

Практическая работа №1

Парадокс Монти Холла

Парадокс Монти Холла — одна из известных задач теории вероятностей, решение которой, на первый взгляд, противоречит здравому смыслу.

Парадокс основан на американском телешоу «Let's Make a Deal» и назван в честь ведущего этой передачи.

Правила игры

Участнику игры нужно выбрать одну из трёх дверей. За одной из дверей находится приз, за двумя другими дверями ничего нет. Участнику выбирает одну из дверей, например, номер 1, после этого ведущий, который знает, где находится приз, открывает одну из оставшихся дверей, например, номер 3, за которой ничего нет. После этого он спрашивает участника: не желаете ли вы изменить свой выбор и выбрать дверь номер 2? Участник может согласиться либо остаться при своём выборе.

Вопрос

Увеличиваются ли шансы участника выиграть приз, если он примет предложение ведущего и изменит свой выбор?

Предположения

- Математики утверждают, что увеличатся на 60%.
- Разум подсказывает, что нет.

Задание

Напишите программу, которая подтвердит или опровергнет доводы математиков.

Помощь

Для решения задачи вам потребуется импортировать модуль random:

```
import random
```

и использовать его метод randrange. Метод randrange принимает в качестве аргумента два числа n1 и n2 и генерирует случайное число в диапазоне от n1 до n2-1:

```
random.randrange(2, 5)
```

вернёт либо 2, либо 3, либо 4

Практическая работа №2

Парадокс дней рождения

В группе, состоящей из 23 или более человек, вероятность совпадения дней рождения (число и месяц) хотя бы у двух людей превышает 50%. Например, если в классе 23 ученика или более, то более вероятно то, что у кого-то из одноклассников дни рождения придется на один день, чем то, что у каждого будет свой неповторимый день рождения.

Для 60 и более человек вероятность такого совпадения превышает 99%.

Утверждение не является парадоксом в строгом научном смысле: логического противоречия в нём нет, а парадокс заключается лишь в различиях между интуитивным восприятием ситуации человеком и результатами математического расчёта.

Интуитивное восприятие

В группе из 23 человек вероятность совпадения дней рождения у двух человек столь высока, потому что рассматривается вероятность совпадения дней рождения у любых двух человек в группе. Эта вероятность определяется количеством пар людей, которые можно составить из 23 человек. Так как порядок людей в парах не имеет значения, общее число таких пар равно числу сочетаний из 23 по 2, то есть $(23 \times 22) / 2 = 253$ пары.

В формулировке парадокса речь идёт именно о совпадении дней рождения у каких-либо двух членов группы. Одно из распространённых заблуждений состоит в том, что этот случай путают с другим случаем, на первый взгляд похожим, когда из группы выбирается один человек, и оценивается вероятность того, что день рождения каких-либо других членов группы совпадёт с днём рождения выбранного человека. В последнем случае вероятность совпадения значительно ниже.

Задание 1

Утверждение кажется неочевидным, поэтому напишите программу, которая его подтвердит или опровергнет.

Для удобства можно считать, что в каждом месяце 28 дней.

Задание 2

Оформите код решения парадокса в виде функции `birthday()`, которая принимает в качестве параметра количество итераций и возвращает процент в виде числового значения.

Задание 3

Оформите код решения предыдущего парадокса (Практикум-1) в виде функции `monty_hall()`, которая принимает в качестве параметра количество итераций и возвращает процент в виде числового значения.

Объедините обе функции в один модуль под именем `paradox` и используйте как:

```
import paradox

print( paradox.monty_hall(10000) )

print( paradox.birthday(1000) )
```

Практическая работа №3

Игра “Поле чудес”

«Поле чудес» - советская и российская телеигра, российский аналог американской телевизионной программы «Колесо Фортуны».

Задача

Написать упрощённую версию игры, которая запускается в консоли для одного игрока.

Задание 1

- У вас есть список слов: “книга”, “месяц”, “ручка”, “шарик”, “олень”, “носок”.
Обратите внимание на то, что длина всех слов одинакова.
- Необходимо выбрать случайным образом слово и отрисовать его, используя вместо букв какие-либо символы, например “\u25A0”.
- Необходимо установить счётчик “жизни” в какое-либо значение, например 5
- Предложить игроку ввести букву или всё слово целиком.
 - Если буква правильная, то слово перерисовывается с видимой буквой.
 - Если буква неправильная, то у игрока отнимается одна “жизнь”.
 - Если игрок ввёл слово и это слово правильно, либо это последняя правильная буква, либо у игрока закончились “жизни”, то игра заканчивается.

```

>>> %Run wheel-of-fortune.py
■ ■ ■ ■ ■ | ♥x3
Назовите букву или слово целиком: е
■ ■ Е ■ ■ | ♥x3
Назовите букву или слово целиком: р
Неправильно. Вы теряете жизнь
■ ■ Е ■ ■ | ♥x2
Назовите букву или слово целиком: о
О ■ Е ■ ■ | ♥x2
Назовите букву или слово целиком: в
Неправильно. Вы теряете жизнь
О ■ Е ■ ■ | ♥x1
Назовите букву или слово целиком: ь
О ■ Е ■ Ъ | ♥x1
Назовите букву или слово целиком: олень
О Л Е Н Ъ
Вы выиграли! Приз в студию!

