В какой момент времени инициализируются поля объекта класса?



В конструкторе объекта



При первом обращении к объекту, после его создания



При первом запуске программы

Вопрос 2

Дана следующая иерархия классов:

class C { ... };

class B : public C { ... };

class A : public B { ... };

Напишите через запятую порядок создания объектов при создании объекта типа C (например: A,B,C).

C, B, A

Вопрос 3

Для иерархии классов из вопроса 2 напишите через запятую порядок разрушения объектов при разрушении объекта типа A

A, B, C

Вопрос 4

Какими полезными свойствами обладает функция DoMeal из первой лекции?

4 балла



При добавлении нового типа животного или фрукта функцию не придётся изменять



Полезных свойств нет, функция только скрывает настоящий тип объекта, что сильно затрудняет дальнейшую работу с ним



Функция может быть вызвана для любых объектов, унаследованных от базовых классов Animal и Fruit



Функция скрывает настоящий тип объекта, упрощая код, т. к. она явно сообщает, что для её аргументов важна только базовая составляющая

Вопрос 5

Какие проблемы содержатся в определении класса Aristocrat?

class Aristocrat {

public:

    Aristocrat(string title, string first\_name, string second\_name):

        title\_(title),

        first\_name\_(title\_ + first\_name),

        second\_name\_(title\_ + second\_name)

    {

    }

    string first\_name\_;

    string second\_name\_;

    string title\_;

};



поля first\_name\_ и last\_name\_ видны снаружи и могут быть изменены непреднамеренно



нельзя будет создать объект данного класса, т.к. у конструктора класса пустое тело, а значит он ничего не делает



нельзя будет создать объект данного класса, т.к. конструктор у класса объявлен в public-секции



поля first\_name\_ и last\_name\_ не будут проинициализированы правильно

Вопрос 6

Как отработает код ниже?

class Logger {

public:

    Logger(string logger\_type) {

        logger\_type\_ = logger\_type;

    }

private:

    const string logger\_type\_;

};

Logger logger("my\_logger");



программа отработает нормально



код не скомпилируется



в переменной logger\_type\_ окажется пустая строка



программа упадёт при попытке создания объекта logger

7.

Вопрос 7

Как надо определить класс-наследник, который содержит в себе экземпляр базового класса, чтобы все его поля были проинициализированы?

class A {

public:

    A(string name): name\_(name) {

    }

private:

    string name\_;

};



class B : public A {

public:

    B(string name) : base\_(name) {

    }

private:

    A base\_;

};



class B : public A {

public:

    B(string name) : A(name) {

    }

private:

    A base\_;

};



class B : public A {

public:

    B(string name) : name\_(name), base\_(name) {

    }

private:

    A base\_;

};



class B : public A {

public:

    B(string name) : A(name), base\_(name) {

    }

private:

    A base\_;

};

8.

Вопрос 8

Какие из приведённых ниже строчек отработают, если x - это объект класса X, а y - объект класса Y?

class X {

public:

    void setProtectedMember(int value) {

        protected\_ = value;

    }

protected:

    int  protected\_;

    void protectedFunction(int value) {

        private\_ = value;

    }

private:

    int private\_;

};

class Y : public X {

public:

    void setProtected(int value) {

        protected\_ = value;

    }

    void getProtected() {

        cout << protected\_<< "\n";

    }

    void useProtectedFunction(int value) {

        protectedFunction(value);

    }

};



y.getProtected();



y.useProtectedFunction(1);



x.setProtectedMember(0);



x.protectedFunction(5);



x.protected\_ = 3;



x.private\_ = 4;



y.setProtected(1);