

ДЕНЬ 2. КОНСОЛЬНЫЕ ПРИЛОЖЕНИЯ 2

2.1. Калькулятор перевода единиц измерений

Пользователь вводит значение с указанием единицы измерения (например, 1 m3), необходимо пересчитать значение в другую единицу измерения согласно варианту. Программа должна поддерживать оба направления пересчета. Программа должна поддерживать ввод в качестве разделителя дробной части как точку, так и запятую. После каждого перевода единиц измерения программа должна завершать свою работу

Примеры

Перевод куб. метров в литры (1 m3 = 1000 l) и наоборот

Введите значение:
> 1 m3

Результат: 1000 l

Введите значение:
> 100 l

Результат: 0.1 m3

Введите значение:
> 0,02 m3

Результат: 20 l

Введите значение:
> 0.003 m3

Результат: 3 l

Варианты

№	Перевод	№	Перевод	№	Перевод	№	Перевод
1	1 hf = 735.5 W	6	1 h = 60 min	11	1 d = 2.54 cm	16	1 km = 1000 m
2	1 p = 0.4536 kg	7	1 min = 60 s	12	1 ft = 0.3048 m	17	1 c = 100 kg
3	1 to = 31.101 g	8	1 d = 24 h	13	1 m = 100 cm	18	1 gal = 3.7854 L
4	1 y = 12 month	9	1 y = 365 d	14	1 kg = 1000 g	19	1 \$ = 67.2 rub
5	1 ha = 0.01 km2	10	1 s = 1000 ms	15	1 KB = 1024 B	20	1 eur = 77.1 rub

2.2. Прочсть данные из консоли и отсортировать по заданному полю

Пользователь вводит записи, состоящие из нескольких полей. Как только он вводит пустую строку в первое из полей, ввод завершается. Необходимо отсортировать записи по заданному полю и вывести их в виде таблицы

Пример

Запись: фамилия (строка), возраст (целое число), результат (вещественное число).
Сортировка: результат (по убыванию)

Введите данные:

Фамилия: Петров
Возраст: 20
Результат: 5.3

Фамилия: Иванов
Возраст: 22
Результат: 7.6

Фамилия: Сидоров
Возраст: 20
Результат: 5

Фамилия:

Фамилия	Возраст	Результат
Иванов	22	7.6
Петров	20	5.3
Сидоров	20	5

Варианты

№	Запись	Сортировка
1	Фамилия (строка), рост (вещественное число), вес (вещественное число)	Рост (по убыванию)
2	Фамилия (строка), должность (строка), оклад в руб (целое число)	Оклад в руб. (по убыванию)
3	Фамилия (строка), число ролей (целое число), гонорар в млн. руб. (вещественное число)	Гонорар в млн. руб. (по убыванию)
4	Адрес отправления (строка), адрес доставки (строка), вес (вещественное число)	Вес (по убыванию)
5	Город (строка), улица (строка), номер дома (число)	Город (по возрастанию)
6	Адрес сайта (строка), число посетителей (целое число), число уникальных посетителей (целое число)	Число уникальных посетителей (по возрастанию)
7	Дисциплина (строка), номер курса (целое число), количество часов (целое число)	Количество часов (по возрастанию)
8	Фамилия (строка), номер группы (строка), номер в группе (целое число)	Номер в группе (по возрастанию)
9	Название цеха (строка), план выпуска деталей (целое число), фактический выпуск деталей (целое число)	Фактический выпуск деталей (по убыванию)
10	Название товара (строка), количество на складе (целое число), количество зарезервированных (целое число)	Количество на складе (по возрастанию)
11	Название материала (строка), объем (вещественное число), вес (вещественное число)	Объем (по возрастанию)

12	Компания (строка), сумма поступлений в млн. руб. (вещественное число), сумма списаний в млн. руб. (вещественное число)	Компания (по возрастанию)
13	Номер телефона (строка), имя оператора (строка), баланс в копейках (целое число)	Баланс в копейках (по убыванию)
14	Автомобильный номер (строка), год выпуска (целое число), пробег в км (целое число)	Пробег в км (по убыванию)
15	Производитель (строка), объем выпуска (вещественное число), средняя цена (вещественное число)	Средняя цена (по убыванию)
16	Фамилия (строка), оценка за теорию (целое число), оценка за практику (целое число)	Фамилия (по возрастанию)
17	Номер заказа (строка), описание (строка), выполнен или нет (логический тип)	Номер заказа (по возрастанию)
18	Тема письма (строка), адресат (строка), есть ли вложения (логический тип)	Адресат (по возрастанию)
19	Название (строка), число сезонов (целое число), год выпуска первого сезона (целое число)	Число сезонов (по возрастанию)
20	Фамилия (строка), год поступления (целое число), средний балл (вещественное число)	Средний балл (по убыванию)

