

- Ошибка может возникнуть, когда имеется несколько функций с одинаковыми именами, но разными типами параметров. Явная специализация позволяет вызывать необходимую функцию, если находится одна единственная наиболее подходящая функция. Однако, если не окажется одной единственной подходящей функции, то произойдёт ошибка компиляции.

```
void f(int x) {...}
void f(long x) {...}
```

```
int main()
{
    f(5.0) // compile error
}
```

- Шаблонная функция может одинаково хорошо соответствовать, как и нешаблонная. Тогда вызывается нешаблонная
- При перегрузке может возникнуть ситуация, когда больше 1 шаблона подходят для инстанцирования. При этом возникнет ошибка.

```
template<typename T>
T sum(T a, T b)
{
    return a+b;
}
```

```
template<typename T>
double sum(double a, double b)
{
    return a+b;
}
```

```
int main()
{
    cout<<sum(2.5, 1.6)<<endl; // run time error
    return 0;
}
```