

Блок контроля знаний

Вопрос 1. Укажите объектно-ориентированный язык программирования

- А) Ассемблер
- Б) С
- В) С++
- Г) Pascal

Вопрос 2. Чему будет равна переменная `a`, после выполнения кода: `int a; for(a = 0; a < 10; a++) {}`?

- А) 9
- Б) 10
- В) 8
- Г) 1

Вопрос 3. Название С++ предложил

- А) Бьерн Страуструп
- Б) Кэн Томпсон
- В) Рик Масситти
- Г) Дональд Кнут

Вопрос 4. Какой из нижеперечисленных операторов не является циклом в С++?

- А) `do while`
- Б) `repeat until`
- В) `for`
- Г) `while`

Вопрос 5. Чтобы подключить заголовочный файл в программу на С++, например, `iostream`, необходимо написать:

- А) `include #iostream,h;`
- Б) `#include <>c iostream` внутри скобок
- В) `#include <>; c iostream.h` внутри скобок
- Г) `include (iostreamh)`

Вопрос 6. Какой служебный знак ставится после оператора `case` ?

- А) -
- Б) :
- В) .
- Г) ;

Вопрос 7. Какая из следующих записей - правильный комментарий в C++?

- А) /* комментарий */
- Б) ** Комментарий **
- В) {комментарий}
- Г) */ Комментарий */

Вопрос 8. Что будет выведено на экран после выполнения программы?

```
1  intmain()
2  {
3      for(int i = 0; i < 4; ++i)
4      {
5          switch(i)
6          {
7              case0 : std::cout << "0";
8              case1 : std::cout << "1"; continue;
9              case2 : std::cout << "2"; break;
10             default: std::cout << "D"; break;
11         }
12         std::cout << ".";
13     }
14     return0;
15 }
```

- А) 01.2.D.
- Б) 011.2.D
- В) Ошибка компиляции в строке 10
- Г) 0112.D.

Вопрос 9. До каких пор будут выполняться операторы в теле цикла while ($x < 100$)?

- А) Пока x строго меньше 100
- Б) Пока x меньше или равен 100
- В) Пока x больше 100
- Г) Пока x равен 100

Вопрос 10. Программа, переводящая входную программу на исходном языке высокого уровня в эквивалентную ей выходную программу на низкоуровневом языке, близком машинному коду:

- А) транслятор
- Б) компилятор

- В) интерпретатор
- Г) сканер

Вопрос 11. Какому зарезервированному слову программа передаёт управление в случае, если значение переменной или выражения оператора `switch` не совпадает ни с одним константным выражением?

- А) `all`
- Б) `contingency`
- В) `other`
- Г) `default`

Вопрос 12. Какими знаками заканчивается большинство строк кода в Си++?

- А) ; (точка с запятой)
- Б) , (запятая)
- В) : (двоеточие)
- Г) . (точка)

Вопрос 13. Язык программирования С++ разработал

- А) Кен Томпсон
- Б) Никлаус Вирт
- В) Дональд Кнут
- Г) Бьерн Страуструп

Вопрос 14. Каков результат работы следующего фрагмента кода?

```
1   intx = 0;
2
3   switch(x)
4   {
5
6       case1: cout << "Один";
7
8       case0: cout << "Ноль";
9
10      case2: cout << "Привет мир";
11
12  }
```

- А) Привет мир
- Б) Ноль
- В) НольПривет мир
- Г) Один

Вопрос 15. Какой оператор не допускает перехода от одного константного выражения к другому в операторе case?

- А) stop;
- Б) end;
- В) точка с запятой
- Г) break;

Вопрос 16. Какое значение, по умолчанию, возвращает программа операционной системе в случае успешного завершения?

- А) Программа не возвращает значение.
- Б) 0
- В) 1
- Г) -1

Вопрос 17. Какие служебные символы в C++ используются для обозначения начала и конца блока кода (операторные скобки)?

- А) <>
- Б) begin end
- В) { }
- Г) ()

Вопрос 18. Какую функцию обязательно должны содержать все программы на C++?

- А) program()
- Б) system()
- В) main()
- Г) start()

Вопрос 19. Какой из следующих операторов C++ - оператор сравнения двух переменных?

- А) :=
- Б) equal
- В) =
- Г) ==

Вопрос 20. Какой оператор используется для выполнения одного действия, если его условие истинно, и другого действия, если условие ложно?

- А) while
- Б) if
- В) for

Г) case

Вопрос 21. К чему применяется операция инкремента ++?

А) к переменным

Б) к константам

В) к выражениям

Г) к строкам

Вопрос 22. В языке Си++ указатель - это:

А) специальный значок, показывающий, что это динамическая переменная

Б) символическое представление адреса ячейки памяти

В) символ, указывающий на что-либо

Г) метка

Вопрос 23. Составной оператор - это:

А) последовательность операторов, заключенная в фигурные скобки { }

Б) последовательность операторов, заключенная квадратные скобки []

В) последовательность операторов, заключенная в операторные скобки begin ... end

Г) последовательность операторов, заключенная в круглые скобки ()

Вопрос 24. Логическое «не равно» обозначается:

А) <>

Б) ||

В) !

Г) !=

Вопрос 25. Логическое «и» обозначается:

1: =

2: ||

3: &

4: &&

Вопрос 26. Логическое «не» обозначается:

А) !

Б) !!

В) ||

Г) not

Вопрос 27. Каков будет результат выполнения операторов:

```
int x,y,a;
```

```
x=5;
```

```
y=x*2+7;
```

```
a=y/4;
```

А) $x = 5, y = 17, a = 4,25$

Б) $x = 5, y = 17, a = 4$

В) $x = 5, y = 10, a = 2,25$

Г) $x = 5, y = 32, a = 8$

Вопрос 28. Что такое GCC.

А) Компилятор языка C от GNU

Б) Коллекция компиляторов от GNU под различные платформы

В) Компилятор C и C++ для Linux

Г) Компилятор для C++

Вопрос 29. Какой из нижеперечисленных языков программирования не входит в состав GCC?

А) Fortran

Б) C++

В) Basic

Г) ADA

Вопрос 30. Какие из перечисленных архитектур поддерживает GCC?

А) Alpha

Б) ARM

В) x86

Г) Все перечисленные

Вопрос 31. Инкапсуляция – это...

А) механизм, который объединяет данные и методы, манипулирующие этими данными, и защищает и то и другое от внешнего вмешательства или неправильного использования

Б) механизм, который позволяет использовать в дочерних классах функционал родительского класса и, в случае необходимости, дополнять его

В) механизм, который представляет собой способность к изменению функционала, унаследованного от базового класса

Г) объединение данных в группы по типу значения

Вопрос 32. Наследование – это...

А) механизм, который объединяет данные и методы, манипулирующие этими данными, и защищает и то и другое от внешнего вмешательства или неправильного использования

Б) механизм, который позволяет использовать в дочерних классах функционал родительского класса и, в случае необходимости, дополнять его

- В) механизм, который представляет собой способность к изменению функционала, унаследованного от базового класса
- Г) копирование внутренних данных класса

Вопрос 33. Полиморфизм – это...

А) механизм, который объединяет данные и методы, манипулирующие этими данными, и защищает и то и другое от внешнего вмешательства или неправильного использования

Б) механизм, который позволяет использовать в дочерних классах функционал родительского класса и, в случае необходимости, дополнять его

В) механизм, который представляет собой способность к изменению функционала, унаследованного от базового класса

Г) механизм, позволяющий использовать один и тот же интерфейс для общего класса действий

Вопрос 34. Что является центральным понятием ООП?

- А) структура
- Б) класс
- В) переменная
- Г) функция

Вопрос 35. Класс является

- А) представителем объекта
- Б) описанием структуры объекта и методов работы с ним
- В) частью объекта
- Г) реализатором объекта

Вопрос 36. Методы - это

- А) переменные, принадлежащие классу или экземпляру класса
- Б) процедуры и функции класса
- В) синтаксическая надстройка, позволяющая осуществлять в форме вызовов функции
- Г) специальные математические алгоритмы

Вопрос 37. Абстрактный тип данных – это...

- А) пользовательский тип данных с полностью скрытой (инкапсулированной) внутренней структурой
- Б) тип данных, который может принимать как целочисленные значения, так и символьные значения

- В) тип данных, который может принимать как логические, так и строковые значения
- Г) неопределенный тип данных, который может принимать любые значения

Вопрос 38. Класс характеризуется ...

- А) множеством значений, которые могут принимать объекты класса
- Б) множеством функций, задающих операции над объектами
- В) множеством значений, которые могут принимать объекты класса, и множеством функций, задающих операции над объектами
- Г) структурированностью данных по типу значений

Вопрос 39. Синтаксическое описание класса в соответствии с правилами C++ выглядит следующим образом:

- А) **struct** *Имя_класса* { *определение_членов_класса* };
- Б) **class** *Имя_класса* { *определение_членов_класса* };
- В) **class** *Имя_класса* /*определение_членов_класса* /;
- Г) **struct** *Имя_класса* / *определение_членов_класса* /;

Вопрос 40. Чем обеспечивается принцип инкапсуляции?

- А) вводом в класс ключевых переменных
- Б) вводом в класс ключевых методов
- В) вводом в класс сигналов
- Г) вводом в класс областей доступа

Вопрос 41. Что из нижеперечисленного не относится к областям доступа в классе?

- А) private
- Б) public
- В) main
- Г) protected

Вопрос 42. Данные и методы, расположенные в области private ...

- А) доступны только внутри класса
- Б) доступны как внутри, так и извне класса
- В) доступны классам-наследникам
- Г) доступны как внутри, так и наследникам класса

Вопрос 43. Данные и методы, расположенные в области public ...

- А) доступны только внутри класса
- Б) доступны как внутри так и извне класса

- В) доступны только классам-наследникам
- Г) доступны только извне класса

Вопрос 44. Данные и методы, расположенные в области protected ...