2.3. Выполнить указанную операцию над набором чисел, записанных в файле

Программа запрашивает путь к файлу, в котором записаны числа, каждый на новой строке. Необходимо прочитать числа из файла и выполнить заданную операцию

Пример

Просуммировать числа из файла

Входной файл:

```
10
55.4
12.1
40
```

Программа:

```
Введите путь к файлу:
> digits.txt
```

Результат: 117.5

Варианты

№	Операция	№	Операция
1	Вычислить среднее значение	11	Разделить все числа на первое из чисел и вычислить сумму результатов
2	Просуммировать все числа, которые меньше, чем предыдущее число	12	Просуммировать все четные числа

3	Просуммировать дробные части всех чисел	13	Умножить числа на их порядковые номера и вычислить среднее
4	Перемножить все отрицательные числа	14	Вычислить среднее из квадратов чисел
5	Просуммировать квадраты всех чисел	15	У каждого второго числа поменять знак и просуммировать все числа
6	Умножить числа на их порядковые номера и сложить результаты	16	У каждого второго числа поменять знак и посчитать среднее
7	Просуммировать все нечетные числа	17	Перемножить все положительные числа
8	Перемножить дробные части всех чисел	18	Просуммировать все числа, которые больше первого из чисел
9	Просуммировать модули всех чисел	19	Извлечь квадратный корень из суммы квадратов чисел
10	Просуммировать все числа, которые больше, чем предыдущее число	20	Округлить числа и сложить результаты

2.4. Разработка теста с вводом нескольких вариантов ответов и сохранением результатов в файл

Вам необходимо разработать программу для тестирования знаний на заданную тему, посвященную прикладному программированию и языку C#. Тест должен состоять из 3-ех вопросов, формулировку которых вам нужно придумать самостоятельно.

Перед началом теста программа запрашивает имя пользователя.

Далее программа должна выводить вопросы **в случайном порядке** (для этого используйте класс *Random*). После вывода вопроса программа предлагает на выбор 4 варианта ответа. При этом **правильными являются несколько из них** (от двух до трех).

Для того, чтобы ответить, пользователь вводит номера вариантов ответа через пробел. Пользователь может вводить числа в любом порядке. Если пользователь ввел одно и то же число несколько раз, то оно должно учитываться только однажды. Числа вне диапазона 1-4 должны игнорироваться.

После ввода вариантов пользователь нажимает клавишу *Enter*, и программа должна вывести либо следующий вопрос (если они не кончились), либо результат тестирования. Результат тестирования считается по следующей схеме:

Каждый ответ на вопрос оценивается от 0 до 1 балла:

- если пользователь ничего не вписал или выбрал только неверные ответы, он зарабатывает 0 баллов
- если пользователь выбрал хотя бы один верный ответ, то:
 - за каждый верный ответ он получает $1/R$ баллов, где R – количество верных ответов на этот вопрос
 - за каждый неверно вписанный ответ он получает -0.5 баллов штрафа, при этом оценка за вопрос не может стать меньше 0

После ответа на все вопросы число заработанных баллов делится на количество вопросов и выводится на экран в виде процентов.

Кроме вывода на экран, результат должен записывать в файл *“results.txt”* в формате: <Имя>: <Процент>. Каждая попытка прохождения теста записывается на новой строке.

Пример

Тема: переменные

Введите ваше имя:
> Misha

Вопрос 1. Какими из способов можно задать переменную в C#?

- 1) `int x;`
 - 2) `var y;`
 - 3) `var z = 5;`
 - 4) `x: int;`
- > 3 1

Вопрос 2. Что можно указать переменной ее при объявлении?

- 1) тип
 - 2) начальное значение
 - 3) имя
 - 4) дорогу
- > 1 2 4

Вопрос 3. Какие символы допустимы в названии переменной?

- 1) Тире
 - 2) Латинские буквы
 - 3) Цифры, если они стоят не в начале
 - 4) Знак доллара
- > 2

Ваш результат: 55.6%

Результат был рассчитан следующим образом:

- 1) полностью верно = 1 балл
- 2) 1 и 2 – верно, 3 - пропущено, 4 – нет = $1/3 + 1/3 - 0.5 \sim 0.167$ баллов
- 3) 2 – верно, 3 – пропущено = $1/2 = 0.5$ баллов

Итого: $(1 + 0.167 + 0.5) / 3 \sim 0.556$ или 55.6%

В файле *results.txt* добавится строка:

Misha: 55.6%

Варианты

№	Преобразование	№	Преобразование
1	Функции и методы в C#	11	Тестирование
2	Типы данных C#	12	Ключевые слова C#
3	Механизм событий	13	Объектно-ориентированное программирование
4	Обобщения (generics) в C#	14	Обработка файлов
5	Этапы разработки ПО	15	Операторы языка C#
6	Стандартная библиотека C#	16	Структуры в C#
7	UML-диаграммы	17	Отличия C# от C

8	Массивы в C#	18	Преобразование типов в C#
9	Классы в C#	19	Механизм исключений
10	Коллекции в C#	20	Форматирование вывода в C#

2.5. Обработка файла с учетом аргументов командной строки

Вам необходимо реализовать программу, принимающую входную информацию через аргументы командной строки.