```

Примерный вид программы.

Задание 2

- Предложите игроку после каждого тура сыграть ещё или отказаться.
- При согласии количество жизней восстанавливается
- Игра ведётся до тех пор, пока не закончатся слова в списке.

Задание 3

Предложите игроку выбрать уровень сложности текущей игры. Например, при лёгком уровне игроку будет выдаваться 7 “жизней”, при среднем - 5, а при сложном - 3 “жизни”.

Задание 4

Добавьте в список слова разной длины.

Дополнение к практической работе №3

Выполнить практическую работу №3 из сборника со следующими дополнительными условиями:

1. "Игра" должна состоять минимум из двух модулей, оба этих модуля должны быть упакованы в пакет
2. Список слов для игры формируется из текстового файла, с расширением .txt. По умолчанию файл со списком слов лежит в пакете с основными модулями для игры.
3. Создать переменную, отвечающую за количество правильно угаданных слов (record) и после каждого угаданного слова увеличивать ее значение на единицу. В конце игры выводить сообщение с количеством угаданных слов, после чего рекорд должен быть записан в текстовый файл. При дальнейших играх рекорды должны сравниваться и если количество очков после очередной итог больше, чем количество очков, записанное в файле, то файл нужно обновить, записав в него новый рекорд.
4. Пункты 2 и 3 реализовать в отдельном модуле в виде пары функций (можно и больше)

По итогу должен получиться один пакет, состоящий минимум из 2-х модулей и минимум 1-го текстового файла, до первого запуска, со списком слов.

Практическая работа №4

Работа с файлами и текстом

Задача

Задача состоит в том, чтобы подсчитать количество уникальных слов в текстовом файле и записать их в другой файл в алфавитном порядке.

Задание 1

Создайте в текстовом редакторе текстовый файл по имени “data.txt” и запишите в него любой произвольный текст. Текст можно скопировать из любого источника, например из веб-страницы.

Задание 2

Создайте функцию `read_file`.

Функция должна принимать имя файла, прочитать его и вернуть **список** уникальных слов из этого файла в нижнем регистре. Например:

```
words = read_file('data.txt')
# ['артемий', 'лиге', 'соперники', 'буллитов', 'серии', 'гагарина',
  'реализовал', 'форвард', 'время', 'кубок', 'выступает', ...]
```

Задание 3

Создайте функцию `save_file`.

Функция должна принимать имя файла для записи и список уникальных слов. В функции подсчитайте количество уникальных слов и запишите его в файл вместе со всеми словами отсортированными в алфавитном порядке. Например:

```
save_file('count.txt', words)
```

```
# Содержимое файла count.txt
```

```
Всего уникальных слов: 67
```

```
=====
```

```
артемий
```

```
буллит
```

```
буллитов
```

```
в
```

```
видео
```

```
витязь
```

```
вместе
```

```
время
```

```
встречи
```

```
выиграл
```

```
выступает
```

```
...
```

Практическая работа №5

Обработка ошибок при работе с файлом

Дано

Файл с содержимым в формате на каждой строке число, например:

3
12
23
42

Число в первой строке указывает на общее количество чисел в файле.

Задание

Написать программу, которая

- запрашивает у пользователя имя файла,
- зачитывает этот файл
- возвращает список чисел из файла для последующей обработки, например [12, 23, 42]

Задача

Отследите все возможные ошибки, которые могут произойти в коде.

Попробуйте поработать с файлом без оператора with

Практическая работа №6

Обработка файлов регулярными выражениями

Задача

- На железнодорожном вокзале ведётся журналирование рейсов
- Каждый рейс в файле журнала занимает одну строку:
 - Рейс номер-поезда прибыл/отправился из/в город в время
- В файле журнала может присутствовать и другая информация
- Текущий журнал выглядит как:
 - Рейс 365 прибыл из Сасово в 12:56:30
сообщение получено в 12:57:20
Сохранено в базу данных
Рейс 452 отправился в Сочи в 13:04:22
сообщение получено в 13:11:32
Ошибка записи в базу данных

Задание

- Необходимо зачитать файл журнала
- Выбрать необходимую информацию
- Представить информацию в виде:
- [время]: Поезд № номер-поезда из/в город
Например:

[12:56:30] - Поезд № 365 из Сасово

[13:04:22] - Поезд № 452 в Сочи

- Записать преобразованную информацию в новый файл

Практическая работа №7

В данной работе попробуйте использовать все знания, навыки и умения полученные на курсе.