- А) доступны только внутри класса
- Б) доступны как внутри так и извне класса
- В) доступны как классам-наследникам, так и внутри класса
- Г) доступны только извне класса

Вопрос 45. В какой последовательности должны располагаться области доступа при описании класса?

- А) private – public – protected
- Б) public – private – protected
- В) protected – private – public
- Г) порядок описания областей доступа не имеет значения

Вопрос 46. К какой области доступа будут относиться методы в описании класса:

```
class test
{
    void function1();
    void function2();
}
```

- А) оба метода будут относиться к области видимости public
- Б) оба метода будут относиться к области видимости private
- В) оба метода будут относиться к области видимости protected
- Г) такое описание недопустимо по правилам C++

Вопрос 47. По правилам C++ определение метода класса вне класса осуществляется следующим образом (имя класса X, метод int func1(int a)):

- А) X: int func1(int a)
- Б) X; int func1(int a)
- В) int X:func1(int a)
- Г) int X::func1(int a)

Вопрос 48. Указатель this – это ...

- А) указатель на конструктор данного класса
- Б) указатель на деструктор данного класса
- В) указатель на текущий объект
- Г) указатель на основной метод класса

Вопрос 49. Указатель – это...

- А) переменная, содержащая адрес другой переменной
- Б) константа, содержащая адрес другой переменной
- В) переменная, содержащая адрес конструктора класса
- Г) переменная, содержащая адрес деструктора класса

Вопрос 50. По правилам C++ указатель объявляется следующим образом:

- А) `int ptr;`
- Б) `int *ptr;`
- В) `int &ptr;`
- Г) `*int ptr;`

Вопрос 51. Для взятия адреса переменной `int a` необходимо выполнить команду:

- А) `*a;`
- Б) `$a;`
- В) `&a;`
- Г) `@a;`

Вопрос 52. Что будет выведено на экран в результате выполнения программы

```
#include <iostream>
```

```
int main ()  
{  
    int a = 5;  
    int *ptr = &a;  
    std::cout<<ptr<<std::endl;  
}
```

- А) 5
- Б) 0
- В) 25
- Г) адрес переменной a

Вопрос 53. Что будет выведено на экран в результате выполнения программы

```
#include <iostream>
```

```
int main ()
```

```
{  
    int a = 5;  
    int *ptr = &a;  
    std::cout<<*ptr<<std::endl;  
}
```

- A) 5
- Б) 0
- В) 25
- Г) адрес переменной *a*

Вопрос 54. Что будет выведено на экран в результате выполнения программы
`#include <iostream>`

```
int main ()  
{  
    int a = 5;  
    int *ptr = &a;  
    ++ ptr;  
    std::cout<<*ptr<<std::endl;  
}
```

- A) 5
- Б) значение ячейки памяти, имеющей адрес на 1 больше чем адрес переменной *a*
- В) 25
- Г) адрес переменной *a*

Вопрос 55. Что будет выведено на экран в результате выполнения программы
`#include <iostream>`

```
int main ()  
{  
    int a = 5;  
    int *ptr = &a;  
    ++ *ptr;  
    std::cout<<*ptr<<std::endl;  
}
```

- A) 5

Б) значение ячейки памяти, имеющей адрес на 1 больше чем адрес переменной *a*

В) 6

Г) адрес переменной *a*

Вопрос 56. Что будет выведено на экран в результате выполнения программы

```
#include <iostream>
```

```
int main ()  
{  
    int a = 5;  
    float *ptr = &a;  
    ++ *ptr;  
    std::cout<<*ptr<<std::endl;  
}
```

А) 5

Б) значение ячейки памяти, имеющей адрес на 1 больше, чем адрес переменной *a*

В) 6

Г) ничего, т.к. произойдет ошибка компиляции из-за несоответствия типа переменной и типа указателя

Вопрос 57. Что будет выведено на экран в результате выполнения программы

```
#include <iostream>
```

```
int main ()  
{  
    int a = 5;  
    int *ptr = &a;  
    ++ *ptr;  
    int *ptr2 = ptr;  
    std::cout<<*ptr2<<std::endl;  
}
```

А) 5

Б) значение ячейки памяти, имеющей адрес на 1 больше чем адрес переменной *a*

В) 6

Г) адрес переменной *a*

Вопрос 58. Что будет выведено на экран в результате выполнения программы

```
#include <iostream>
```

```
int main ()  
{  
    int a = 5;  
    int *ptr = &a;  
    ++ *ptr;  
    int *ptr2 = &ptr;  
    std::cout<<*ptr2<<std::endl;  
}
```

А) 5

Б) ничего, т.к. произойдет ошибка компиляции из-за несоответствия типа

В) 6

Г) адрес переменной *a*

Вопрос 59. Что будет выведено на экран в результате выполнения программы

```
#include <iostream>
```

```
int main ()  
{  
    int a = 5;  
    int *ptr = &a;  
    ++ *ptr;  
    int **ptr2 = &ptr;  
    std::cout<<**ptr2<<std::endl;  
}
```

А) 5

Б) ничего, т.к. произойдет ошибка компиляции из-за несоответствия типа

В) 6

Г) адрес переменной *a*

Вопрос 60. Что будет выведено на экран в результате выполнения программы

```
#include <iostream>
```

```
int main ()
{
    int a = 5;
    int *ptr = &a;
    ++ *ptr;
    int **ptr2 = &ptr;
    std::cout<<*ptr2<<std::endl;
}
```

- А) 5
- Б) ничего, т.к. произойдет ошибка компиляции из-за несоответствия типа
- В) 6
- Г) адрес указателя ptr

Вопрос 61. Какое из нижеперечисленных действий не разрешено для указателя?

- А) умножение
- Б) инкремент
- В) декремент
- Г) разыменовывание

Вопрос 62. Какое значение будет выведено на экран в процессе выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
int main ()
{
    uint16_t a = 1234;
    uint16_t *ptr = &a;
    uint8_t *ptr2 = (uint8_t*)ptr;
    printf("%d\r\n",*ptr2);
}
```

- А) 1234
- Б) 210
- В) 200
- Г) 128

Вопрос 63. Какое значение будет выведено на экран в процессе выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
int main ()
{
    uint16_t a = 234;
    uint16_t *ptr = &a;
    uint8_t *ptr2 = (uint8_t*)ptr;
    printf("%d\r\n", *ptr2);
}
```

- A) 1234
- Б) 210
- В) 234
- Г) 128

Вопрос 64. Какое значение будет выведено на экран в процессе выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
int main ()
{
    uint16_t a = 1234;
    uint16_t *ptr = &a;
    uint8_t *ptr2 = (uint8_t*)ptr;
    ptr2++;
    printf("%d\r\n", *ptr2);
}
```

- A) 1234
- Б) 210
- В) 234
- Г) 4

Вопрос 65. Какое действие выполняет конструктор копирования?

- A) Создает объект из любого объекта
- Б) Создает объект из объектов того же класса

В) Создает объект, который является указателем на скопированный объект

Г) Все предыдущие ответы неверны

Вопрос 66. Какое значение должен возвращать деструктор?

А) указатель на класс

Б) объект класса

В) статус кода, указывающий, правильно ли класс уничтожен

Г) деструкторы не возвращают значения

Вопрос 67. Для инициализации данных - членов класса используется

А) Конструктор

Б) Деструктор

В) Данные - члены автоматически инициализируются

Г) Все предыдущие ответы неверны

Вопрос 68. Какие функции должны иметь все классы?

А) Никаких

Б) Конструктор

В) Деструктор

Г) Конструктор и деструктор

Вопрос 69. Как по правилам C++ будет выглядеть имя конструктора для класса MyClass?

А) MyClass

Б) ~MyClass

В) :MyClass

Г) myclass

Вопрос 70. Как по правилам C++ будет выглядеть имя деструктора для класса MyClass?

А) MyClass

Б) ~MyClass

В) :MyClass

Г) myclass

Вопрос 71. По правилам C++ динамическое создание объекта на основе класса MyClass осуществляется следующим образом:

А) MyClass A;

Б) MyClass A = MyClass();

- В) MyClass *A=MyClass();
- Г) MyClass *A = new MyClass;

Вопрос 72. По правилам C++ статическое создание объекта на основе класса MyClass осуществляется следующим образом:

- А) MyClass A;
- Б) MyClass A = MyClass();
- В) MyClass *A=MyClass();
- Г) MyClass *A = new MyClass;

Вопрос 73. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class Print
{
public:
    Print()
    {
        std::cout<<"Class created ";
    }
    Print(int a)
    {
        std::cout<<"Class created, a="<<a<<" ";
    }

    ~Print()
    {
        std::cout<<"Class destroy ";
    }
};

int main ()
{
    Print a;
}
```

- А) Class created
- Б) Class created Class destroy

- В) Class created, a= Class destroy
- Г) Class destroy

Вопрос 74. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class Print
{
public:
    Print()
    {
        std::cout<<"Class created ";
    }
    Print(int a)
    {
        std::cout<<"Class created, a="<<a<<" ";
    }

    ~Print()
    {
        std::cout<<"Class destroy ";
    }
};
```

```
int main ()
{
    Print a(1);
}
```

- А) Class created
- Б) Class created Class destroy
- В) Class created, a=1 Class destroy
- Г) Class destroy

Вопрос 75. Что будет выведено на экран в результате выполнения программы?

```
class Print
{
public:
    Print(int a=0)
    {
```

```

        std::cout<<"Class created, a="<<a<<" ";
    }

    ~Print()
    {
        std::cout<<"Class destroy ";
    }
};

```

```

int main ()
{
    Print a;
}

```

- A) Class created
- Б) Class created Class destroy
- B) Class created, a=1 Class destroy
- Г) Class created, a=0 Class destroy

Вопрос 76. Что будет выведено на экран в результате выполнения программы?

