Value Manipulation

```
template <int N>
struct Factorial{
    static int const value= N * Factorial<N-1>::value;
};
template <>
struct Factorial<1>{
    static int const value = 1;
};

std::cout << Factorial<5>::value << std::endl;
std::cout << 120 << std::endl;
```

```
Factorial<5>::value
  → 5*Factorial<4>::value
  → 5*4*Factorial<3>::value
  → 5*4*3*Factorial<2>::value
  → 5*4*3*2*Factorial<1>::value   →   5*4*3*2*1= 120
```

Type Manipulation

```
template <typename T>
struct RemoveConst{
  typedef T type;
};

template <typename T>
struct RemoveConst<const T>{
  typedef T type;
};

int main(){
  std::is_same<int, RemoveConst<int>::type>::value;       // true
  std::is_same<int, RemoveConst<const int>::type>::value  // true
}
```