Дано

В файле books.csv находится каталог товаров в формате CSV:

isbn|title|author|quantity|price

978-1-43545-500-9|Python Programming for the Absolute Beginner|Michael Dawson|10|18.90

978-0-59600-372-2|XSLT Cookbook|Sal Mangano|15|34.60

978-0-32168-056-3|Programming in Python 3|Mark Summerfield|8|27.28

978-1-44935-573-9|Learning Python|Mark Lutz|21|34.20

978-0-47178-597-2|Ajax For Dummies|Steve Holzner|32|11.80

978-1-78439-700-5|Mastering Python Networking|Eric Chou|23|31.49

978-8-59037-986-7|Programming in Lua|Roberto Ierusalimschy|12|37.10

978-1-78439-658-9|Machine Learning in Java|Bostjan Kaluza|45|34.99

Задание 1

- Необходимо реализовать функцию, которая выбирает данные для книг с указанным параметром и возвращает список списков вида:

```
[ ['isbn', 'title', 'author', quantity, price], [...], ... ]
```

-
- Например, вызов функции как `get_books('python')` должен вернуть следующий список списков (выбираются книги, где в названии есть подстрока `python`):

```
[['978-1-43545-500-9', 'Python Programming for the Absolute  
Beginner, Michael Dawson', 10, 18.9], [...], ...]
```

Задание 2

- Напишите функцию `get_totals(list)`, которая должна принять в качестве параметра список (результат функции из Задания 1) и вернуть список кортежей вида:

```
['isbn', quantity*price]
```

- Например:

```
[('978-1-43545-500-9', 189.0), ('978-0-59600-372-2', 519.0),  
(...), ...]
```

Задание 3

Измените функцию `get_totals()` так, чтобы она добавляла к сумме какое-то значение, если сумма ниже, чем определённое число.

Например, мы имеем список вида:

```
[('978-1-43545-500-9', 189.0), ('978-0-59600-372-2', 519.0), (...),  
...]
```

Тогда, если сумма меньше, чем 500 и мы хотим увеличить сумму на 100, список должен выглядеть так:

```
[('978-1-43545-500-9', 289.0), ('978-0-59600-372-2', 519.0), (...),  
...]
```

Практическая работа №8. Установка и использование сторонних фреймворков

1. Создайте новый проект в PyCharm и назовите его «English Learn»
2. Скопируйте один из своих диалогов в социальной сети и сохраните его в текстовом файле
3. Перейдите по ссылке: <https://pypi.org/project/pymorphy2/>
4. Ознакомьтесь с технической документацией по фреймворку pymorphy2
5. Установите его в свою среду разработки, используя модуль pip и команду install в терминале
6. Приведите все слова в вашем текстовом файле в нормальную форму
7. Затем из списка слов в текстовом файле создайте словарь: ключ слово, а значение количество упоминания этого слова в тексте (Модуль 7. Задание 7 из задачника). Наиболее часто упоминаемые слова должны идти в начале словаря, наименее упоминаемые в конце
8. Перейдите по ссылке: <https://pypi.org/project/translate/>
9. Ознакомьтесь с технической документацией по фреймворку translate
10. Установите его в свою среду разработки, используя модуль pip и команду install в терминале
11. Переведите все слова в словаре на английский язык, сохраняя порядок следования (от наиболее упоминаемых к наименее упоминаемым)
12. Сохраните полученные результаты в текстовый документ в следующей структуре:

Исходное слово | Перевод | Количество упоминаний

Практическая работа №9. Модуль OS. Взаимодействие с компонентами операционной системы

1. Создайте новый проект в PyCharm и назовите его «Office_Tweaks»
2. В данном проекте Вам необходимо будет реализовать следующий набор функций (см. изображение ниже)

```
Текущий каталог: C:\Users\sa32\PycharmProjects\pythonProject6\Новая папка (2)

Выберите действие:

0. Сменить рабочий каталог
1. Преобразовать PDF в Досх
2. Преобразовать Досх в PDF
3. Произвести сжатие изображений
4. Удалить группу файлов
5. Выход
```

Для реализации данного функционала необходимо использовать методы из модуля OS, а также методы из модулей, установленных со сторонних репозиторийев.

При запуске программа отображает текущий рабочий каталог - папку в которой находится исполняемый файл (модуль с расширением .py).

Все действия, перечисленные под пунктами 1-4, будут выполняться над файлами, которые находятся в рабочем каталоге. В любой момент его можно изменить, выбрав соответствующий пункт в меню и вставив в консоль ссылку на другой каталог.