```

#include <iostream>
#include <stdio.h>

```

```

class Print
{
public:
    void finc1 (int a)
    {
        std::cout<<"int a="<<a;
    }

    void finc1 (float a)
    {
        std::cout<<"float a="<<a;
    }
};

```

```

int main ()
{
    Print a;
    a.finc1(2);
}

```

```
}
```

- А) ничего не будет выведено на экран
- Б) произойдет ошибка компиляции из-за одинакового имени функций
- В) int a=2
- Г) float a=2

Вопрос 77. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class Print
{
public:
    void func1 (int a)
    {
        std::cout<<"int a="<<a;
    }

    void func1 (double a)
    {
        std::cout<<"double a="<<a;
    }
};
```

```
int main ()
{
    Print a;
    a.func1(2.9);
}
```

- А) ничего не будет выведено на экран
- Б) произойдет ошибка компиляции из-за одинакового имени функций
- В) int a=2
- Г) double a=2.9

Вопрос 78. Какой тип конструктора может генерироваться автоматически?

- А) конструктор умолчания
- Б) конструктор с одним параметром
- В) конструктор с несколькими параметрами

Г) конструкторы не могут генерироваться автоматически

Вопрос 79. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
```

```
#include <stdio.h>
```

```
class Print
```

```
{
```

```
    void fnc1 (int a)
```

```
    {
```

```
        std::cout<<"int a="<<a;
```

```
    }
```

```
public:
```

```
    void fnc1 (double a)
```

```
    {
```

```
        std::cout<<"double a="<<a;
```

```
    }
```

```
};
```

```
int main ()
```

```
{
```

```
    Print a;
```

```
    a.fnc1(2);
```

```
}
```

А) ничего не будет выведено на экран

Б) произойдет ошибка компиляции

В) int a=2

Г) double a=2.9

Вопрос 80. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
```

```
#include <stdio.h>
```

```
class Print
```

```
{
```

```
protected:
```

```
    void fnc1 (int a)
```

```
{
    std::cout<<"int a="<<a;
}

public:
    void func1 (double a)
    {
        std::cout<<"double a="<<a;
    }
};

int main ()
{
    Print a;
    a.func1(2);
}
```

- А) ничего не будет выведено на экран
- Б) int a=2
- В) произойдет ошибка компиляции
- Г) double a=2.9

Вопрос 81. Сколько деструкторов может иметь класс?

- А) 1
- Б) 2
- В) 5
- Г) сколько угодно, при условии, что входные параметры будут отличаться

Вопрос 82. Что такое наследование классов?

- А) Механизм, позволяющий расширять функционал родительского класса путем создания дочернего с дополнительными членами-данных и методами
- Б) Механизм сериализации класса, позволяющий преобразовывать в последовательность байтов любой сложный объект
- В) Механизм копирования данных из одного класса в другой
- Г) Механизм сжатия данных, хранящихся в классе

Вопрос 83. Каким образом влияет квалификатор доступа `private` на механизм наследования?

- А) Все методы и члены данных, расположенные в данном квалификаторе доступны как внутри дочернего класса, так и извне
- Б) Все методы и члены данных, расположенные в данном квалификаторе, недоступны для дочернего класса
- В) Все методы и члены данных, расположенные в данном квалификаторе, доступны для внутреннего использования в дочернем классе, но недоступны извне класса
- Г) Все методы и члены данных, расположенные в данном квалификаторе, недоступны для внутреннего использования дочерним классом, но доступны извне класса

Вопрос 84. Каким образом влияет квалификатор доступа `public` на механизм наследования?

- А) Все методы и члены данных, расположенные в данном квалификаторе доступны как внутри дочернего класса, так и извне
- Б) Все методы и члены данных, расположенные в данном квалификаторе, недоступны для дочернего класса
- В) Все методы и члены данных, расположенные в данном квалификаторе, доступны для внутреннего использования в дочернем классе, но недоступны извне класса
- Г) Все методы и члены данных, расположенные в данном квалификаторе, недоступны для внутреннего использования дочерним классом, но доступны извне класса

Вопрос 85. Каким образом влияет квалификатор доступа `protected` на механизм наследования?

- А) Все методы и члены данных, расположенные в данном квалификаторе, доступны как внутри дочернего класса, так и извне
- Б) Все методы и члены данных, расположенные в данном квалификаторе, недоступны для дочернего класса

- В) Все методы и члены данных, расположенные в данном квалификаторе, доступны для внутреннего использования в дочернем классе, но недоступны извне класса
- Г) Все методы и члены данных, расположенные в данном квалификаторе, недоступны для внутреннего использования дочерним классом, но доступны извне класса

Вопрос 86. Как по правилам C++ объявить класс В который является наследником класса А?

- А) class A : B
- Б) class A; B
- В) class B; A
- Г) class B : A

Вопрос 87. Какой из методов приведенного класса не будет доступен дочернему при применении механизма наследования?

```
Class A
{
    void Finc1();
public:
    void Func2();
    void Func3();
protected:
    void Func4();
}
```

- А) Finc1()
- Б) Func2()
- В) Func3()
- Г) Func4()

Вопрос 88 Какой из методов приведенного класса не будет доступен дочернему при применении механизма наследования?

```
Class A
{
public:
    void Finc1();
    void Func2();
protected:
    void Func3();
private:
    void Func4();
}
```


- A) Func1()
- Б) Func2()
- В) Func3()
- Г) Func4()

Вопрос 89. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
protected:
    void func1 (int a)
    {
        std::cout<<"int a="<<a;
    }

    void func1 (double a)
    {
        std::cout<<"double a="<<a;
    }
};
```

```
class B :A
{
public:
    void func1 (int a)
    {
        std::cout<<"class B ";
        A::func1(a);
    }

    void func1 (double a)
    {
        std::cout<<"class B ";
        A::func1(a);
    }
};
```

```
int main ()
{
    B b;
```

```
b.func1(2);  
}  
A) class B int a=2  
Б) class B double a=2  
B) int a=2  
Г) double a=2
```

Вопрос 90. Что будет выведено на экран после выполнения программы?

```
#include <iostream>  
#include <stdio.h>  
  
class A  
{  
public:  
    void func1 (int a)  
    {  
        std::cout<<"int a="<<a;  
    }  
  
    void func1 (double a)  
    {  
        std::cout<<"double a="<<a;  
    }  
};  
  
class B :A  
{  
public:  
    void func1 (int a)  
    {  
        std::cout<<"class B ";  
        A::func1(a);  
    }  
  
    void func1 (double a)  
    {  
        std::cout<<"class B ";  
        A::func1(a);  
    }  
};  
  
int main ()
```

```
{  
    A b;  
    b.func1(2);  
}
```

- A) class B int a=2
- Б) class B double a=2
- В) int a=2
- Г) double a=2

Вопрос 91. Что будет выведено на экран после выполнения программы?

```
#include <iostream>  
#include <stdio.h>
```

```
class A  
{  
protected:  
    void func1 (int a)  
    {  
        std::cout<<"int a="<<a;  
    }  
  
    void func1 (double a)  
    {  
        std::cout<<"double a="<<a;  
    }  
};
```

```
class B :A  
{  
public:  
    void func1 (int a)  
    {  
        std::cout<<"class B ";  
        A::func1(a);  
    }  
  
    void func1 (double a)  
    {  
        std::cout<<"class B ";  
        A::func1(a);  
    }  
};
```

```
int main ()
{
    A b;
    b.func1(3.14);
}
A) class B int a=3
Б) class B double a=3.14
B) int a=3.14
Г) double a=3.14
```

Вопрос 92. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>

class A
{
protected:
    void func1 (int a)
    {
        std::cout<<"int a="<<a;
    }

    void func1 (double a)
    {
        std::cout<<"double a="<<a;
    }
};

class B :A
{
public:
    void func1 (int a)
    {
        std::cout<<"class B ";
        A::func1(a);
    }

    void func1 (double a)
    {
        std::cout<<"class B ";
        A::func1(a);
    }
};
```

```
    }  
};  
  
int main ()  
{  
    B b;  
    b.func1(3.14);  
}
```

- A) class B int a=3
- Б) class B double a=3.14
- В) int a=3.14
- Г) double a=3.14

Вопрос 93. Что будет выведено на экран после выполнения программы?

```
#include <iostream>  
#include <stdio.h>  
  
class A  
{  
  
public:  
    A()  
    {  
        std::cout<<" Create A ";  
    }  
};  
  
class B :A  
{  
public:  
    B()  
    {  
        std::cout<<" Create B ";  
    }  
};  
  
int main ()  
{  
    B b;  
}
```

- A) Create A
- Б) Create B
- В) Create A Create B
- Г) Create B Create A

Вопрос 94. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>

class A
{
public:
    A()
    {
        std::cout<<" Create A ";
    }
};

class B :A
{
public:
    B()
    {
        std::cout<<" Create B ";
    }
};

int main ()
{
    A b;
}
```

- A) Create A
- Б) Create B
- В) Create A Create B
- Г) Create B Create A

Вопрос95. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
public:
    ~A()
    {
        std::cout<<" destroy A ";
    }
};
```

```
class B :A
{
public:
    ~B()
    {
        std::cout<<" destroy B ";
    }
};
```

```
int main ()
{
    B b;
}
```

- A) destroy A destroy B
- Б) destroy B destroy A
- В) destroy A
- Г) destroy B

Вопрос 96. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
public:
    ~A()
    {
        std::cout<<" destroy A ";
    }
}
```

```
};  
  
class B :A  
{  
public:  
    ~B()  
    {  
        std::cout<<" destroy B ";  
    }  
};
```

```
int main ()  
{  
    A b;  
}
```

- A) destroy A destroy B
- Б) destroy B destroy A
- В) destroy A
- Г) destroy B

Вопрос 97. Что будет выведено на экран после выполнения программы?

