**1. Расскажите как генерируется исключение.**

Исключение представляет событие, которое возникает при выполнении программы и нарушает ее нормальный ход. Если исключение не обработано, то программа прекращает свою работу.

Поэтому при возникновении исключений их следует обрабатывать.

**2. Расскажите методику обработки исключений.**

Для обработки исключений применяется конструкция try..catch..finally. В блок try помещаются те действия, которые потенциально могут вызвать исключение. Блок catch перехватывает возникшее исключение и обрабатывает его. Блок finally выполняет некоторые завершающие действия

**3. Какое ключевое слово служит для обозначения блока кода, в котором можно генерировать исключение?**

throw

**4. Какие ключевые слова используются для обработки и генерации исключений?**

Ключевое слово *throw* служит для генерации исключений, *try* - для выделения фрагмента кода, в котором генерируется исключение, а *catch* - для обработки исключения из предыдущего фрагмента *try*. При этом блок *catch* следует сразу после *try*, а самих блоков *catch* может быть сколько угодно.

**Расскажите об механизме обработке исключения?**

Если же в блоке try вдруг возникает исключение, то обычный порядок выполнения останавливается, и среда CLR начинает искать блок catch, который может обработать данное исключение. Если нужный блок catch найден, то он выполняется, и после его завершения выполняется блок finally.

**5. Что будет, если в программе нет предложения catch, способного обработать исключение?**

Если нужный блок catch не найден, то при возникновении исключения программа аварийно завершает свое выполнение.

**6. Что такое фильтры исключения? Приведите пример**

Определённые условия, при которых будет вызвано исключение.

Фильтр исключения позволяет указать дополнительные условия, при которых используется обработчик исключения.

Эти условия принимают форму булева выражения, перед которым

ставится ключевое слово when.

catch (Exception ex) when

(ex.GetType() != typeof(System.FormatException))

{

// Обработка всех ранее не перехваченных исключений,

// кроме того, которое называется FormatException

}

**7. Могут ли исключения быть вложенными?**

try могут быть вложенные

**8. Какой синтаксис нужно использовать в C# для отлова любого возможного**

**исключения?**

finally

catch (Exception ex);

**9. Чем следует руководствоваться при размещении обработчиков исключения?**

в try помещать блоки кода, где возможно появление ошибок

**10. Что будет выведено на консоль в результате выполнения фрагмента листинга?**

static void Main(string[] args)

{

string[] str = new string[5];

try

{

str[4] = "anything";

Console.WriteLine("It's OK");

}

catch (IndexOutOfRangeException e)

{

Console.WriteLine("IndexOutOfRangeException");

}

catch (Exception e)

{

Console.WriteLine("Exception");

}

}

}

**11. Как повторно сгенерировать то же самое исключение в блоке обработчике catch?**

catch (DivideByZeroException)

{

Console.WriteLine("Деление на ноль!");

throw;

}

**12. Какие методы содержаться в классе Exception? Где и как их можно использовать?**

StackTrace, InnerException, Message, HelpLink, HResult Source, TargetSite и Data

при обработке исключений выводить результат их выполнения в консоль