**Syberry Academy: Тестовая задача. Rover v01**

**UPD 13 Апреля 11:30**:   
параметр для метода в C# поменялся   
название метода в C# поменялся   
поменяли условие вывода в случае нескольких правильных ответов   
поменяли условие движения ровера

**Дедлайн и условия проверки**

Вы читаете техническое задание тестовой задачи Syberry Academy.

**У вас есть семь дней на выполнение этой задачи.** Отсчет начинается с момента, когда мы прислали вам это письмо. Письмо с заданием вам пришло в четверг. Письмо с решением отправьте не позже 15:00 следующего четверга. Например, если задание пришло 8 апреля, то решение вы должны прислать не позже 15 апреля 15:00.

По нашей оценке на изучение задачи и на ее решение у вас уйдет от двух до семи дней. Планируйте свое время.

Мы не проверяем задачи, присланные после дедлайна. Если вы прислали решение в четверг в 15:01, это значит, что вы пропустили дедлайн. Такие задания дисквалифицируются.

Внимательно прочитайте требования к задаче. **Ваши решения проверяются автоматически, мы не будем вручную проверять ваш код.** Если ваше решение не соответствует условиям, скрипт его дисквалифицирует.

Мы ищем людей, которые умеют планировать свое время и разбираться в сложных задачах. Поэтому в этой задаче строгий дедлайн и строгие требования.

Мы понимаем, что задание сложное. Такое задание нужно для того, чтобы вы проверили себя. Если вы можете выполнить это задание за неделю, то вы справитесь и с заданиями в Академии, и на работе в Syberry. Если вы не смогли победить сложность этой задачи, то скорее всего вам надо еще немного поучиться. Если задача для вас неподъемная, то вам будет сложно учиться и работать.

Следующий набор в Академию будет в декабре, а еще один - весной 2022. Если вы не справились с задачей сейчас, вы сможете попробовать себя еще раз чуть позже.

Мы уверены, что эту задачу можно решить за 12-16 часов, даже если вы не знаете ничего о продвинутых техниках программирования. 80% вашего времени уйдет на обдумывание решения и чтение теории. Это нормально. Более того, это именно то, что мы ожидаем от наших студентов. Начинайте программировать только тогда, когда поймете задачу и смоделируете ее решение.

[И не уходите.](https://bureau.ru/soviet/20200424/)

**Задача**

Вы — инженер, проектирующий роверы-беспилотники. Вам надо спроектировать путь ровера по заранее известной местности с максимальной экономией заряда.

**Местность**

Вам пришли данные о местности в закодированном виде: фотография, сконвертированная в матрицу с числами. Одна матрица — это прямоугольный снимок размером х на y метров. Вот пример одной такой сконвертированной фотографии, на ней снимок в 100 на 100 метров:  
Фото 1:   
0 2 3 4 1  
2 3 4 4 1  
3 4 5 6 2  
4 5 6 7 1  
6 7 8 7 1

Числа показывают высоту над уровнем моря. 0 — это высота ровно на уровне моря, а, например, 4 — это 4 единицы над уровнем моря. На Фото 1 закодирован холм, пологий слева и резко обрывающийся справа.

Небольшой холмик выглядел бы вот так  
Фото 2:   
0 1 1 1 0  
1 1 3 1 1  
0 1 1 1 0  
0 0 0 0 0

А вот так: ложбина между двумя холмами  
Фото 3:   
1 1 2 3 4  
1 0 1 2 3  
2 1 1 1 2  
3 3 1 0 1  
4 3 1 1 0

На этих данных - скала или овраг, так как виден очень резкий перепад высот в середине снимка  
Фото 4:   
1 1 6 7 7  
1 1 6 7 8  
1 6 7 8 9

А на этом - маленькая ямка   
Фото 5:   
3 4 4 4 4 3  
3 2 1 1 1 4  
4 2 1 1 3 4  
4 4 2 2 3 4

Данные придут вам в виде матрицы с неотрицательными числами. Размер матрицы NxM.

**Ровер**

Ровер всегда движется из верхней левой точки [0][0] в правую нижнюю точку [N - 1][M - 1], где N и M - это длина и ширина матрицы. Это надо для того, чтобы разрезать фотографию на одинаковые куски, обработать их по-отдельности, а потом склеить весь путь.

У вашего ровера есть несколько ограничений:

1. Движение  
   Из любой точки ровер может двигаться только в четыре стороны: на север, юг, запад, восток. Ровер не может ехать по-диагонали — эта функция еще не реализована. Ровер не может вернуться в ту точку, в которой уже был.
2. Заряд  
   Ровер ездит на заряде. Вы знаете, что для ровера очень затратно подниматься и опускаться. Он тратит единицу заряда на само движение, и дополнительные единицы на подъем и спуск. Ровер бы вообще спокойно жил, если бы ездил по асфальту в Беларуси, тогда бы он тратил себе линейно заряд и в ус не дул, но жизнь его сложилась иначе.
3. Расход заряда  
   Заряд расходуется по правилу:  
   На 1 шаг ровер всегда тратит 1 единицу заряда. На подъем или спуск ровер тратит заряд, пропорциональный сложности подъема или спуска. Сложность подъема или спуска - это разница между высотами.

Например, в такой местности   
1 2  
1 5  
на путь из [0][0] в [0][1] ровер потратит 2 единицы заряд: 1 единица заряда на само движение, и еще 1 единицу заряда на подъем в [0][1]. А из [0][1] в [1][1] ровер потратит 4 единицы заряда: 1 единица на само движение, и 3 единицы (5 - 2) на подъем

Вам надо рассчитать путь ровера из верхей левой [0][0] точки в правую нижнюю [N - 1][M - 1] точку с минимальной тратой заряда.  
Вы не заранее знаете размер фотографии, которую будете обрабатывать, N и M - произвольные неотрицательные числа.

