

# Практикум 12

## Шаблони

### 1. Проста функция.

Да се реализира темплейтна (шаблон на) функция за сортиране на масив. Какво се очаква поддържат задължително обектите, които я използват?  
Темплейт или функция от по-висок ред?

### 2. Шаблонен клас.

Да се реализира темплейтен вектор. Може да използвате наготово кода от упражнения за начало на темплейтния ви вектор. Очаква ли се той да се грижи за изчистване на паметта на обектите от шаблонния тип?

### 3. Template specialization for the Vector class. [Challenge]

Write a template specialization for the Vector class of boolean elements. The current vector implementation you have would use elements of type `bool`, which are 1 byte in size each. As we know, a value of `True/False` can be represented by 1/0. We could use each of the bits as a value, reducing our memory 8-fold. In short, we'd like to create a specialization of the Vector that stores booleans as bits instead of a full byte.

#### Constructors:

`MyVector()` - create empty array.

`MyVector(size_t A)` - create array with A zeros (unset bits / false values).

`MyVector(const char* Str)` - parses the given string representation of bits.

(E.g.: `MyVector<bool> vec("1001")` creates an array with values `True, False, False, True`)

#### Methods:

`operator std::string()` - Cast to string.

`size_t Count()` - count of set bits (ones).

`bool IsSet(index)` - tests if a given bit is set.

`bool Any()` - returns if any of the bits is set.

`bool None()` - returns true if no bits are set.

`bool All()` - returns true if all bits are set.

For the extra curious, implement `operator[]` that returns an internal (private) proxy object that can be used to set/unset bits in the vector. The proxy object should hold a reference or a pointer to the block of the vector it'll be modifying, as well as an index - which part of the block it will be modifying.

Example usage:

```
MyVector<bool> vec("1001001001001");  
vec[10] = true;  
vec[11] = true;  
std::cout << vec;  
// Should print out 1001001001111
```