5 grudnia 2025

Kurs rozszerzony języka Python

Lista 9.

Na pracownię należy wykonać jedno z poniższych zadań. Każde zadanie jest warte 5 punktów.

Zadanie 1.

Korzystając z biblioteki `matplotlib` zaprogramuj własnego *snake'a*: najpierw poukładaj na wykresie losowe kwadraty, a następnie zaprogramuj animowanego węża który będzie losowo wędrował po wykresie. Zabawa kończy się, gdy głowa węża natknie się na własny ogon lub jeden z kwadracików. Można przyjąć, że liczba segmentów węża jest ustalana zmienną globalną.

Zadanie 2.

Gra w życie to przykład *automatu komórkowego* wymyślonego przez Johna Conwaya. Reguły działania są opisane m.in. w Wikipedii (`https://pl.wikipedia.org/wiki/Gra_w_%C5%BCycie`). Korzystając z pakietu `matplotlib` zaprogramuj symulację takiego automatu.

Co prawda gra w życie toczy się na nieskończonej planszy, ale możesz przyjąć jakieś sensowne ograniczenia. Jako stan początkowy może być ustalony bądź losowy.

Zadanie 3.

Inną odmianą automatu komórkowego jest *mrówka Langtona* (`https://pl.wikipedia.org/wiki/Mr%C3%B3wka_Langtona`). Zaprogramuj w `matplotlib` symulację takiego automatu.

Zadanie 4.

Zaprogramuj zbiór funkcji modyfikujących zadany obraz tak, aby modyfikacja była

- istotnie *weselsza*, tj. miała mniej koloru szarego (tj. koloru spełniającego warunek $R \approx G \approx B$). Zaproponuj jakąś strategię wskazywania koloru zastępczego na podstawie kolorów wokół obszarów szarych;
- istotnie *smutniejsza*, dodając więcej szarości do pierwotnego obrazka.

Pokaż obraz pierwotny i zmodyfikowany na jednym wykresie. Dla chętnych: zamiast dwóch obrazów można zrobić animację stopniowego przechodzenia obrazu pierwotnego do docelowego.

Zadanie 5.

Zaprogramuj funkcję, która dla zadanego obrazu skonstruuje obraz zawierający sam *obrys* przedmiotów lub osób znajdujących się na obrazie. Proponowana strategia jest taka: poszukujemy granicy między przedmiotem a tłem lub innymi przedmiotami szukając istotnych zmian kolorów między sąsiadującymi pikselami.

Zaproponuj przynajmniej dwie realizacje *istotnej zmiany kolorów* między pikselami. Zaprezentuj wyniki działania na jednym wykresie (tj. obrazu pierwotnego i obrysów uzyskanych różnymi metodami).

Zadanie 6.

W zadaniu tym zabezpieczymy nasze obrazki przed przeglądaniem przez niepowołane osoby poprzez ich *zaszumienie*. Algorytm jest następujący:

- generujemy losowy obraz o tych samych wymiarach co obraz do zabezpieczenia. Można to zrobić tak jak na wykładzie, choć można to zrobić też wywołując odpowiednią funkcję pakietu `NumPy`. Plik ten zachowujemy w sekretnym miejscu;

- za pomocą tego losowego obrazka modyfikujemy oryginalny obraz; proszę zaproponować przynajmniej dwa sposoby i porównać uzyskane wyniki;
- zaprogramuj dodatkowo funkcję odszyfrowującą obraz zaszumiony, korzystając z zapisanego wcześniej losowego obrazka.

Marcin Młotkowski