# Вариант 4 конкурсных заданий *практического* этапа Конкурса

В современном научном сообществе возник настоящий переизбыток информации о различных препаратах и их свойствах. Учёные из разных стран мира публикуют тысячи новых рецептов и формул, которые необходимо систематизировать и упорядочить. Это затрудняет процесс поиска необходимых данных и замедляет научный прогресс. Для решения этой проблемы, вам необходимо создать специальную программу, которая позволит быстро и эффективно находить нужные рецепты препаратов и значительно сокращать время, затрачиваемое на поиски. Эта программа станет незаменимым помощником для учёных, позволяя им сосредоточиться на более важных задачах и исследованиях.

Думаю, что Вы готовы приступать, но сначала необходимо **создать репозиторий** для проекта и задач, которые Вы будете выполнять. Обязательно сделайте его **PUBLIC**, а то мы не сможем проверить решение вашей последней задачи. Каждую задачу вам необходимо будет правильно оформить и залить в репозиторий. *Код, который Вы напишете необходимо задокументировать, чтобы другие программисты могли понять, что делает код и за что отвечает. Примеры документирования приведены ниже.*

**Пример документирования кода на языке С++**

/\*\*

\* Это описание функции foo

\*

\* @param str это описание аргумента str

\* @param pattern это описание аргумента pattern

\* @return это описание того, что вернет функция

\*/

int foo(std::string str, std::string& pattern)

{

...

};

**Пример документирования на языке Python**

def complex(real=0.0, imag=0.0):

"""Описание функции complex.

Описание аргументов:

real – описание аргумента

imag – описание аргумента

"""

if imag == 0.0 and real == 0.0: return complex\_zero

...

Теперь можете приступать к решению задач!

К задачам прикреплен файл *scientist.txt,* который хранит в себе информацию о ФИО ученого, создавшего рецепт; Название препарата; Дата создания; Ингредиенты необходимые для приготовления

Столбцы: *ScientistName, preparation, date(в формате ГГГГ-ММ-ДД), components(ингредиенты препарата через пробел)*

Разделитель *«#».*

*Кодировка: “utf-8”*

***Задача 1***

Перед началом создания программы для оптимизации поиска препаратов, необходимо удалить из списка ученых, которые выдают препарат за свой. Определить злоумышленника очень просто: оригинальный препарат имеет самую раннюю дату создания. Все что идет позже - фейк. Создайте новый файл с названием *scientist\_origin.txt(данные должны быть отсортированы по дате в порядке возрастания).* Вам также необходимо срочно найти всех подельников препарата Аллопуринол, так как их рецептура неверна и из-за этого пострадало много людей. Отправьте отчет полиции в формате:

“Разработчиками Аллопуринола были такие люди (результаты выведите в порядке возрастания даты):

<ФИО ученого> - <Дата создания>

…

Оригинальный рецепт принадлежит: <ФИО ученого>”

Для решения задачи можно воспользоваться встроенными сортировками или решением задачи 2.

*В задаче запрещено использование сторонних библиотек(Pandas и др)*

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка (для языка Python – PEP 257). После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория.*

***Задача 2***

Чтобы поиск препаратов в файле *scientist.txt* не занимал много времени отсортируйте данные по столбцу с датой в порядке возрастания, а также выведите пять самых ранних препаратов в формате: “<ФИО автора>: <название препарата>”

В задаче нельзя использовать сторонние сортировки, для решения воспользуйтесь быстрой сортировкой.

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория*

***Задача 3***

**Ввод: стандартный ввод**

**Вывод: стандартный вывод**

Напишите программу, которая по заданной дате выводит ученого в формате фамилия и инициаллы и его препарат, если учёных несколько то вывести всех. Вывод сделать в формате:

“Ученый <Фамилия И.О.> создал препарат: <препарат> - <дата>”, если по заданной дате ничего не найдено, то вывести “В этот день ученые отдыхали”

Программа должна постоянно запрашивать дату и остановиться только в случае ввода слова ‘эксперимент’.

Поиск необходимо осуществить с помощью двоичного поискав файле*scientist.txt***.**

Формат ввода даты: ДД.ММ.ГГГГ

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория*

***Задача 4***

Вам необходимо создать личные кабинеты для каждого ученого, чтобы каждый мог видеть информацию о себе, а так же иметь возможность быстрой отправки документов о новом изобретение. Для этого необходимо создать логины и пароли для каждого из ученых. Реализуйте методы/функции, которые будут генерировать логины и пароли для пользователей. Логин должен состоять из фамилии и инициалов, например, если ученого зовут Гайчеров Борис Вадимович, его логин должен выглядеть как Гайчеров\_БВ. Также для каждого пользователя необходимо сгенерировать пароль, пароль должен состоять из 10 символов, включать в себя заглавные, строчные буквы английского алфавита и цифры.

“Гайчеров Борис Вадимович, Мелоксикам, 1442-08-15, Chemical\_45 Chemical\_24 Chemical\_59” → “Гайчеров Борис Вадимович, Мелоксикам, 1442-08-15, Chemical\_45 Chemical\_24 Chemical\_59, Гайчеров\_БВ, PO891Lkov6”

*На вход подается файл scientist.txt, который необходимо записать в список, для каждого элемента сгенерировать логин и пароль, после чего дополнить список сгенерированными элементами. Последним этапом полученный список записать в новый scientist\_password.csv файл. Назвать эти столбцы “login” и “password”*

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория*

***Задача 5***

Количество новых ученых увеличивается в геометрической прогрессии, в связи с этим поиск по ФИО пользователя будет работать неэффективно. Необходимо составить хэш-таблицу, в которой будет выстроено соответствие ФИО и значения хэша ФИО. Необходимо сгенерировать хеш для каждого значения использовав представленный ниже алгоритм. Полученный хеш записать в начале строки. Назвать столбец “hash”

Для хэширования необходимо использовать следующий алгоритм.

1. Необходимо сгенерировать таблицу перестановок. В таблице находятся перемешанные числа от 0 до 1023 включительно.

2. Берем остаток от деления ASCII кода символа на 1024 – получаем индекс

3. По индексу достаем элемент из таблицы.

4. Складываем значения всех хешей и берем остаток от деления на 2048

5. Записываем значение в строку

*На вход подается файл scientist.txt результаты необходимо записать в новый scientist \_with\_hash.csv файл.*

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория*

***Задача 6***

Ваш код будет использоваться программистом, которого возьмут на работу, поэтому он должен быть правильно оформлен и выложен на GitHub. Весь написанный код должен быть задокументирован согласно стандартам документирования кода выбранного языка.

Также необходимо оформить README.md для Вашего репозитория. Пункты, которые должны быть описаны:

1. Название проекта

2. Описание проекта

3. Оглавление (необязательно)

4. Как установить и запустить проект

5. Как использовать проект