

Задание по C++ №9, осень 2018 года

1. По аналогии с функцией `printf` с лекции напишите функцию `print_values` с переменным числом аргументов, которая для каждого аргумента выводит его тип и значение в поток `std::cout`. Функция должна работать с произвольным числом аргументов. Указание: для вывода имени типа нужно использовать `std::type_info::name`.

Например, вызов функции

```
print_values(0, 3.5, "Hello");
```

должен вывести на стандартный вывод примерно следующее (таким будет вывод на g++).

```
i: 0
```

```
d: 3.5
```

```
PKc: Hello
```

2. Реализуйте следующий класс `Array`:

```
template<class T>
struct Array
{
    explicit Array(size_t size = 0);
    Array(Array const& a);
    Array & operator=(Array const& a);
    ~Array();

    size_t size() const;
    T & operator[](size_t i);
    T const& operator[](size_t i) const;

    // перемещающий конструктор

    // перемещающий оператор присваивания

private:
    size_t size_;
    T * data_;
};
```

3. Напишите шаблонную функцию `to_pair`, которая принимает произвольный `std::tuple` и два индекса внутри и возвращает `std::pair`, содержащий элементы переданного `std::tuple` с соответствующими индексами.

Пример:

```
auto t = std::make_tuple(0, 3.5, "Hello");
std::pair<double, char const*> p = to_pair<1,2>(t);
// p содержит 3.5 и "Hello"
```

Замечание: реализация должна работать в рамках стандарта C++11.