

simple

```
template <typename T>
void foo_impl (const T& val, std::true_type) {
    std::cout << "foo() called for pointer to " << *val << std::endl;
}
// foo() implementation for non-pointer types:

template <typename T>
void foo_impl (const T& val, std::false_type) {
    std::cout << "foo() called for value to " << val << std::endl;
}

template <typename T>
void foo (const T& val)
{
    foo_impl (val, std::is_pointer<T>());
}
```