GitHub Copilot:

```
std::string
std::string_view
std::string
std::string
std::string
std::string_view
std::string_view
std::string_view
std::string_view
```

GitHub Copilot:

```
bool
remove(storage&
contacts,
std::string_view
name)
contacts
remove
contacts
true
contacts
contacts
contacts
storage&
contacts
storage
contacts
```

Summary of Discussion on C++ Code

GitHub Copilot

November 6, 2023

1 Code Overview

The provided C++ code defines a function named `add` in the file `contact_list.cpp`. The function is used to add a contact to a storage object. The function signature is as follows:

```
bool add(storage& contacts, std::string_view name, number_t number)
```

2 Function Components

The function takes three parameters:

- `contacts`: A reference to a `storage` object.
- `name`: A `std::string_view` representing the name of the contact.
- `number`: A `number_t` representing the contact's number. `number_t` is a type alias for `int64_t`.

The function returns a boolean value indicating whether the contact was added successfully.

3 Function Logic

The function first checks if the `name` is empty or already exists in the `contacts.names` vector. If either condition is true, the function returns `false`.

If the name is valid, the function adds the `name` and `number` to the `contacts.names` and `contacts.numbers` vectors, respectively, and returns `true`.

4 C++ Concepts

The function uses several key C++ concepts:

- **References**: The `contacts` parameter is passed by reference, allowing the function to modify the original `storage` object.

- **STL Algorithms and Iterators**: The function uses the `std::find` algorithm to search for `name` in the `contacts.names` vector. The `begin()` and `end()` functions return iterators to the beginning and end of the vector.
- **Vector Operations**: The function uses the `push_back` function to add elements to the end of a vector.