Пример:

```
Текущий каталог: C:\Users\sa32\PycharmProjects\pythonProject6\Новая папка (2)

Выберите действие:

0. Сменить рабочий каталог
1. Преобразовать PDF в Досх
2. Преобразовать Досх в PDF
3. Произвести сжатие изображений
4. Удалить группу файлов
5. Выход

Ваш выбор: 0
Укажите корректный путь к рабочему каталогу: C:\Users\sa32\Pictures\Разное

Текущий каталог: C:\Users\sa32\Pictures\Разное

Выберите действие:

0. Сменить рабочий каталог
1. Преобразовать PDF в Досх
2. Преобразовать Досх в PDF
3. Произвести сжатие изображений
4. Удалить группу файлов
5. Выход

Ваш выбор:
```

3. Модуль для конвертации документов из формата .pdf в формат .docx (действие 1), а также техническая документация по его работе: <https://pypi.org/project/pdf2docx/>
4. Модуль для конвертации документов из формата .docx в .pdf (действие 2), а также техническая документация по его работе: <https://pypi.org/project/docx2pdf/>
5. Для работы с изображениями рекомендуется использовать библиотеку Pillow (ссылка: <https://pypi.org/project/Pillow/>). В отличие от предыдущих модулей, функционал этой библиотеки довольно обширный, поэтому всю техническую документацию от создателей для такой простой задачи изучать избыточно. Ссылка на описание базовых функции для решения поставленной задачи (действие 3): <https://www.geeksforgeeks.org/change-image-resolution-using-pillow-in-python/>
6. После выбора функций 1-3, в консоли должны отображаться пронумерованные для выбора имена файлов, соответствующие исходному формату в выбранном действии.

Примеры:

```
Текущий каталог: C:\Users\sa32\PycharmProjects\pythonProject6\Новая папка (2)

Выберите действие:

0. Сменить рабочий каталог
1. Преобразовать PDF в Docx
2. Преобразовать Docx в PDF
3. Произвести сжатие изображений
4. Удалить группу файлов
5. Выход

Ваш выбор: 1
Список файлов с расширением .pdf в данном каталоге:

1. Практическое задание 1. Стрельцов А.С..pdf
2. Практическое задание 2. Стрельцов А.С..pdf
3. Практическое задание 2. Стрельцов А.С..pdf.docx.pdf
4. Практическое задание 3. Стрельцов А.С..pdf
5. Практическое задание 3. Стрельцов А.С..pdf.docx.pdf
6. Практическое задание 5. Стрельцов А.С Антиплагиат..pdf
Введите номер файла для преобразования (чтобы преобразовать все файлы из данного каталога введите 0):
```

```
Текущий каталог: C:\Users\sa32\PycharmProjects\pythonProject6\Новая папка (2)

Выберите действие:

0. Сменить рабочий каталог
1. Преобразовать PDF в Docx
2. Преобразовать Docx в PDF
3. Произвести сжатие изображений
4. Удалить группу файлов
5. Выход

Ваш выбор: 2

Список файлов с расширением .docx в данном каталоге:

1. Практическое задание 1. Стрельцов А.С..pdf.docx

Введите номер файла для преобразования (чтобы преобразовать все файлы из данного каталога введите 0):
```

```
Текущий каталог: C:\Users\sa32\Pictures\Разное

Выберите действие:

0. Сменить рабочий каталог
1. Преобразовать PDF в Досх
2. Преобразовать Досх в PDF
3. Произвести сжатие изображений
4. Удалить группу файлов
5. Выход

Ваш выбор: 3

Список файлов с расширением ('.jpeg', '.gif', '.png', '.jpg') в данном каталоге:

1. GKAPPv0QgLI.jpg
2. GSbFWHgK0Lo.jpg
3. H0mJUn6kxds.jpg
4. IaQjRYIYSWQ.jpg
5. ipC4L0-j9TY.jpg
6. Ue7_x6P8PU8.jpg

Введите номер файла для преобразования (чтобы преобразовать все файлы из данного каталога введите 0): 0
Введите параметры сжатия (от 0 до 100%): 50
```

7. В пункте 4 должен быть реализован следующий функционал:

```
Текущий каталог: C:\Users\sa32\PycharmProjects\pythonProject6\Новая папка (2)

Выберите действие:

0. Сменить рабочий каталог
1. Преобразовать PDF в Досх
2. Преобразовать Досх в PDF
3. Произвести сжатие изображений
4. Удалить группу файлов
5. Выход

Ваш выбор: 4
Выберите действие:

1. Удалить все файлы начинающиеся на определенную подстроку
2. Удалить все файлы заканчивающиеся на определенную подстроку
3. Удалить все файлы содержащие определенную подстроку
4. Удалить все файлы по расширению
Введите номер действия: 4
Введите подстроку: docx

Файл: "Практическое задание 1. Стрельцов А.С..pdf.docx" успешно удалён!
```

Для реализации данных действий удобно использовать строковые методы: `startswith()`, `endswith()`.