Пример запуска программы:

```
program.exe input.txt output.txt -from 5 -to 8
```

Аргументы:

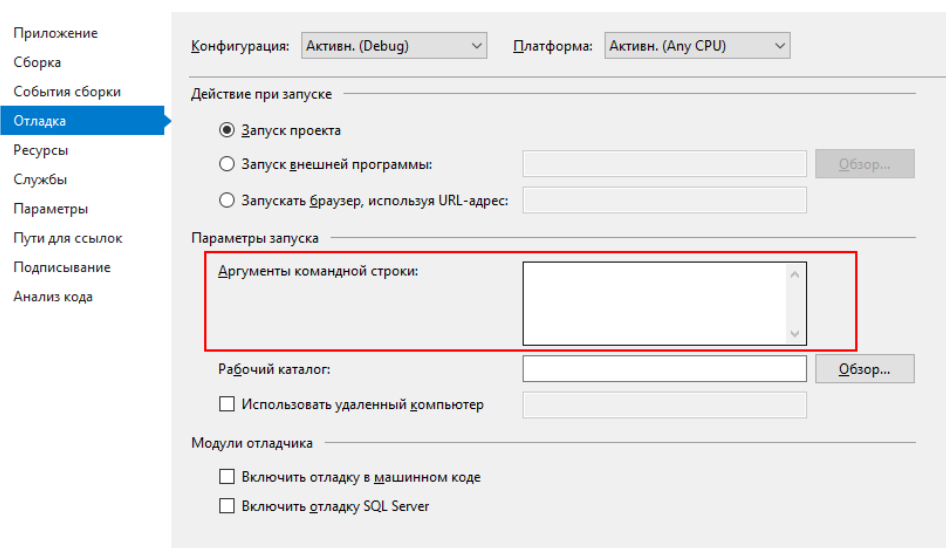
- *input.txt* – входной файл, который нужно обработать. Обязательный аргумент. Если не задан, то выдать сообщение «Укажите входной файл» и завершить работу
- *output.txt* – выходной файл, куда нужно сохранить результат. Если аргумент не передан, то результат нужно вывести в консоль. Этот аргумент всегда идет после аргумента с входным файлом
- *-from 5* – с какой строки файла начинать обработку. Если не задан, то обработка начинается с начала файла
- *-to 8* – до какой строки файла, включительно, проводить обработку. Если не задан, то обработка производится до конца файла

Порядок аргументов может быть произвольным

Получить значение аргументов можно в главной функции программы:

```
static void Main(string[] args) // args - аргументы
```

Для тестирования указать аргументы командной строки можно в настройках проекта:



Пример

Обработка: удаление каждого второго слова

Файл *poem.txt*

В томленьях грусти безнадежной
В тревогах шумной суеты,
Звучал мне долго голос нежный
И снились милые черты.

Шли годы. Бурь порыв мятежный
Рассеял прежние мечты,
И я забыл твой голос нежный,
Твои небесные черты.

Запуск программы:

```
program.exe -from 2 poem.txt res.txt -to 6
```

После запуска программы должен создаться *res.txt*:

В шумно,
Звучал долго нежный
И милые

Шли Бурь мятежный

Варианты

№	Преобразование	№	Преобразование
1	Удалить все слова без гласных букв	11	Удалить все гласные
2	Удалить все слова размером меньше, чем из 5 букв	12	Удалить знаки препинания
3	Удалить все слова без согласных букв	13	Перевернуть каждое слово в строке (абг деж -> гба жед)
4	Перевести в верхний регистр все гласные	14	Перевернуть строки (абг деж -> жед гба)
5	Перевести в верхний регистр все согласные	15	Перевести все буквы первого слова каждой строке в верхний регистр
6	Перемешать в случайном порядке все слова в строке	16	Удалить все слова размером больше, чем из 5 букв
7	Удалить первое и последнее слова в строках	17	Поменять местами слова в каждой паре слов: аб вг де жз -> вг аб жз де
8	Перевести первую букву каждого слова в верхний регистр	18	Удалить все согласные
9	Удалить все слова, в которых больше 5 согласных	19	Удалить все слова, в которых меньше 5 согласных
10	Удаление каждого второго слова	20	Перевести в верхний регистр каждую вторую букву слов: абв гдеж -> аБв гДеЖ