

Практические задания

1. В компьютерной игре герой (класс Hero) может перемещаться между двумя точками (метод move) различными способами: идти пешком, ехать на лошади, лететь и т. п. Реализовать классы, позволяющие пользователю выбирать и менять в ходе выполнения программы способ перемещения героя, используя паттерн “стратегия” ([strategy](#)). Продемонстрировать работу реализованных классов.
2. Написать аннотацию с целочисленным параметром. Создать класс, содержащий публичные, защищенные и приватные методы (2–3 каждого вида) с параметрами, аннотировать любые из них. Вызвать из другого класса все аннотированные защищенные и приватные методы столько раз, сколько указано в параметре аннотации. Вызывающий методы код не должен зависеть от количества и типов параметров этих методов.
3. Реализовать программу-переводчик.
 - 3.1. При запуске программы выполняется чтение словаря из файла, записанного в следующем формате:

слово или выражение / перевод
 - 3.2. Затем запрашивается пользовательский ввод в консоли и выполняется перевод введенного текста.
 - 3.3. Перевод осуществляется по следующим правилам:
 - регистр букв игнорируется
 - если искомого слова нет в словаре – выводится без перевода
 - если есть несколько подходящих вариантов, выбирается вариант с максимальной длиной левой части. Например:
 - словарь:
 - look | смотреть
 - look forward | ожидать
 - текст:
 - dog look to the window, dog look forward
 - перевод:
 - dog смотреть to the window, dog ожидать
 - 3.4. Результат перевода выводится в консоль
 - 3.5. Создать и применить пользовательские исключения:
 - `InvalidFileFormatException` см. пункт 3.1;
 - `FileReadException` файла не существует, нет доступа к файлу и т.д.

4. С использованием только Stream API реализовать следующие методы:
- метод, возвращающий среднее значение списка целых чисел;
 - метод, приводящий все строки в списке в верхний регистр и добавляющий к ним префикс «_new_»;
 - метод, возвращающий список квадратов всех встречающихся только один раз элементов списка;
 - метод, принимающий на вход коллекцию и возвращающий ее последний элемент или кидающий исключение, если коллекция пуста;
 - метод, принимающий на вход массив целых чисел, возвращающий сумму чётных чисел или 0, если чётных чисел нет;
 - метод, преобразовывающий все строки в списке в Map, где первый символ – ключ, оставшиеся – значение;

Курсовая работа

Разработать приложение с графическим интерфейсом для заданий 1–4. Для этого приложения должна быть реализована возможность выбора из списка любого приложения, ввод входных данных и его выполнение. Модифицировать задания 1–4 так, чтобы весь вывод происходил в текстовых областях, защищённых от редактирования. Предусмотреть для заданий:

- 2 - выбор файлов словаря и текста для перевода, возможность ручного ввода текста
- 4 - ввод входных данных для методов

Отчёт по курсовой должен содержать титульный лист, диаграмму классов и перечень выполненных работ.

Отправка на проверку

Рассмотрим взаимодействие на примере студента Test Student в репозитории bio и заданием «Rosalind»:

- После перехода по ссылке для своей группы, выберите себя в списке студентов

Join the classroom:

spbstu-bio-3540202/20201

To join the GitHub Classroom for this course, please select yourself from the list below to associate your GitHub account with your school's identifier (i.e., your name, ID, or email).

Can't find your name? [Skip to the next step →](#)

| Identifiers | |
|-----------------------------|---|
| Test Student | > |
| Алексеев Николай Николаевич | > |
| Ахмедов Джошгун Азер оглы | > |
| Богданова Ксения Сергеевна | > |
| Григоренко Сергей Андреевич | > |

- Примите задание к выполнению

spbstu-bio-3540202/20201

Accept the assignment —


Rosalind

Once you accept this assignment, you will be granted access to the `rosalind-AlekseiMaslakov` repository in the `spbstu-bio` organization on GitHub.

Accept this assignment

Имя репозитория будет состоять из двух частей – префикса rosalind и вашего имени на github.

- После этого начнёт создание вашего личного приватного репозитория
You accepted the assignment, **Rosalind** . We're configuring your repository now. This may take a few minutes to complete. Refresh this page to see updates.

 Your assignment is due by **May 23, 2023, 18:00**

Note: You may receive an email invitation to join [spbstu-bio](#) on your behalf. No further action is necessary.


Обратите внимание, в зависимости от настроек вашего аккаунта может потребоваться подтверждение через почту.

- Когда репозиторий будет готов вы получите уведомление


You're ready to go!

You accepted the assignment, **Rosalind**.

Your assignment repository has been created:

 <https://github.com/spbstu-bio/rosalind-AlekseiMaslakov>

We've configured the repository associated with this assignment ([update](#)).

 Your assignment is due by **May 23, 2023, 18:00**

Чтобы отправить задание на проверку:

- В репозитории нужно создать отдельную ветку на каждое сдаваемое задание, в качестве имени можно использовать lab_N, где N – номер лабораторной.
- Затем загружаете в ветку решение задачи и делаете pull request своей ветки в main.
- При наличии замечаний они будут описаны в качестве комментария в пул реквесте. Закрывать уже созданный пул реквест не нужно, все исправления, отправленные в соответствующую ветку, подтянутся автоматически.
- Если замечаний/вопросов нет, и ветка влита, то задача считается принятой.
- Решения, отправленные не по установленному алгоритму, проверяться и учитываться при выставлении зачёта не будут.

Проверка заданий будет проводиться в часы ваших практических занятий, все возникающие вопросы можно задать во время занятий голосом, текстом в MS Teams или по почте (maslakov_ap@spbstu.ru).

Для всех не сдавших решения задач до последнего занятия включительно, защита выполненных работ будет осуществляться в MS Teams.