and heterological

Unlike heterological, where it can be neither heterological nor autological, autological can be both.
By that we mean that if we claim that "autological" is autological, we will not find any holes that will imply otherwise

Autological can be both autological and heterological

Unlike heterological, where it can be neither heterological nor autological, autological can be both.
By that we mean that if we claim that "autological" is autological, we will not find any holes that will imply otherwise, but similarly if we claim that "autological" is heterological we won't find any holes in our logic.

Autological can be both autological and heterological

Unlike heterological, where it can be neither heterological nor autological, autological can be both.
By that we mean that if we claim that "autological" is autological, we will not find any holes that will imply otherwise, but similarly if we claim that "autological" is heterological we won't find any holes in our logic.
This is the *dual* version of the paradox

Trying to resolve the paradox

One might try to resolve the paradox by redefining "heterological" to "words that are not autological and are not the word 'heterological'"

Trying to resolve the paradox

One might try to resolve the paradox by redefining "heterological" to "words that are not autological and are not the word 'heterological'", this will just shift the problem to "nonautological".

Trying to resolve the paradox

One might try to resolve the paradox by redefining "heterological" to "words that are not autological and are not the word 'heterological'", this will just shift the problem to "nonautological". In fact, any way of trying to resolve it by excluding some cases will just shift the problem into different words.

Trying to resolve the paradox

One might try to resolve the paradox by redefining "heterological" to "words that are not autological and are not the word 'heterological'", this will just shift the problem to "nonautological". In fact, any way of trying to resolve it by excluding some cases will just shift the problem into different words.

The Partition Claim

Every word is either **B** or not

Cannot hold for every "B".

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory
- *Curry's Paradox* in formal logic

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory
- *Curry's Paradox* in formal logic
- *The Halting Problem* in computer science

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory
- *Curry's Paradox* in formal logic
- *The Halting Problem* in computer science
- *Girard's paradox* in type theory

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory
- *Curry's Paradox* in formal logic
- *The Halting Problem* in computer science
- *Girard's paradox* in type theory
- *Curry's Paradox* (again) in lambda calculus

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory
- *Curry's Paradox* in formal logic
- *The Halting Problem* in computer science
- *Girard's paradox* in type theory
- *Curry's Paradox* (again) in lambda calculus
- ...

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory
- *Curry's Paradox* in formal logic
- *The Halting Problem* in computer science
- *Girard's paradox* in type theory
- *Curry's Paradox* (again) in lambda calculus
- ...

Pretty much any subject that tries to be "foundational" (be the 'building blocks' of everything else) have a variation of the paradox.

The source of the paradox

The reason the paradox exists is because we are trying to ask self-referential questions: "is *heterological* heterological?".
This paradox is not limited to linguistics, it is called:

- *Russell's Paradox* in set theory
- *Curry's Paradox* in formal logic
- *The Halting Problem* in computer science
- *Girard's paradox* in type theory
- *Curry's Paradox* (again) in lambda calculus
- ...

Pretty much any subject that tries to be "foundational" (be the 'building blocks' of everything else) have a variation of the paradox.
In formal settings (like maths or computer science) we resolve the paradox by restricting the formal rules we are using.