**План**

Сделайте план пути и планируемый расход в txt файле. Назовите файл path-plan.txt  
Для фотографии  
0 4  
1 3  
план будет такой:

path-plan.txt  
[0][0]->[1][0]->[1][1]  
steps: 2  
fuel: 5

Ровер едет из 0 в 1 в 3, сделает два шага, потратит 5 заряда. Если бы он поехал сначала в 4, потом в 3, он бы сделал то же количество шагов, но потратил бы 7 заряда. Оптимальный путь: 2 шага и 5 заряда.

Если на карте есть несколько вариантов пути, выберите любой из них.

**Требования**

Реализуйте функцию calculateRoverPath(map) по требованиям.  
Результат запишите в txt файл.  
Используйте [готовые файлы](/SyberryAcademy/syberry-academy-e07-test-task/-/tree/master/predefined-files) для решения. Не меняйте название файла, название методов и параметров.   
Напишите для себя .gitignor. Не добавляйте в репозиторий .idea, src, venv, DS\_STORE и подобные файлы. В репозитории должен лежать только файл с решением и README.md

**Java и C#**   
Class name: Rover   
Метод calculateRoverPath должен быть static   
Не добавляйте packages   
Версия Java 14+   
Версия C# 8.0+   
Не используйте System.out.\*, Console.Write\* и любые другие методы для выведения данных в консоль  
Используйте [готовые файлы](/SyberryAcademy/syberry-academy-e07-test-task/-/tree/master/predefined-files) для решения

**JS**   
Не используйте console.log и любые другие методы для выведения данных в консоль   
Используйте [готовые файлы](/SyberryAcademy/syberry-academy-e07-test-task/-/tree/master/predefined-files) для решения

**Ограничения**

Нельзя использовать библиотеки для реализации алгоритма. Реализовать алгоритм вы можете только своими силами.  
Можно использовать библиотеки для записи в файл.

**Совет**

Вы можете решать задачу так, как считаете нужным. Мы советуем применить [Dynamic programming](https://en.wikipedia.org/wiki/Dynamic_programming).

**Как сдать сделанную задачу**

* Создайте свой приватный репозиторий в Gitlab. Назовите его Name-Surname-Academy-Test-Task. Мы дисквалифицируем ваше решение, если репозиторий будет публичным. Это надо для того, чтобы все участники решали честно и не списывали.
* Мы будем смотреть на процесс вашей работы по вашим коммитам. [Пишите хорошие сообщения в коммитах.](https://chris.beams.io/posts/git-commit/)
* В репозитории должно быть два файла rover.js (or Rover.cs, Rover.java) и README.md.
* В файле Rover.\_\_ должен быть метод calculateRoverPath. Вы можете добавить любой другой функционал в этот файл, любые другие функции и комментарии. Главное, чтобы функция calculateRoverPath работала.
* Нам нужен доступ в ваш репозиторий, чтоб оценить вашу работу. Добавьте пользователя SyberryAcademy в ваш репозиторий, дайте ему права Maintainer. Не ставьте Expiration date.
* Линк на репозиторий с задачей пришлите нам на имейл [academy@syberry.com](mailto:academy@syberry.com). Тема письма: "Name Surname Syberry Academy%Language%", где %Language% - это язык, на котором вы писали ваш код. Например, "Jane Doe Syberry Academy Java"
* На почте стоит фильтр. Если вы неправильно напишете тему письма или если в письме не будет линка на гитлаб, то письмо автоматически попадет в спам, и мы его не увидим.

**Что дальше**

* Мы проверим вашу задачу: запустим сотни тестов, которые будут оценивать соответствие требованиям, качество решения, плагиат и сообщения в коммитах.
* Если ваша задача пройдет наши тесты, наш рекрутер напишет вам email с приглашением на техническое собеседование
* Если ваша задача не пройдет тесты, мы отправим вам автоматический email с отказом
* Мы проверяем ваш код в пятницу после вашего дедлайна, а приглашения на интервью и отказы рассылаем во вторник с 13:00 до 18:00. То есть, если вы успели к дедлайну и прислали задачу до четверга, то в пятницу наша система его проверит. В понедельник наш рекрутер обработает результаты, и во вторник пришлет вам ответ.
* Если вы не получили ответ во вторник после 18:00, напишите нам. Тема письма: "Name Surname Syberry Academy Нет ответа на тестовое". В письме приложите линк на гитлаб и вашу заявку.
* На почте стоит фильтр. Если вы неправильно напишете тему письма или если в письме не будет линка на гитлаб, то письмо автоматически попадет в спам, и мы его не увидим.

**Перед отправкой задания проверьте еще раз эти требования**

* Файл Rover.\_\_ лежит в вашем репозитории
* Репозиторий приватный
* Вы добавили пользователя SyberryAcademy в репозиторий и дали ему права Maintainer
* Вы запустили свое решение со всеми тестовыми данными и оно показало те же результаты, как и у нас
* Вы не используете методы вывода текста в консоль
* У вашего репозитория следующая структура:

Jane-Doe-Academy-Test-Task

Rover.cs

README.md

Keanu-Reeves-Academy-Test-Task

Rover.java

README.md

Henry-Cavill-Academy-Test-Task

rover.js

README.md

**Если вы нашли ошибку**

Чтобы поправить ошибку, сделайте мерж реквест в этот репозиторий с описанием ошибки. Assignee реквеста SyberryAcademy