An incorrect number of generic arguments was provided.

Erroneous code example:

```
struct Foo<T> { x: T }

struct Bar { x: Foo }              // error: wrong number of type arguments:
                                   //        expected 1, found 0
struct Baz<S, T> { x: Foo<S, T> } // error: wrong number of type arguments:
                                   //        expected 1, found 2

fn foo<T, U>(x: T, y: U) {}
fn f() {}

fn main() {
    let x: bool = true;
    foo::<bool>(x);                // error: wrong number of type arguments:
                                   //        expected 2, found 1
    foo::<bool, i32, i32>(x, 2, 4); // error: wrong number of type arguments:
                                   //        expected 2, found 3
    f::<'static>();                // error: wrong number of lifetime arguments
                                   //        expected 0, found 1
}
```

When using/declaring an item with generic arguments, you must provide the exact same number:

```
struct Foo<T> { x: T }

struct Bar<T> { x: Foo<T> }              // ok!
struct Baz<S, T> { x: Foo<S>, y: Foo<T> } // ok!

fn foo<T, U>(x: T, y: U) {}
fn f() {}

fn main() {
    let x: bool = true;
    foo::<bool, u32>(x, 12);             // ok!
    f();                                 // ok!
}
```