1. У бібліотеці Rasterio для зчитування та запису растрових даних використовуються функції
   * rasterio.open()
   * rasterio.write()
   * rasterio.read()
   * rasterio.record()
   * rasterio.cut()
2. Яка функція в бібліотеці Rasterio дозволяє відкривати растрові файли?
   * rasterio.open()
   * rasterio.read()
   * rasterio.write()
   * rasterio.load()
   * rasterio.save()
3. Яка функція в бібліотеці Rasterio дозволяє записувати зображення до растрового файлу?
   * rasterio.write()
   * rasterio.record()
   * rasterio.save
   * rasterio.upgrade()
   * rasterio.open()
4. Яку інформацію можна зчитати за допомогою бібліотеки Rasterio?
   * Інформацію про географічний простір
   * Інформацію про координатну систему
   * Інформацію про тип об’єкту на зображенні
   * Інформацію про кількість пікселів у зображенні
   * Інформацію про зміну якості зображення у часі
5. Бібліотека Rasterio є потужним інструментом для
   * Роботи з геопросторовими даними, зокрема з растровими даними
   * Обробки та аналізу геометричних об'єктів
   * Виконання різних операцій з геоданими, такі як об'єднання, перетин, зріз тощо
   * Візуалізації геоданих на географічних картах
   * Знаходження географічних об’єктів на зображеннях, таких як річки, гірські вершини та інші
6. «Функція rasterio.plot.show() дозволяє відображати растрові дані в зручному для сприйняття форматі, а також можливість створювати зображення з декількох різних растрових даних». Яка з можливостей(функцій) бібліотеки Rasterio була щойно описана?
   * Візуалізація растрових даних
   * Робота з референційною системою координат
   * Обробка та аналіз растрових даних
   * Зчитування та аналіз растрових даних
   * Робота з метаданими
7. Бібліотека Rasterio надає багато функцій для обробки та аналізу растрових даних, таких як
   * фільтрація
   * рекласифікація
   * агрегація
   * розділення
   * перетворення
8. Бібліотека Rasterio дозволяє візуалізувати растрові дані за допомогою різних інструментів, таких як
   * Matplotlib
   * Rasterio.plot
   * NumPy
   * Bz2
   * Marshal
9. При зчитуванні растрових даних, Rasterio дозволяє
   * Здійснювати операції з різними банками кольорів
   * Проводити аналіз системи координат
   * Редагувати зображення
   * Розпізнавати об’єкти на зображенні
   * Аналізувати частоту пікселів
10. Яка інформація може міститись в метаданих растрового файлу?
    * Інформація про растровий файл, така як координати, система координат, розмір, тип даних
    * Інформація про кольорову гаму зображення
    * Інформація про формат зображення
    * Інформація про розмір зображення
    * Інформація про розташування зображення на екрані
11. Опишіть результат виконання даного прикладу:

import rasterio.plot

with rasterio.open('image.tif') as src:

hist = src.read(1).flatten()

plt.hist(hist, bins=50)

plt.title('Image histogram')

plt.xlabel('Pixel values')

plt.ylabel('Frequency')

plt.show()

* + Ми відкрили растровий файл та обчислили гістограму його зображення з використанням бібліотеки Matplotlib
  + Ми відкрили растровий файл та візуалізували його зображення з використанням кастомної кольорової мапи
  + Ми відкрили растровий файл та визначили вікно для візуалізації зображення
  + Ми виконали фільтрацію растрових даних з використанням операції гладкої середньої або медіани
  + Ми виконали розрахунок середнього значення та стандартного відхилення

1. Яка функція бібліотеки Rasterio дозволяє отримати систему координат растрового файлу?
   * src.crs
   * src.bounds
   * src.profile
   * src.transform
   * rasterio.mask.mask
2. Яка функція бібліотеки Rasterio надає можливість виконувати розрахунки над растровими даними, такі як розрахунок середнього значення або стандартного відхилення?
   * rasterio.mask.mask
   * src.bounds
   * src.transform
   * src.profile
   * rasterio.windows.Window
3. Що виконує функція src.profile ?
   * Повертає словник, що містить всі метадані растрового файлу, включаючи систему координат, границі тощо
   * Повертає матрицю перетворення, що використовується для перетворення піксельних координат в географічні координати
   * Повертає границі растрового файлу у форматі (ліва координата, нижня координата, права координата, верхня координата)
   * Проводить розрахунок середнього значення або стандартного відхилення
   * Виконує фільтрацію растрових даних з використанням операції гладкої середньої або медіани
4. Яка функція повертає матрицю перетворення, що використовується для перетворення піксельних координат в географічні координати?
   * src.transform
   * src.bounds
   * src.profile
   * src.mask
   * rasterio.windows.Window
5. Яка функція повертає границі растрового файлу у форматі (ліва координата, нижня координата, права координата, верхня координата)?
   * src.bounds
   * src.read()
   * src.open()
   * src.profile
   * src.mask
6. Що виконує функція rasterio.mask.mask ?
   * Надає можливість виконувати розрахунки над растровими даними, такі як розрахунок середнього значення або стандартного відхилення
   * Виконує фільтрацію растрових даних з використанням операції гладкої середньої або медіани
   * Проводить розрахунок середнього значення або стандартного відхилення
   * Повертає словник, що містить всі метадані растрового файлу, включаючи систему координат, границі тощо
   * Повертає границі растрового файлу у форматі (ліва координата, нижня координата, права координата, верхня координата)
7. Опишіть результат виконання даного прикладу:

import rasterio.plot

with rasterio.open('image.tif') as src:

cmap = plt.cm.get\_cmap('RdYlBu')

rasterio.plot.show(src, cmap=cmap)

* + Ми відкрили растровий файл та візуалізували його зображення з використанням кастомної кольорової мапи
  + Ми відкрили растровий файл та обчислили гістограму його зображення з використанням бібліотеки Matplotlib
  + Ми відкрили растровий файл та визначили вікно для візуалізації зображення
  + Ми продемонстрували візуалізацію зображення з використанням бібліотеки Matplotlib
  + Ми відкрили растровий файл та знаходили його мінімальне та максимальне значення

1. Яка дія відбувається в даному рядку коду?

histogram = np.histogram(data, bins=100, range=(min\_value, max\_value))

* + Обчислення гістограми значень пікселів
  + Відкриття растрового файлу та його читання
  + Знаходження граничних значень
  + Знаходження мінімального значення
  + Знаходження максимального значення

1. Що виконує даний код?

thresholds = [0.1, 0.3, 0.5, 0.7, 0.9]

boundaries = np.percentile(data, thresholds)

* + Знаходження граничних значень
  + Відкриття растрового файлу та його читання
  + Обчислення гістограми значень пікселів
  + Знаходження мінімального значення
  + Знаходження максимального значення