

C++17起对lambda捕获增加了扩展

*this捕获

```
class A
{
public:
    A() = default;
    void test()
    {
        auto foo = [*this]()mutable /**this捕获是把当前的对象拷贝一份而不是引用(*this捕获需要C++17起)
        {
            a = 1;
            std::cout << a << " " << b << " " << c << std::endl;
        };
        foo();
        std::cout << a << std::endl; //a依旧为0不做变化
    }
private:
    int a{}, b{}, c{};
};
```

constexpr捕获

```
int main()
{
    auto foo1 = [] { return 5 * 3; };
    std::array<int, foo1()> a{};
    //合法,因为从c++17起,只要能在编译期求值的lambda都会隐含constexpr

    auto g = [](int x)constexpr -> int //显式声明
    {
        return x * 5;
    };

    std::cout << g(2);
}
```