**Тема 1. Основы CLR. 24 часа**

1. Читать Библию С# стр. 21-48
2. CLR via C# стр. 28 – 122 ([https://github.com/micros-uz/csharp/blob/master/Level%201/Books/Рихтер%20Дж.%20-%20%20CLR%20via%20C%23.%20Программирование%20на%20платформе%20Microsoft%20.NET%20Framework%204.5%20на%20языке%20C%23%20(Мастер-класс)%20-%202013.pdf)](https://github.com/micros-uz/csharp/blob/master/Level%201/Books/%D0%A0%D0%B8%D1%85%D1%82%D0%B5%D1%80%20%D0%94%D0%B6.%20-%20%20CLR%20via%20C%23.%20%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%20%D0%BD%D0%B0%20%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%20Microsoft%20.NET%20Framework%204.5%20%D0%BD%D0%B0%20%D1%8F%D0%B7%D1%8B%D0%BA%D0%B5%20C%23%20(%D0%9C%D0%B0%D1%81%D1%82%D0%B5%D1%80-%D0%BA%D0%BB%D0%B0%D1%81%D1%81)%20-%202013.pdf))
3. Ко второму пункту видео <https://www.youtube.com/playlist?list=PLIIXgDT0bKw6i4jQkHVgIrnsP-DcvBieq> (1,2,3 ролики)
4. Представьте себе, что вам предстоит собеседовать соискателя. Составьте для него вопросы по этой теме (письменно).

**Тема 2. Основы С#. 16 часов**

1. Читать Библию С# стр. 48 – 88.
2. Решите задачи:

**Без использования массивов. Все числа – целые. Во всех задачах, если пользователь вводит не число – предлагать ему исправить ввод.**

1. Ввести трехзначное число. Посчитать сумму цифр, вывести на экран.
2. Вводить числа, пока не будет введен ноль. Вывести число с максимальной суммой цифр в нем
3. Ввести n чисел (n задается пользователем). Посчитать сумму трех первых нечетных.
4. Ввести n чисел (n задается пользователем). Посчитать сумму трех последних нечетных.
5. Ввести m и n, где m и n - два целых числа не больше 20. Звездочками на экране нарисовать:

а) прямоугольник вида:

\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*

m и n – строны прямоугльника.

б) треугольник с катетом m вида:

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

в)

г)

д)

е) Вывести из звездочек ромб со стороной n

1. Посчитать сумму ряда

а) 1 + 2 + 3 + .................. + 50

             б)  2+4+6+8+10 + .....  +50

             в)  1+3+5+7+9....

1. Посчитать сумму  6 + 10 + 14 + ................... + 46. Сколько слагаемых?
2. Посчитать сумму  6 + 10 + 14 + ...................   , всего 10 слагаемых.
3. Посчитать сумму 1+2+4+8+16+....., всего 11 слагаемых.
4. Посчитать сумму  6 + 10 + 14 + ..................., Остановиться, когда сумма превысит 100. Сколько слагаемых?
5. Посчитать сумму  6 + 10 + 14 + ..................., последнюю, которая еще не превышает 100. Сколько слагаемых?
6. Посчитать сумму первых n чисел Фиббоначи: 1 + 1 + 2 + 3 + 5 + 8 +  13 + … n задается пользователем

**Массивы**

1. Ввести n целых чисел (n задается пользователем). Какая цифра встречается чаще других? Если таких цифр несколько – вывести ту из них, которая обозначает наибольшее число, а также сколько раз она встретилась.
2. Ввести массив целых чисел. Длина массива задается пользователем. Определить, упорядочен ли он по возрастанию. По убыванию?
3. Ввести массив целых чисел в диапазоне [-100,100]. Длина массива задается пользователем.

а) Найти минимальный элемент

б) Найти максимальный элемент

в) найти минимальное нечетное

г) найти минимальное четное

д) найти минимальный и максимальный элементы и поменять их местами

4. Ввести два упорядоченных массива (контроль за корректностью ввода). Слить их в один упорядоченный массив без использования сортировки.

5. Считать из файла массив целых чисел. Упорядочить по возрастанию. Вывести обратно в файл.

6. Считать из файла массив целых чисел. Упорядочить по убыванию. Вывести обратно в файл.

7. Сдвинуть массив [1..n] циклически влево (вправо) на *m* позиций. «Падающие» элементы должны уходить в хвост (в голову).

**Строки**

1. Даны переменные

hello = "Привет!"

name = "Меня зовут ..."

age = "Мне ... лет"

Оперируя переменными составить полноценное предложение всеми возможными способами (форматирование, интерполяция)

2. Дан массив строк ["apple", "banana", "orange", "kiwi", "mango"]

- Вывести все значения через ", "

- Вывести все значения построчно

3. Сравнить две строки и вывести результат сравнения

- "привет" и "здравствуйте"

- "двацдать" и "двенадцать"

- "синус" и "синусоида"

- "14" и "81"

4. Найти в строке индекс первого вхождения буквы "о"

- Хорошо в лесу...

