Общий ТЕСТ для кандидатов

### Найти проблему (проблемы) в коде

1. Рассмотрите приведенный ниже фрагмент кода:

template <typename T> void Func2(std::auto\_ptr<T> v1, std::auto\_ptr<T> v2, std::auto\_ptr<T> v3)

{

<…>

}

template <typename T> bool Func1()

{

try

{

std::auto\_ptr<T> p(new T);

Func2(p, std::auto\_ptr<T> (new T), std::auto\_ptr<T>(new T));

p->SomeMethod();

return true;

}

catch(const std::bad\_alloc&)

{

return false;

}

}

Ответьте на вопросы:

1. Есть ли в приведенном коде какие-либо проблемы?
2. Является ли он exception safe?

Если вы считаете, что проблемы есть, укажите их (в стиле:  строка кода - суть проблемы; например: printf(s) - использование неинициализированной переменной).

2. Что некорректного в следующем коде? Достаточно указать строку кода и коротко описать проблему.

class Base

{

public:

    Base()

    {

        Init();

    }

    virtual void Init() = 0;

    void Shutdown() throw(std::bad\_alloc);

    ~Base()

    {

        Shutdown();

    }

};

class Derived : public Base

{

public:

    Derived();

    void Init();

    void Shutdown();

    ~Derived()

    {

        Shutdown();

    }

};

void DoSomething(std::auto\_ptr<Base>) throw(std::exception);

void f()

{

    try

    {

        std::auto\_ptr<Base> b(new Derived);

        DoSomething(b);

    }

    catch(const std::exception & e)

    {

        // ...

    }

}

3. Что некорректного в следующем коде? Достаточно указать строку кода и коротко описать проблему

int fn (int);

void foo ()

{

int \* a = new int [5];

for (int i = 0; i < 5; ++i)

a [i] = fn (i);

delete a;

}

4. Что некорректного в следующем коде? Достаточно указать строку кода и коротко описать проблему

class String

{

public:

String () : m\_str (0) {}

String (const char \*s) : m\_str (strdup (s)) {}

~String () { free (m\_str); }

void operator= (const char \*s)

{ free (m\_str); m\_str = strdup (s); }

operator const char \* ()

{ return m\_str; }

private:

const char \* m\_str;

};

### 2. Задачи

1. Написать код блокирующей очереди, имеющей следующий интерфейс (для реализации использовать С++ либо C#):

      void Put(object o)

      object Get()

      void Disable()

Get должен блокироваться если нет элементов в очереди.

Disable должен освобождать все ждущие потоки. При этом Get должен возвращать null (или кидать исключение - как больше нравится). Все последующие вызовы к Get должны просто возвращать null.

Представить набор тестов для проверки корректности имплементации.

2. Написать реализацию функции strlen. Представить набор тестов для проверки корректности имплементации.

3. Поменять значение двух целочисленных (integer) переменных без использования дополнительных ячеек памяти. Представить набор тестов для проверки корректности имплементации.

### Теория тестирования

* 1. Что есть требование? Какого типа бывают? Какова роль специалистов по автоматическому тестированию применительно к требованиям?
  2. Что такое регрессионное тестирование и когда стоит проводить?
  3. Каковы роль, цель и стратегия автоматизации тестирования?
  4. Для приведенного описания требования:  
     During removal the product must uninstall automatically:   
      - Program files  
      - All directories and registry keys created during installation and operation  
      - All temporary and auxiliary objects created during product operation.   
     Требуется:  
     i. Составить сообщение с постановкой задачи по тестированию этого требования/спецификации на английском языке.  
     ii. Составить отчет о потенциальной ошибке для заданного сценария.
  5. Какие Вам известны средства автоматизации тестирования? Каков Ваш уровень знания перечисленных продуктов?