```
#include <iostream>  
#include <stdio.h>  
  
class A  
{  
  
public:  
    A()  
    {  
        std::cout<<" create A ";  
    }  
  
    ~A()  
    {  
        std::cout<<" destroy A ";  
    }  
};  
  
class B :A
```



```

{
public:
    B()
    {
        std::cout<<" create B ";
    }

    ~B()
    {
        std::cout<<" destroy B ";
    }
};

int main ()
{
    B b;
}

```

- A) create A create B destroy A destroy B
- Б) create A destroy A create B destroy B
- В) create B destroy B create A destroy A
- Г) create A create B destroy B destroy A

Вопрос 98. Что будет выведено на экран после выполнения программы?

```

#include <iostream>
#include <stdio.h>

class A
{
public:
    A()
    {
        std::cout<<" create A ";
    }

    ~A()
    {
        std::cout<<" destroy A ";
    }
};

```

```

class B :A
{
public:
    B()
    {
        std::cout<<" create B ";
    }

    ~B()
    {
        std::cout<<" destroy B ";
    }
};

int main ()
{
    A b;
}

```

- A) create A destroy A
- Б) create A destroy A create B destroy B
- В) create B destroy B
- Г) create A create B destroy B destroy A

Вопрос 99. В какой последовательности вызываются конструкторы родительского и дочернего класса при применении механизма наследования?

- A) сначала вызывается конструктор дочернего класса, затем конструктор родительского
- Б) сначала вызывается конструктор родительского класса, а затем дочернего
- В) вызывается конструктор только родительского класса
- Г) вызывается конструктор только дочернего класса

Вопрос 100. В какой последовательности вызываются деструкторы родительского и дочернего класса при применении механизма наследования?

- A) сначала вызывается деструктор родительского класса, а затем деструктор дочернего класса

- Б) вызывается только деструктор родительского класса
- В) сначала вызывается деструктор дочернего класса, а затем деструктор родительского класса
- Г) вызывается только деструктор дочернего класса

Вопрос 101. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
```

```
public:
```

```
    A()
    {
        std::cout<<" create A ";
    }

```

```
    A(int a)
    {
        std::cout<<" create A a="<<a;
    }

```

```
    ~A()
    {
        std::cout<<" destroy A ";
    }

```

```
};
```

```
class B :A
```

```
{
```

```
public:
```

```
    B()
    {
        std::cout<<" create B ";
    }

```

```
    B(int b)
    {
        std::cout<<" create B b="<<b;
    }

```

```

~B()
{
    std::cout<<" destroy B ";
}
};

```

```

int main ()
{
    B b(2);
}

```

- A) create A create B b=2 destroy B destroy A
- Б) create A create B destroy B destroy A
- B) create A a=2 create B b=2 destroy B destroy A
- Г) create A create B destroy B b=2 destroy A b=2

Вопрос 102. Что будет выведено на экран после выполнения программы?

```

#include <iostream>
#include <stdio.h>

```

```

class A
{
public:
    A()
    {
        std::cout<<" create A ";
    }

    A(int a)
    {
        std::cout<<" create A a="<<a;
    }

    ~A()
    {
        std::cout<<" destroy A ";
    }
};

```

```

class B :A
{
public:
    B() : A()
    {
        std::cout<<" create B ";
    }

    B(int b) : A(b)
    {
        std::cout<<" create B b="<<b;
    }

    ~B()
    {
        std::cout<<" destroy B ";
    }
};

```

```

int main ()
{
    B b(2);
}

```

- A) create A create B b=2 destroy B destroy A
- Б) create A create B destroy B destroy A
- B) create A a=2 create B b=2 destroy B destroy A
- Г) create A create B destroy B b=2 destroy A b=2

Вопрос 103. Что будет выведено на экран после выполнения программы?

```

#include <iostream>
#include <stdio.h>

```

```

class A
{
public:
    A()
    {

```

```

    std::cout<<" create A ";
}

A(int a)
{
    std::cout<<" create A a="<<a;
}

~A()
{
    std::cout<<" destroy A ";
}
};

```

```

class B :A
{
public:
    B() : A()
    {
        std::cout<<" create B ";
    }

    B(int b) : A()
    {
        std::cout<<" create B b="<<b;
    }

    ~B()
    {
        std::cout<<" destroy B ";
    }
};

```

```

int main ()
{
    B b(2);
}

```

- A) create A create B b=2 destroy B destroy A
- B) create A create B destroy B destroy A
- B) create A a=2 create B b=2 destroy B destroy A

Г) create A create B destroy B b=2 destroy A b=2

Вопрос 104. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
```

```
#include <stdio.h>
```

```
class A
```

```
{
```

```
public:
```

```
    A()
```

```
    {
```

```
        std::cout<<" create A ";
```

```
    }
```

```
    A(int a)
```

```
    {
```

```
        std::cout<<" create A a="<<a;
```

```
    }
```

```
    ~A()
```

```
    {
```

```
        std::cout<<" destroy A ";
```

```
    }
```

```
};
```

```
class B :A
```

```
{
```

```
public:
```

```
    B() : A()
```

```
    {
```

```
        std::cout<<" create B ";
```

```
    }
```

```
    B(int b) : A(4)
```

```
    {
```

```
        std::cout<<" create B b="<<b;
```

```
    }
```

```
    ~B()
```

```
    {
```

```
        std::cout<<" destroy B ";
    }
};
```

```
int main ()
{
    B b(2);
}
```

- A) create A=4 create B b=2 destroy B destroy A
- Б) create A create B destroy B destroy A
- В) create A a=2 create B b=2 destroy B destroy A
- Г) create A create B destroy B b=2 destroy A b=2

Вопрос 105. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
```

```
public:
```

```
    A()
```

```
    {
```

```
        std::cout<<" create A ";
```

```
    }
```

```
    A(int a)
```

```
    {
```

```
        std::cout<<" create A a="<<a;
```

```
    }
```

```
    ~A()
```

```
    {
```

```
        std::cout<<" destroy A ";
```

```
    }
```

```
};
```

```
class B :A
```

```
{
```



```

public:
    B() : A()
    {
        std::cout<<" create B ";
    }

    B(int b) : A(4)
    {
        std::cout<<" create B b="<<b;
    }

    ~B()
    {
        std::cout<<" destroy B ";
    }
};

```

```

int main ()
{
    A b(2);
}

```

- A) create A=4 create B b=2 destroy B destroy A
- Б) create A create B destroy B destroy A
- В) create A a=2 destroy A
- Г) create A create B destroy B b=2 destroy A b=2

Вопрос 106. По правилам наследования C++ открытые (public) методы родительского класса не будут доступны из объекта типа дочернего класса в случае (где А – родительский класс, В - дочерний):

- A) class B : A
- Б) class B : public A
- В) class B : private A
- Г) class A : private B

Вопрос 107. По правилам наследования C++ открытые (public) методы родительского класса не будут доступны из объекта типа дочернего класса, но при этом будут доступны наследникам дочернего класса (где А – родительский класс, В - дочерний):

- A) class B : A
- Б) class B : public A

- В) class B : private A
- Г) class B : protected A

Вопрос 108. По правилам наследования C++ открытые (public) методы родительского класса будут доступны из объекта типа дочернего класса в случае (где А – родительский класс, В - дочерний):

- А) class B : A
- Б) class A : protected B
- В) class B : private A
- Г) class B : protected A

Вопрос 109. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
public:
    void func1()
    {
        std::cout<<" func1 ";
    }

    void func2()
    {
        std::cout<<" func2 ";
    }
};
```

```
class B : public A
{
public:
    void func3()
    {
        std::cout<<" func3 ";
        A::func1();
    }
};
```

```
int main ()
```

```
{
    B b;
    b.func2();
    b.func3();
}
```

- А) Ничего, т.к. произойдет ошибка компиляции
- Б) func2 func3 func1
- В) func1 func3 func2
- Г) func2 func3

Вопрос 110. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
public:
    void func1()
    {
        std::cout<<" func1 ";
    }

    void func2()
    {
        std::cout<<" func2 ";
    }
};
```

```
class B : public A
{
public:
    void func3()
    {
        std::cout<<" func3 ";
        A::func2();
    }
};
```

```
int main ()
```

```
{
    B b;
    b.func2();
    b.func3();
}
```

- А) Ничего, т.к. произойдет ошибка компиляции
- Б) func2 func3 func2
- В) func1 func3 func2
- Г) func2 func3

Вопрос 111. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
public:
    void func1()
    {
        std::cout<<" func1 ";
    }

    void func2()
    {
        std::cout<<" func2 ";
    }
};
```

```
class B : private A
{
public:
    void func3()
    {
        std::cout<<" func3 ";
        A::func1();
    }
};
```

```
int main ()
```

```
{
    B b;
    b.func2();
    b.func3();
}
```

- А) Ничего, т.к. произойдет ошибка компиляции
- Б) func2 func3 func1
- В) func1 func3 func2
- Г) func2 func3

Вопрос 112. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
public:
    void func1()
    {
        std::cout<<" func1 ";
    }

    void func2()
    {
        std::cout<<" func2 ";
    }
};
```

```
class B : protected A
{
public:
    void func3()
    {
        std::cout<<" func3 ";
        A::func1();
    }
};
```

```
int main ()
```

```
{  
    B b;  
    b.func3();  
}
```

- A) Ничего, т.к. произойдет ошибка компиляции
- Б) func3 func1
- В) func1 func3 func2
- Г) func2 func3

Вопрос 113. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>  
#include <stdio.h>
```

```
class A  
{  
public:  
    void func1()  
    {  
        std::cout<<" func1 ";  
    }  
  
    void func2()  
    {  
        std::cout<<" func2 ";  
    }  
};
```

```
class B : protected A  
{  
public:  
    void func3()  
    {  
        std::cout<<" func3 ";  
        A::func1();  
    }  
};
```

```
class C : B  
{  
public:
```

```
void func4()
{
    B::func1();
}
};
```

```
int main ()
{
    C c;
    c.func4();
}
```

- A) Ничего, т.к. произойдет ошибка компиляции
- Б) func4 func1
- В) func1
- Г) func2 func3

Вопрос 114. Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
class A
{
public:
    void func1()
    {
        std::cout<<" func1 ";
    }

    void func2()
    {
        std::cout<<" func2 ";
    }
};
```

```
class B : protected A
{
public:
    void func3()
    {
        std::cout<<" func3 ";
    }
};
```

```

        A::func1();
    };
};

class C : B
{
public:
    void func4()
    {
        B::func1();
    }
};

int main ()
{
    C c;
    c.func1();
}

```

- А) Ничего, т.к. произойдет ошибка компиляции
- Б) func4 func1
- В) func1
- Г) func2 func3

Вопрос 115. Что такое параметрический полиморфизм?

