

- (1) (3 балла) Для каждого из приведенных утверждений укажите, является ли оно верным (+) или не является (-).
- (a) Объекты `constexpr` инициализируются объектами, значения которых известны во время компиляции.
 - (b) Когда конструктор `std::shared_ptr` вызывается с обычным указателем — он создает управляющий блок.
 - (c) Переменяющий оператор присваивания всегда обязан выполнять последовательное почленное перемещение нестатических членов-данных класса.
 - (d) В C++11 реализован автоматический вывод типа возвращаемого значения функции.
- (2) (3 балла) Что такое `constexpr` и `constinit` переменные? Как они работают и в чём их отличия?
- (3) (3 балла) Что такое `std::string_view`? Приведите пример, когда его использование может быть полезным.
- (4) (3 балла) Приведите примеры `lvalue`, `xvalue` и `rvalue` выражений.
- (5) (3 балла) В чём преимущество использования `std::make_shared<T>()` в сравнении с конструкторией `std::shared_ptr<T>(new T)`?
- (6) (3 балла) Что такое локальность данных в памяти? Что происходит, когда данные обладают плохой локальностью?
- (7) (3 балла) Опишите общую схему и идею паттерна проектирования Посетитель (Visitor). В каких случаях этот паттерн может использоваться?
- (8) (3 балла) Что такое идиома PImpl? Для чего она может использоваться?
- (9) (2 балла) Какой паттерн проектирования используется, когда необходимо уведомлять объекты об изменениях данных в других объектах?
Вопрос сгенерирован при помощи ChatGPT
- (10) (2 балла) В программе есть класс, который взаимодействует с базой данных, чтобы получать информацию о продуктах в интернет-магазине. Вместо реального обращения к базе данных через ORM-фреймворк этот класс в unit-тесте использует объект, который предоставляет predefined данные о продуктах. К какому из перечисленных типов устранения зависимостей относится указанный подход?
- (a) dummy
 - (b) stub
 - (c) fake
 - (d) mock
- Вопрос сгенерирован при помощи ChatGPT*
- (11) (4 балла) Какая проблема в работе с памятью есть в следующем коде? Как её можно исправить?

```

void f(std::shared_ptr<C> ptr, int param);

void g() {
    C* ptr = new C;
    for (int i = 0; i < 5; ++i) {
        f(std::shared_ptr<C>(ptr), i);
    }
}

```

- (12) (4 балла) Что напечатает следующая программа, работающая под 64-bit платформой (указатели на все типы имеют размер 8 байтов)?

```

#include <stdint>
#include <iostream>

struct S {
    uint16_t a : 12;
    int16_t b : 9;
    uint32_t c : 12;
    char d : 5;
};

int main() {
    std::cout << sizeof(S) << " " << alignof(S) << std::endl;
    return 0;
}

```

- (13) (4 балла) Что напечатает программа?

```

#include <iostream>
#include <functional>

int f(int x, int y, int z) {
    return -x - y - 2 * z;
}

int g(int x, int y) {
    return 2 * x * x - 2 * y;
}

int main() {
    auto h = std::bind(
        g,
        std::placeholders::_2,
        std::bind(f, std::placeholders::_1, 0, std::placeholders::_1)
    );

    std::cout << h(1, 1, 2) << std::endl;

    return 0;
}

```

- (14) (4 балла) Перепишите следующий код таким образом, чтобы область видимости переменной `x` была ограничена условным оператором.

```
int x = std::rand();
if (x % 2 == 0) {
    x >>= 1;
    std::cout << "case 1: " << x << std::endl;
} else {
    --x;
    std::cout << "case 2: " << x << std::endl;
}
```

(15) (4 балла) В программе на C++ определены следующие конструкции:

```
int f();
const int& g();
int&& h();
// ...
int x = 5;
int *p = &x;
```

Укажите, к какому классу (lvalue, xvalue, rvalue) относятся следующие выражения:

```
f();
g();
h();
*p;
14;
static_cast<int&&>(x);
std::move(5);
```