

Проверочная работа 4 (Итоговая проверочная работа)

1. Дать определение понятиям (Кратко, 1-2 строки на определение):

- 1.1 программирование, язык программирования;
- 1.2 алгоритм, программа, среда разработки (IDE);
- 1.3 операнд, операция, итерация;
- 1.4 объектно-ориентированное программирование, класс и объект, библиотека классов;

2. Записать пропущенное слово.

- 2.1 ***** – этап разработки компьютерной программы, на котором обнаруживают, локализируют и устраняют ошибки.
- 2.2 ***** – представляет набор однотипных данных. Привести пример использования.
- 2.3 ***** – вызов метода из самого метода. Привести пример использования.
- 2.4 ***** – оценка количества базовых действий, необходимых для выполнения алгоритма.
- 2.5 ***** – создание нескольких методов с одинаковым названием, но разными аргументами.
- 2.6 ***** – совокупность объектов заданного типа. Данные типы предоставляют гибкий способ работы с группами объектов и применяются при решении специальных задач.
- 2.7 Оператор «*****» означает экранирование всей строки.
- 2.8 ***** – это блок кода, содержащий ряд инструкций.

3. Какие компоненты включает в себя среда разработки (не меньше 3)?

4. Расставить на свои места этапы преобразования C# кода в машинный код:

Машинный код (1), код на промежуточном языке MSIL (2), JIT компилятор (3), код на языке C# (4), компилятор C# (5).

5. Классификация типов данных по организации в памяти. Привести примеры по каждой группе (не менее 10 в сумме, только названия).

6. Что означает ключевое слово **var**?

7. Назвать типы вычислительных процессов (3 типа).

8. Перечислить операторы **ветвления**. Привести примеры использования.

9. Перечислить **циклы**. Привести примеры использования.

10. Что произойдет, если поместить специальное слово **continue** в тело цикла?

11. Что произойдет, если поместить специальное слово **break** в тело цикла?

12. Найти ошибки в алгоритме и исправить. Для чего нужен (что делает) данный алгоритм?

```
private static double Method(double[] arr, bool reverse)
{
    for (int i = 0; i < length(arr); i+=1)
    {
        for (int j = i; j < length(arr); j--)
        {
            if (arr[i].CompareTo(arr[j]) = (reverse ? -1 : 1))
                arr[i] = arr[j];
                arr[j] = arr[i];
        }
    }
}
```

13. Что будет выведено на консоль?

```
var a = 1;
var b = "2";
var c = 3;
var d = 4;
var person = new Person { Id = 5 };
method(ref a, b, out c, d, person, new Person());
Console.WriteLine($"{a} {b} {c} {d} {person.Id}");
```

```
void method(
    ref int a,
    string b,
    out int c,
    double d,
    Person p1,
    Person p2)
{
    a = 10;
    b = "9";
    c = 8;
    d = 7;
    p1.Id = 6;
    p2 = new Person() { Id = 5 };
}
```

```
class Person
{
    public int Id;

    public Person()
    {
        Id = 1;
        Console.WriteLine(Id);
    }
}
```

14. Какое ключевое слово необходимо указать, если мы хотим, чтобы метод принимал переменное число аргументов?

15. Какого типа данных переменная **Vasya**? Зачем нужен данный тип?

`(string, int) Vasya = ("Вася", 175);`

Дать название словами, писать «(string, int)» не нужно.

16. Что будет, если поместить оператор **return** в метод с возвращаемым типом **void**, какое значение вернет метод?
17. Для чего используется **StringBuilder**, в чем его преимущество перед **string**, в каких ситуациях нужно его использовать?
18. Напишите метод, переводящий все символы строки из нижнего регистра в верхний, а из верхнего в нижний.
- Пример: фразу «Привет Мир!» должно перевести в «пРИВЕТ мИР!»*
- (Подсказка: все нужные методы есть в структуре **char**, т.е. **char.#####**)*

Примечание:

Задание 1 оценивается в 4 балла, задание 2 – в 8 баллов, 13 и 18 оцениваются в 2 балла, остальные в 1 балл. Всего 30 баллов.