- Эх, дороги, пыль да туман

- Семнадцать вариантов решения

5. Найти в строке индекс последнего вхождения буквы "у"

- Где такое интересное место?

- У меня дома есть ноутбук.

- Винтажный стул

6. Вставить в строку другую строку

- Какой ... день

- замечательный

7. Заменить в строке слово "магазин" на "парк"

- Привет, я иду в магазин

8. Удалить из строки слово "большого"

- Сегодня в зоопарке я видел большого жирафа

9. Привести предложение "ПрыгаЮщие БуквЫ" к нижнему, а затем к верхнему регистру

10. Разделить строку на элементы массива

- Первый рабочий день прошел на ура

3. Представьте себе, что вам предстоит собеседовать соискателя. Составьте для него вопросы по этой теме (письменно).

**Тема 3. Основы ООП (24 часа)**

1. Читать Библию C# стр. 88 – стр. 141, выполняя примеры из книги. Измените сарай на любую свою фантазию, выполните пример из книги относительно нее. Так же поступите и с другими примерами.
2. Задача.

**Библиотекарь.**

У библиотекаря в фонде хранятся журналы и книги. Написать простое приложение (консольное) для ввода информации о книгах и журналах и удаления записей.

1. Приложение должно по запросу «-help» выводить справочник доступных команд.
2. Приложение заканчивает свою работу при введении команды «-end».
3. Данные о книгах и журналах хранятся только на время выполнения приложения.
4. Для книг известны следующие параметры: код, название, имеющееся количество, автор, жанр, год издания, издательство.
5. Для журналов известны следующие параметры: код, название, периодичность, имеющееся количество, издательство, год, номер.
6. Программа должна обеспечивать

- ввод книги или журнала

- удаление книги или журнала

- изменение параметров уже введенных позиций  
- поиск позиции по названию (если несколько с одинаковым – выводить все)

При написании приложения заложить возможность подключить другие способы хранения информации (файл, база данных), реализовывать эти способы на данном этапе не надо.

1. Представьте себе, что вам предстоит собеседовать соискателя. Составьте для него вопросы по этой теме (письменно).

**Тема 4. ООП – продолжение (40 часов)**

1. Читать Библию C# стр. 212 – 303

2. Задачи

2.1. Выполнить задачу про калькулятор отсюда: https://habr.com/ru/post/143925/  
Лучше всего сначала прочитать, потом закрыть и попробовать сделать.

2.2. Чат-бот

Написать чат-бот, который:

- Умеет здороваться, если ему сказать «Привет», «Здравствуй», «Здравствуйте», «Добрый день», «Добрый вечер», «Доброе утро» или «Доброй ночи».

- На вопрос «Как тебя зовут?» отвечает «Шарпик».

- на строку, содержащую слово «Анекдот» - отвечает анекдотом.

- На вопрос «Сколько времени» или «Который час» - сообщает сколько сейчас времени.

- На строку, которую не может опознать отвечает афоризмом.

- На слова «Пока» или «До свидания» - прощается и прекращает работу.  
  
Варианты ответов содержатся в наборах «Приветствия», «Анекдоты», «Прощания», «Афоризмы». В этом варианте ответы хранятся в фейковых репозиториях (хард-код). Предусмотреть возможность хранения в файлах или БД, реализовывать на данном этапе не надо.

Предусмотреть возможность вводить вопросы не из консоли, а через веб, но на данном этапе реализовывать не надо.

3. Представьте себе, что вам предстоит собеседовать соискателя. Составьте для него вопросы по этой теме (письменно).

**Тема 5. Хранение данных (16 часов)**

1. Читать Библию C# глава 14

2. Задачи:

2.1 В задаче «Библиотекарь» реализовать хранение в бинарных файлах  
2.2 В задаче «Чат-бот» реализовать хранение в xml-файлах (или одном, принять решение самостоятельно)

3. Представьте себе, что вам предстоит собеседовать соискателя. Составьте для него вопросы по этой теме (письменно).

**Тема 6. Многопоточность (32 часа)**

* + - 1. Библия С# глава 15
      2. Задачи:

2.1 Разобрать, выполнить задачу https://metanit.com/sharp/tutorial/11.4.php

2.2 Реализовать заполнение массива на 1 миллион элементов рандомными целыми в одном потоке и в нескольких, сравнить время выполнения.

2.3 Упорядочить массив, заполненный на предыдущем шаге, по возрастанию в нескольких потоках а) пузырком, б) быстрой сортировкой.

2.4 В задаче «Чат-бот» внести изменения таким образом, чтобы консоль была свободна после ввода команды, а результат выводился по готовности.

**TODO**

**Тема 7. Основы БД**

1. Реляционные БД – идея.

2. Простые выборки, условия, join, объединение таблиц, индексы.

3. Просто прочитать про ADO по Библиии, не выполнять упражнений (на случай столкновения с ним в старых проектах)

**Тема 8. EF**

Теория

Задача о библиотекаре с хранением в БД

**Тема 9. Основы web**

http, виды запросов, структура, Request, Response, задача о чат-боте через веб.

**Тема 10. MVC Core**

наше задание о библиотеке.