- А) это механизм, позволяющий применять один и тот же алгоритм к разным типам данных
- Б) это механизм, позволяющий запускать один и тот же код на разных ОС
- В) это механизм, позволяющий запускать один и тот же код на разных архитектурах
- Г) это механизм передачи данных через глобальные сети

Вопрос 116. Что такое шаблон в C++?

- А) это свод правил по оформлению исходного кода
- Б) это средство языка C++, предназначенное для кодирования обобщённых алгоритмов, без привязки к некоторым параметрам (например, типам данных, размерам буферов, значениям по умолчанию).
- В) это класс, для работы с текстовыми документами
- Г) это класс, для работы с графическим материалом

Вопрос 117. Каким ключевым словом обозначается шаблон в C++?

- А) define
- Б) struct
- В) template
- Г) function

Вопрос 118. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template< typename T >
T sum (T a, T b)
{
    return a+b;
}
```

```
int main ()
{
    int a =2;
    int b = 5;
    int c = sum (a,b);

    std::cout<<"C="<<c<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
- Б) C=
- В) C=3
- Г) C=7

Вопрос 119. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template< typename T >
T sum (T a, T b)
{
    return a+b;
}
```

```
int main ()
{
    float a =2.5;
    int b = 5;
    float c = sum (a,b);

    std::cout<<"C="<<c<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
Б) C=2.5
В) C=7
Г) C=7.5

Вопрос 120. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template< typename T >
T sum (T a, T b)
{
    return a+b;
}
```

```
int main ()
{
    float a =2.5;
    float b = 5;
    float c = sum (a,b);

    std::cout<<"C="<<c<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
Б) C=2.5
В) C=7
Г) C=7.5

Вопрос 121. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template< typename T >
T sum (T a, T b)
{
    return a+b;
}
```

```
int main ()
{
    float a =2.5;
    float b = 5;
    int c = sum (a,b);

    std::cout<<"C="<<c<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
- Б) C=2.5
- В) C=7
- Г) C=7.5

Вопрос 122. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template< typename T,typename D >
T sum (T a, D b)
{
    return a+b;
}
```

```
int main ()
{
    float a =2.5;
    int b = 5;
    float c = sum (a,b);
}
```

```
std::cout<<"C="<<c<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
- Б) C=2.5
- В) C=7
- Г) C=7.5

Вопрос 123. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template< typename T,typename D >
T sum (T a, D b)
{
    return a+b;
}
```

```
int main ()
{
    float a =12.5;
    float b = 5.3;
    float c = sum (a,b);

    std::cout<<"C="<<c<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
- Б) C=17.8
- В) C=2.5
- Г) C=7.5

Вопрос 124. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template< typename T,typename D >
```

```
T sum (T a, D b)
{
    return a+b;
}
```

```
int main ()
{
    float a =12.5;
    float b = 5.3;
    int c = sum (a,b);

    std::cout<<"C="<<c<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
Б) C=17.8
В) C=2.5
Г) C=17

Вопрос 125. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template <class Type>
class ABC {
public:
    void setA (Type a_) {a=a_;}

    void setB (Type b_) {b=b_;}

    Type sum()
    {return a+b;}
};
```

```
private:
    Type a;
    Type b;
};
```

```
int main ()
{
```

```
ABC<float> abc;
abc.setA(15);
abc.setB(35);

std::cout<<"sum="<<abc.sum()<<std::endl;
}
```

А) sum=50

Б) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов

В) 40

Г) sum=17

Вопрос 126. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template <class Type>
class ABC {
public:
    void setA (Type a_) {a=a_;}

    void setB (Type b_) {b=b_;}

    Type sum()
    {return a+b;}

private:
    Type a;
    Type b;
};

int main ()
{
    ABC<int> abc;
    abc.setA(5);
    abc.setB(3);

    std::cout<<"sum="<<abc.sum()<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
Б) sum=8
В) 8
Г) sum=9

Вопрос 127. Что будет выведено на экран после выполнения программы?

```
#include <iostream>
#include <stdio.h>
```

```
template <class Type>
class ABC {
public:
    void setA (Type a_) {a=a_;}

    void setB (Type b_) {b=b_;}

    Type sum()
    {return a+b;}
};
```

```
private:
    Type a;
    Type b;
};
```

```
int main ()
{
    ABC<int> abc;
    int a = 10;
    int b = 13;
    abc.setA(a);
    abc.setB(b);

    std::cout<<"sum="<<abc.sum()<<std::endl;
}
```

- А) Ничего, т.к. произойдет ошибка компиляции из-за несоответствия типов
Б) 23
В) sum-23
Г) sum=10

Вопрос 128. Какой класс C++ используется для связи файла с потоком вывода?

- A) ofstream
- Б) ifstream
- В) mutex
- Г) thread

Вопрос 129. Какой класс C++ используется для связи файла с потоком ввода?

- A) ofstream
- Б) ifstream
- В) mutex
- Г) thread

Вопрос 130. Какой класс Qt используется для доступа к файлам?

- A) QWidget
- Б) QString
- В) QFile
- Г) QLineEdit

Вопрос 131. Какой метод класса QFile производит копирование файлов?

- A) copy
- Б) exists
- В) fileName
- Г) open

Вопрос 132. Какой метод класса QFile производит проверку на существование файла?

- A) copy
- Б) exists
- В) fileName
- Г) open

Вопрос 133. Какой метод класса QFile возвращает имя текущего файла?

- A) copy
- Б) exists
- В) fileName
- Г) open

Вопрос 134. Какой метод класса QFile производит открытие файла?

- A) copy
- Б) exists
- В) fileName
- Г) open

Вопрос 135. Какой метод класса QFile производит удаление файла?

- A) remove
- Б) rename
- В) setFileName
- Г) read

Вопрос 136. Какой метод класса QFile производит переименование файла?

- A) remove
- Б) rename
- В) setFileName
- Г) read

Вопрос 137. Какой метод класса QFile устанавливает имя текущего файла?

- A) remove
- Б) rename
- В) setFileName
- Г) read

Вопрос 138. Какой метод класса QFile производит чтение данных из файла?

- A) remove
- Б) rename
- В) setFileName
- Г) read

Вопрос 139. Какой метод класса QFile производит считывание всех данных из файла?

- A) readAll
- Б) write
- В) atEnd
- Г) size

Вопрос 140. Какой метод класса QFile производит запись данных в файл?

- А) readAll
- Б) write
- В) atEnd
- Г) size

Вопрос 141. Какой метод класса QFile сигнализирует о достижении конца файла?

- А) readAll
- Б) write
- В) atEnd
- Г) size

Вопрос 142. Какой метод класса QFile возвращает размер текущего файла?

- А) readAll
- Б) write
- В) atEnd
- Г) size

Вопрос 143. Какой класс Qt предназначен для чтения текстовых данных через поток?

- А) QTextStream
- Б) QDataStream
- В) QLineEdit
- Г) QPushButton

Вопрос 144. Какой класс Qt предназначен для чтения двоичных данных через поток?

- А) QTextStream
- Б) QDataStream
- В) QLineEdit
- Г) QPushButton

Вопрос 145. Какой метод класса QTextStream сигнализирует об окончании данных?

- А) atEnd
- Б) autoDetectUnicode
- В) codec
- Г) pos

Вопрос 146. Какой метод класса QTextStream сигнализирует о наличии юникода в текстовых данных?

- А) atEnd
- Б) autoDetectUnicode
- В) codec
- Г) pos

Вопрос 247. Какой метод класса QTextStream возвращает указатель на текущий текстовый кодек?

- А) atEnd
- Б) autoDetectUnicode
- В) codec
- Г) pos

Вопрос 148. Какой метод класса QTextStream возвращает текущую позицию в потоке?

- А) atEnd
- Б) autoDetectUnicode
- В) codec
- Г) pos

Вопрос 149. Какой метод класса QDataStream сигнализирует об окончании данных?

- А) atEnd
- Б) readBytes
- В) writeBytes
- Г) abortTransaction

Вопрос 150. Какой метод класса QDataStream считывает данные в буфер?

- А) atEnd
- Б) readBytes
- В) writeBytes
- Г) abortTransaction

Вопрос 151. Какой метод класса QDataStream записывает данные в буфер ?

- А) atEnd
- Б) readBytes
- В) writeBytes
- Г) abortTransaction

Вопрос 152. Какой метод класса QDataStream отменяет транзакцию?

- A) atEnd
- Б) readBytes
- В) writeBytes
- Г) abortTransaction

Вопрос 153. Что такое исключение?

- A) это аномальное поведение во время выполнения, которое программа может обнаружить
- Б) это процесс создания объекта
- В) это процесс удаления объекта
- Г) это процесс объявления указателя

Вопрос 154. В какой блок заключатся участок кода, в котором может произойти исключительная ситуация?

- A) then
- Б) try
- В) throw
- Г) catch

Вопрос 155. Вызов какой команды генерирует исключительную ситуацию?

- A) then
- Б) try
- В) throw
- Г)

Вопрос 156. В каком блоке располагаются обработчики исключительных ситуаций?

- A) then
- Б) try
- В) throw
- Г) catch

Вопрос 157. Кой класс является базовым для всех (включая и неграфические) классов Qt?

- A) QObject
- Б) QWidget
- В) QLineEdit
- Г) QLabel

Вопрос 158. Какой класс является базовым для все графических компонентов Qt?

- А) QObject
- Б) QWidget
- В) QLineEdit
- Г) QLabel

Вопрос 159. Что такое Qt?

- А) Компилятор C++
- Б) Отладчик C++
- В) Кроссплатформенный инструментарий разработки ПО на языке программирования C++
- Г) Базовый класс графического интерфейса

Вопрос 160. Какие операционные системы поддерживает Qt?

- А) Windows
- Б) Linux
- В) Mac OS
- Г) Все перечисленные

Вопрос 161. Какие мобильные операционные системы поддерживает Qt?

- А) Android
- Б) Windows Phone
- В) iOS
- Г) Все перечисленные

Вопрос 162. Какой язык программирования не позволяет применять библиотеки Qt при разработке ПО?

- А) Ассемблер
- Б) C++
- В) Python
- Г) Ruby

Вопрос 163. Что такое Qt Assistant?

- А) Компилятор C++
- Б) Интегрированная среда разработки с поддержкой Qt
- В) Встроенный справочник по классам Qt
- Г) Отладчик с поддержкой Qt

Вопрос 164. Что такое GDB?

- А) Компилятор C++

- Б) Интегрированная среда разработки с поддержкой Qt
- В) Встроенный справочник по классам Qt
- Г) Отладчик с поддержкой Qt

Вопрос 165. Что такое Qt Creator?

- А) Компилятор C++
- Б) Интегрированная среда разработки с поддержкой Qt
- В) Встроенный справочник по классам Qt
- Г) Отладчик с поддержкой Qt

Вопрос 166. Для чего предназначен механизм сигналов и слотов?

- А) Для поиска ошибок в исходном коде
- Б) Для передачи информации между различными объектами
- В) Для осуществления соединения по сети
- Г) Для отображения графического материала на мониторе ПК

Вопрос 167. Есть два объекта А и В. В объекте А присутствует сигнал `signalDoIt()`, а в объекте В слот `slotDoIt()`. Для их соединения необходимо вызвать функцию `connect` следующего вида:

- А) `connect(A, signalDoIt(),B, slotDoIt());`
- Б) `connect(*A,SIGNAL(signalDoIt()),*B, slotDoIt());`
- В) `connect(A,SIGNAL(signalDoIt()),B,SLOT(slotDoIt()));`
- Г) `connect(A, signalDoIt(),B,SLOT(slotDoIt()));`

Вопрос 168. Для чего предназначена функция `connect`?

- А) Для подключения локального диска
- Б) Для монтирования файловой системы
- В) Для подключения к локальной СУБД
- Г) Для соединения сигналов и слотов

Вопрос 169. Какой метод класса `QWidget` позволяет задавать заголовок окна?

- А) `setWindowTitle`
- Б) `windowTitle`
- В) `setWindowRole`
- Г) `setWhatsThis`

Вопрос 170. Какой метод класса `QWidget` позволяет ограничить максимальный размер окна?

- А) `setMaximumWidth`
- Б) `setMaximumSize`
- В) `setMinimumSize`

Г) setPalette

Вопрос 171. Какой метод класса QWidget позволяет задать фиксированный размер окна?

А) setMaximumWidth

Б) setMaximumSize

В) setFixedSize

Г) setPalette

Вопрос 172. Какой метод класса QWidget предназначен для установки Layout?

А) setFocus

Б) setGeometry

В) setLocale

Г) setLayout

Вопрос 173. Какие области видимости доступны для слотов?

А) все области, которые доступны и обычным методам класса

Б) только private

В) только protected

Г) только public

Вопрос 174. Какая функция используется для разъединения сигналов и слотов?

А) remove

Б) disconnect

В) delete

Г) unmount

Вопрос 175. Что такое Layouts?

А) классы для ввода данных

Б) классы для вывода данных

В) классы компоновки виджетов

Г) классы для построения графиков

Вопрос 176. Какой из классов Layouts применяется для горизонтального размещения виджетов?

А) все перечисленные

Б) QVBoxLayout

В) QGridLayout

Г) QHBoxLayout

Вопрос 177. Какой из классов Layouts применяется для вертикального размещения виджетов?

- А) все перечисленные
- Б) QVBoxLayout
- В) QGridLayout
- Г) QHBoxLayout

Вопрос 178. Какой из классов Layouts применяется для табличного размещения виджетов?

- А) все перечисленные
- Б) QVBoxLayout
- В) QGridLayout
- Г) QHBoxLayout

Вопрос 179. Какой метод QVBoxLayout позволяет добавить виджет?

- А) insertItem
- Б) addLayout
- В) addSpacing
- Г) addWidget

Вопрос 180. Какой метод QVBoxLayout позволяет добавить фактор растяжения?

- А) insertItem
- Б) addLayout
- В) addSpacing
- Г) addWidget

Вопрос 181. Какой метод QVBoxLayout позволяет добавить Layout?

- А) insertItem
- Б) addLayout
- В) addSpacing
- Г) addWidget

Вопрос 182. Какой метод QVBoxLayout позволяет вставить виджет на указанную позицию?

- А) insertItem
- Б) addLayout
- В) addSpacing
- Г) addWidget

Вопрос 183. Какой метод позволяет задать отступ виджетов от границы компоновки?

- A) setSpacing
- Б) setMargin
- В) setStretch
- Г) setStretchFactor

Вопрос 184. Какой метод позволяет задать расстояние между виджетами на компоновке?

- A) setMargin
- Б) setStretch
- В) setSpacing
- Г) setStretchFactor

Вопрос 185. Какое значение параметра Direction класса QBoxLayout позволяет задать горизонтальное расположение виджетов с порядком заполнения слева на право?

- A) LeftToRight
- Б) RightToLeft
- В) TopToBottom
- Г) BottomToTop

Вопрос 186. Какое значение параметра Direction класса QBoxLayout позволяет задать горизонтальное расположение виджетов с порядком заполнения справа налево?

- A) LeftToRight
- Б) RightToLeft
- В) TopToBottom
- Г) BottomToTop

Вопрос 187. Какое значение параметра Direction класса QBoxLayout позволяет задать вертикальное расположение виджетов с порядком заполнения сверху вниз?

- A) LeftToRight
- Б) RightToLeft
- В) TopToBottom
- Г) BottomToTop

Вопрос 188. Какое значение параметра Direction класса QBoxLayout позволяет задать вертикальное расположение виджетов с порядком заполнения снизу вверх?

- A) LeftToRight

- Б) RightToLeft
- В) TopToBottom
- Г) BottomToTop

Вопрос 189. Какой класс является классом кнопки?

- А) QPushButton
- Б) QLabel
- В) QLineEdit
- Г) QCheckBox

Вопрос 190. Какой класс предназначен для отображения простой текстовой информации без возможности редактирования?

- А) QPushButton
- Б) QLabel
- В) QLineEdit
- Г) QCheckBox

Вопрос 191. Какой класс является однострочным текстовым полем с возможностью ввода и редактирования текста?

- А) QPushButton
- Б) QLabel
- В) QLineEdit
- Г) QCheckBox

Вопрос 192. Какой метод класса QPushButton предназначен для задания текста на кнопке?

- А) setAutoExclusive
- Б) setCheckable
- В) setIcon
- Г) setText

Вопрос 193. Какой метод класса QPushButton позволяет добавить растровое изображение?

- А) setAutoExclusive
- Б) setCheckable
- В) setIcon
- Г) setText

Вопрос 194. Какой сигнал класса QPushButton сигнализирует о том, что кнопка была нажата?

- А) released
- Б) onClicked
- В) onPressed
- Г) pressed

Вопрос 195. Какой класс из перечисленных является виджетом флажка?

- А) QCheckBox
- Б) QLineEdit
- В) QTextEdit
- Г) QLcdNumber

Вопрос 196. Какой из перечисленных классов является переключателем?

- А) QGroupBox
- Б) QRadioButton
- В) QWidget
- Г) QPushButton

Вопрос 197. Какой из перечисленных классов предназначен для группировки виджетов?

- А) QRadioButton
- Б) QWidget
- В) QGroupBox
- Г) QPushButton

Вопрос 198. Какой метод класса QLineEdit позволяет установить текст?

- А) placeholderText
- Б) selectedText
- В) setFrame
- Г) setText

Вопрос 199. Какой сигнал класса QLineEdit сигнализирует о нажатии клавиши Enter?

- А) returnPressed
- Б) selectionChanged
- В) cursorPositionChanged
- Г) customContextMenuRequested

Вопрос 200. Какой метод класса QLineEdit устанавливает режим только для чтения?

- A) setText
- Б) setReadOnly
- В) setEchoMode
- Г) setFixedHeight

Вопрос 201. Какой из перечисленных классов является редактором текста?

- A) QLabel
- Б) QPushButton
- В) QTextEdit
- Г) QRadioButton

Вопрос 202. Какой из перечисленных методов класса QTextEdit устанавливает обычный текст?

- A) setHtml
- Б) selectAll
- В) text
- Г) setPlainText

Вопрос 203. Какой из перечисленных методов класса QTextEdit устанавливает html?

- A) setHtml
- Б) selectAll
- В) text
- Г) setPlainText

Вопрос 204. Какой из перечисленных методов класса QTextEdit копирует выделенный текст в буфер обмена?

- A) cut
- Б) copy
- В) clear
- Г) setHtml

Вопрос 205. Какой из перечисленных методов класса QTextEdit позволяет выделить весь текст?

- A) setHtml
- Б) selectAll
- В) text
- Г) setPlainText

Вопрос 206. Какой из перечисленных методов класса QTextEdit возвращает текст из редактора?

- А) setHtml
- Б) selectAll
- В) text
- Г) setPlainText

Вопрос 207. Какой из перечисленных классов является счетчиком?

- А) QTextEdit
- Б) QLineEdit
- В) QLabel
- Г) QSpinBox

Вопрос 208. С помощью какого метода в классе QSpinBox устанавливается диапазон возможных значений?

- А) setRange
- Б) setValue
- В) setPrefix
- Г) setSuffix

Вопрос 209. С помощью какого метода в классе QSpinBox устанавливается префикс?

- А) setRange
- Б) setValue
- В) setPrefix
- Г) setSuffix

Вопрос 210. С помощью какого метода в классе QSpinBox устанавливается суффикс?

- А) setRange
- Б) setValue
- В) setPrefix
- Г) setSuffix

Вопрос 211. С помощью какого метода в классе QSpinBox устанавливается текущее значение?

- А) setRange
- Б) setValue
- В) setPrefix
- Г) setSuffix

Вопрос 212. Какой из перечисленных классов является элементом ввода даты и времени?

- А) QSpinBox
- Б) QLineEdit
- В) QDateTimeEdit
- Г) QLabel

Вопрос 213. Помощью какого метода класса QDateTimeEdit можно установить дату и время?

- А) setDate
- Б) setTime
- В) setDisplayFormat
- Г) setDateTime

Вопрос 214. Какой из перечисленных классов является классом ползунка?

- А) QSlider
- Б) QDateTimeEdit
- В) QSpinBox
- Г) QLabel

Вопрос 215. Какой метод класса QSlider задает шаг рисования рисок?

- А) setTickPosition
- Б) setTickInterval
- В) setRange
- Г) setValue

Вопрос 216. Какой из перечисленных классов является полосой прокрутки?

- А) QSlider
- Б) QLabel
- В) QScrollBar
- Г) QSpinBox

Вопрос 217. Какой из перечисленных классов является виджетом списка?

- А) QSlider
- Б) QLabel
- В) QSpinBox
- Г) QListWidget

Вопрос 218. Какое расширение имеют динамические библиотеки в операционной системе Windows?

- А) .dll
- Б) .h
- В) .lib
- Г) .so

Вопрос 219. Чем характеризуется способ загрузки динамических библиотек при явной компоновке?

- А) Все экспортируемые функции динамической библиотеки на стадии сборки программы вносятся в ее код.
- Б) Все динамические библиотеки, относящиеся к программе, загружаются автоматически операционной системой при запуске приложения.
- В) Динамические библиотеки подключаются с помощью вызова специальной функции, при необходимости
- Г) Для запуска программы обязательно наличие всех используемых динамических библиотек.

Вопрос 220. Чем характеризуется способ загрузки динамических библиотек при неявной компоновке?

- А) Все экспортируемые функции динамической библиотеки на стадии сборки программы вносятся в ее код.
- Б) Все динамические библиотеки, относящиеся к программе, загружаются автоматически операционной системой при запуске приложения.
- В) Динамические библиотеки подключаются с помощью вызова специальной функции, при необходимости
- Г) Для запуска программы обязательно наличие всех используемых динамических библиотек.

Вопрос 221. Какое расширение обычно имеют динамические библиотеки в операционной системе Linux?

- А) .dll
- Б) .h
- В) .lib
- Г) .so

Вопрос 222. Какую последовательность команд необходимо выполнить для сборки динамической библиотеки в операционной системе Windows, если исходный код библиотеки содержится в файле world.c?

- А) сначала `g++ -c -fPIC world.c`, затем `g++ -shared -o libworld.dll world.o`
- Б) сначала `g++ -shared -o libworld.dll world.o` затем `g++ -c -fPIC world.c`
- В) `g++ -shared -o libworld.dll`
- Г) `g++ -c -fPIC world.c`

Вопрос 223. Какую последовательность команд необходимо выполнить для сборки динамической библиотеки в операционной системе Linux, если исходный код библиотеки содержится в файле `world.c`?

- А) сначала `g++ -shared -o libworld.dll world.o`, затем `g++ -c -fPIC world.c`
- Б) сначала `g++ -c -fPIC world.c` затем `g++ -shared -o libworld.so world.o`
- В) `g++ -shared -o libworld.dll`
- Г) `g++ -c -fPIC world.c`

Вопрос 224. Какую последовательность команд необходимо выполнить для сборки программы, использующую динамическую библиотеку, в операционной системе Windows, если исходный код программы содержится в файле `main.c`, а библиотека имеет имя `libworld.dll`?

- А) `g++ -c main.c`
- Б) сначала `g++ -o binary.exe main.o -L. -lworld -Wl,-rpath,.` затем `g++ -c main.c`
- В) сначала `g++ -c main.c`, затем `g++ -o binary.exe main.o -L. -lworld -Wl,-rpath,.`
- Г) `g++ -o binary.exe main.o -L. -lworld -Wl,-rpath,.`

Вопрос 225. Какую последовательность команд необходимо выполнить для сборки программы, использующую динамическую библиотеку, в операционной системе Linux, если исходный код программы содержится в файле `main.c`, а библиотека имеет имя `libworld.dll`?

- А) `g++ -c main.c`
- Б) сначала `g++ -o binary.exe main.o -L. -lworld -Wl,-rpath,.` затем `g++ -c main.c`
- В) `g++ -o binary.exe main.o -L. -lworld -Wl,-rpath,.`
- Г) сначала `g++ -c main.c` затем `g++ -o binary main.o -L. -lworld -Wl,-rpath,.`

Вопрос 226. Какой класс Qt предоставляет интерфейс для работы с динамическими библиотеками?

- А) `QLibrary`
- Б) `QLineEdit`

- В) QLabel
- Г) QApplication

Вопрос 227. Какой метод класса QLibrary возвращает имя используемой динамической библиотеки?

- А) errorString
- Б) fileName
- В) isLoading
- Г) load

Вопрос 228. Какой метод класса QLibrary позволяет определить, загружена ли динамическая библиотека?

- А) errorString
- Б) fileName
- В) isLoading
- Г) load

Вопрос 229. Какой метод класса QLibrary производит загрузку динамической библиотеки в оперативную память?

- А) errorString
- Б) fileName
- В) isLoading
- Г) load

Вопрос 230. Какой метод класса QLibrary возвращает текст ошибки в случае неудачной загрузки динамической библиотеки?

- А) errorString
- Б) fileName
- В) isLoading
- Г) load

Вопрос 14. Какой метод класса QLibrary возвращает указатель на функцию из динамической библиотеки?

- А) resolve
- Б) setFileName
- В) setFileNameAndVersion
- Г) unload

Вопрос 231. Какой метод класса QLibrary задает имя динамической библиотеки?

- А) resolve

- Б) `setFileName`
- В) `setFileNameAndVersion`
- Г) `unload`

Вопрос 232. Какой метод класса `QLibrary` задает имя динамической библиотеки и версию?

- А) `resolve`
- Б) `setFileName`
- В) `setFileNameAndVersion`
- Г) `unload`

Вопрос 233. Какой метод класса `QLibrary` выгружает динамическую библиотеку из оперативной памяти?

- А) `resolve`
- Б) `setFileName`
- В) `setFileNameAndVersion`
- Г) `unload`

Вопрос 234. Какой метод класса `QLibrary` позволяет проверить, является ли указанный файл динамической библиотекой?

- А) `isLibrary`
- Б) `setFileName`
- В) `setFileNameAndVersion`
- Г) `unload`

Вопрос 235. Что такое Qt Quick?

- А) Это набор технологий, предназначенных для создания анимированных, динамических, пользовательских интерфейсов нового поколения.
- Б) Это набор технологий, предназначенных для создания приложений с поддержкой сети.
- В) Это набор технологий для создания системных служб.
- Г) Это набор технологий для создания драйверов устройств.

Вопрос 236. Что такое QML?

- А) Класс поддержки сети.
- Б) Декларативный язык программирования, основанный на JavaScript, предназначенный для дизайна приложений, делающих основной упор на пользовательский интерфейс.
- В) Класс, реализующий интерфейс для управления системными службами.
- Г) Класс, реализующий интерфейс управления потоками.

Вопрос 237. Какой из нижеперечисленных компонентов является абстрактным базовым типом для всех элементов?

- А) item
- Б) Rectangle
- В) image
- Г) Borderimage

Вопрос 238. Какой из нижеперечисленных компонентов является прямоугольной областью с необязательным контуром?

- А) item
- Б) Rectangle
- В) image
- Г) Borderimage

Вопрос 239. Какой из нижеперечисленных компонентов предназначен для вывода растровых изображений?

- А) item
- Б) Rectangle
- В) image
- Г) Borderimage

Вопрос 240. Какой из нижеперечисленных компонентов является растровым изображением с контуром?

- А) item
- Б) Rectangle
- В) image
- Г) Borderimage

Вопрос 241. Какой из нижеперечисленных компонентов является списком?

- А) Listview
- Б) Gridview
- В) Text
- Г) Webview

Вопрос 242. Какой из нижеперечисленных компонентов является таблицей?

- А) Listview
- Б) Gridview
- В) Text
- Г) Webview

Вопрос 243. Какой из нижеперечисленных компонентов предназначен для отображения текста?

- А) Listview
- Б) Gridview
- В) Text
- Г) Webview

Вопрос 244. Какой из нижеперечисленных компонентов предназначен для отображения Web-содержания?

- А) Listview
- Б) Gridview
- В) Text
- Г) Webview

Вопрос 245. Какие из нижеперечисленных свойств относятся к позиционированию?

- А) x, y, position
- Б) width, height
- В) anchors
- Г) id, parent

Вопрос 246. Какие из нижеперечисленных свойств относятся к свойствам размеров элемента?

- А) x, y, position
- Б) width, height
- В) anchors
- Г) id, parent

Вопрос 247. Какие из нижеперечисленных свойств относятся к свойствам фиксации?

- А) x, y, position
- Б) width, height
- В) anchors
- Г) id, parent

Вопрос 248. Какие из нижеперечисленных свойств относятся к ссылкам на элементы?

- А) x, y, position
- Б) width, height
- В) anchors
- Г) id, parent

Вопрос 249. Какой из нижеперечисленных компонентов является кнопкой?

- А) Button
- Б) MenuBar
- В) Slider
- Г) ProgressBar

Вопрос 250. Какой из нижеперечисленных компонентов является основным меню?

- А) Button
- Б) MenuBar
- В) Slider
- Г) ProgressBar

Вопрос 251. Какой из нижеперечисленных компонентов является ползунком?

- А) Button
- Б) MenuBar
- В) Slider
- Г) ProgressBar

Вопрос 252. Какой модуль qt отвечает за работу с базами данных?

- А) QSql
- Б) QScript
- В) Qml
- Г) QtNetwork

Вопрос 252. Какой параметр необходимо добавить в файл-проект для включения поддержки баз данных?

- А) QT += network
- Б) QT += sql
- В) QT += widgets
- Г) QT += serialport

Вопрос 254. Какой заголовочный файл необходимо подключить для работы с классами модуля QSql?

- А) #include <QtWidgets>
- Б) #include <QSerialPort>
- В) #include <QtSql>
- Г) #include <QLineEdit>

Вопрос 255. Какой класс не относится к уровню драйверов баз данных?

- A) QSqlDriver
- Б) QSqlDriverCreatorBase
- В) QSqlDriverPlugin
- Г) QSqlDatabase

Вопрос 256. Какой класс не относится к программному уровню баз данных?

- A) QSqlDriver
- Б) QSqlDatabase
- В) QSqlQuery
- Г) QSqlError

Вопрос 257. Какой класс не относится к уровню пользовательского интерфейса баз данных?

- A) QSqlQueryModel
- Б) QSqlDriver
- В) QSqlTableModel
- Г) QSqlRelationalTableModel

Вопрос 258. Какой идентификатор менеджера относится к доступу к базам данных Oracle через Oracle Call Interface?

- A) QOCI
- Б) QODBC
- В) QMYSQL
- Г) QPSQL

Вопрос 259. Какой идентификатор менеджера СУБД относится к доступу к базам данных Oracle через Open Database Connectivity?

- A) QOCI
- Б) QODBC
- В) QMYSQL
- Г) QPSQL

Вопрос 260. Какой идентификатор менеджера СУБД относится к СУБД MySql?

- A) QOCI
- Б) QODBC
- В) QMYSQL
- Г) QPSQL

Вопрос 261. Какой идентификатор менеджера СУБД относится к Sybase Adaptive Server?

- A) QOCI
- Б) QODBC
- В) QMYSQL
- Г) QPSQL

Вопрос 262. Какой идентификатор менеджера СУБД относится к SQLite2?

- A) QSQLITE2
- Б) QSQLITE
- В) QIBASE
- Г) QDB2

Вопрос 263. Какой идентификатор менеджера СУБД относится к SQLite3?

- A) QSQLITE2
- Б) QSQLITE
- В) QIBASE
- Г) QDB2

Вопрос 264. Какой идентификатор менеджера СУБД относится к Borland InterBase?

- A) QSQLITE2
- Б) QSQLITE
- В) QIBASE
- Г) QDB2

Вопрос 265. Какой идентификатор менеджера СУБД относится к DB/2 — платформонезависимая база данных, разработанная IBM?

- A) QSQLITE2
- Б) QSQLITE
- В) QIBASE
- Г) QDB2

Вопрос 266. Какой метод класса QSqlDatabase закрывает подключение к базе данных?

- A) close
- Б) connectOptions
- В) databaseName
- Г) driver

Вопрос 267. Какой метод класса QSqlDatabase возвращает текущие настройки подключения?

- A) close
- Б) connectOptions
- В) databaseName
- Г) driver

Вопрос 268. Какой метод класса QSqlDatabase возвращает имя базы данных?

- A) close
- Б) connectOptions
- В) databaseName
- Г) driver

Вопрос 269. Какой метод класса QSqlDatabase возвращает указатель на объект драйвера?

- A) close
- Б) connectOptions
- В) databaseName
- Г) driver

Вопрос 270. Какой метод класса QSqlDatabase возвращает имя драйвера?

- A) driverName
- Б) exec
- В) hostName
- Г) isOpen

Вопрос 271. Какой метод класса QSqlDatabase позволяет выполнить SQL запрос?

- A) driverName
- Б) exec
- В) hostName
- Г) isOpen

Вопрос 272. Какой метод класса QSqlDatabase возвращает имя сервера базы данных?

- A) driverName
- Б) exec
- В) hostName
- Г) isOpen

Вопрос 273. Какой метод класса QSqlDatabase проверяет, открыто ли подключение к базе данных?

- А) driverName
- Б) exec
- В) hostName
- Г) isOpen

Вопрос 274. Какой метод класса QSqlDatabase открывает подключение к базе данных?

- А) open
- Б) password
- В) port
- Г) setConnectOptions

Вопрос 275. Какой метод класса QSqlDatabase возвращает текущий пароль доступа к базе данных?

- А) open
- Б) password
- В) port
- Г) setConnectOptions

Вопрос 276. Какой метод класса QSqlDatabase возвращает порт базы данных?

- А) open
- Б) password
- В) port
- Г) setConnectOptions

Вопрос 277. Какой метод класса QSqlDatabase устанавливает параметры подключения к базе данных?

- А) open
- Б) password
- В) port
- Г) setConnectOptions

Вопрос 278. Какой метод класса QSqlDatabase устанавливает имя базы данных?

- А) setDatabaseName
- Б) setHostName
- В) setPassword
- Г) setPort

Вопрос 279. Какой метод класса QSqlDatabase устанавливает имя сервера базы данных?

- А) setDatabaseName
- Б) setHostName
- В) setPassword
- Г) setPort

Вопрос 280. Какой метод класса QSqlDatabase устанавливает пароль доступа к базе данных?

- А) setDatabaseName
- Б) setHostName
- В) setPassword
- Г) setPort

Вопрос 281. Какой метод класса QSqlDatabase устанавливает порт для доступа к базе данных?

- А) setDatabaseName
- Б) setHostName
- В) setPassword
- Г) setPort

Вопрос 282. Какой метод класса QSqlDatabase устанавливает имя пользователя?

- А) setUsername
- Б) tables
- В) userName
- Г) transaction

Вопрос 283. Какой метод класса QSqlDatabase возвращает список таблиц в базе данных?

- А) setUsername
- Б) tables
- В) userName
- Г) transaction

Вопрос 284. Какой метод класса QSqlDatabase возвращает имя пользователя?

- А) setUsername
- Б) tables
- В) userName
- Г) transaction

Вопрос 285 Что такое системная служба в операционной системе Windows?

- А) это приложения, автоматически запускаемые системой при запуске Windows и выполняющиеся вне зависимости от статуса пользователя
- Б) это набор драйверов для принтера
- В) это приложения для работы с текстовыми документами
- Г) это приложения для компилирования программ из объектных файлов

Вопрос 286. Что такое «Демоны» в операционных системах семейства Linux?

- А) это вирусы
- Б) это фоновые приложения (аналог служб в ОС Windows), выполняющие специфические задачи без участия пользователя
- В) это специальные драйвера, обеспечивающие работу USB-портов
- Г) это системные утилиты, обеспечивающие безопасность

Вопрос 287. Для чего предназначен класс QTcpSocket?

- А) для работы с аудиоинтерфейсом
- Б) для работы с USB-устройствами
- В) для работы с поточными (stream) сокетами
- Г) для работы с дейтаграммными (datagram) сокетами

Вопрос 288. Для чего предназначен класс QUdpSocket?

- А) для работы с аудиоинтерфейсом
- Б) для работы с USB-устройствами
- В) для работы с поточными (stream) сокетами
- Г) для работы с дейтаграммными (datagram) сокетами

Вопрос 289. Какой метод класса QTcpSocket осуществляет сброс текущего соединения?

- А) abort
- Б) bind
- В) connectToHost
- Г) disconnectFromHost

Вопрос 290. Какой метод класса QTcpSocket осуществляет привязку сокета к адресу и порту?

- А) abort
- Б) bind
- В) connectToHost
- Г) disconnectFromHost

Вопрос 291. Какой метод класса QTcpSocket осуществляет подключение к удаленному серверу?

- А) abort
- Б) bind
- В) connectToHost
- Г) disconnectFromHost

Вопрос 292. Какой метод класса QTcpSocket осуществляет отключение от удаленного сервера?

- А) abort
- Б) bind
- В) connectToHost
- Г) disconnectFromHost

Вопрос 293. Какой метод класса QTcpSocket возвращает информацию об ошибках, при их наличии?

- А) error
- Б) flush
- В) isValid
- Г) localAddress

Вопрос 294. Какой метод класса QTcpSocket осуществляет сброс информации из внутреннего буфера?

- А) error
- Б) flush
- В) isValid
- Г) localAddress

Вопрос 295. Какой метод класса QTcpSocket позволяет проверить, готовы ли сокет для передачи данных?

- А) error
- Б) flush
- В) isValid
- Г) localAddress

Вопрос 296. Какой метод класса QTcpSocket возвращает локальный адрес?

- А) error
- Б) flush
- В) isValid
- Г) localAddress

Вопрос 297. Какой метод класса QTcpSocket возвращает локальный порт?

- A) localPort
- Б) readBufferSize
- В) state
- Г) hostFound

Вопрос 298. Какой метод класса QTcpSocket возвращает размер внутреннего буфера?

- A) localPort
- Б) readBufferSize
- В) state
- Г) hostFound

Вопрос 299. Какой метод класса QTcpSocket возвращает текущее состояние сокета?

- A) localPort
- Б) readBufferSize
- В) state
- Г) hostFound