

Zadanie rekrutacyjne na stanowisko: Python Developer

Dla potrzeb zadania załóżmy, że masz do wykorzystania źródło danych:

```
class Source:
    def __init__(self, source_shape: tuple):
        self._source_shape: tuple = source_shape

    def get_data(self) -> np.ndarray:
        rows, cols, channels = self._source_shape
        return np.random.randint(
            256,
            size=rows * cols * channels,
            dtype=np.uint8,
        ).reshape(self._source_shape)
```

Zaproponuj architekturę Producent-Konsument, w której:

- ☐ **Producent** (zrealizowany jako wątek) ma do dyspozycji źródło danych o rozmiarze szerokość=1024 px, wysokość=768 px, liczba kanałów=3. Co 50 ms producent pobiera nowe dane z Source i wrzuca je do kolejki **A**,
- ☐ **Konsument** (zrealizowany jako wątek) ściąga dostępne dane z kolejki **A** i dokonuje następujących operacji przetwarzania:
 - ☐ dwukrotne zmniejszenie rozmiaru obrazu,
 - ☐ aplikuje filtr medianowy o kernelu 5x5.

Po zakończeniu przetwarzania, nowy obraz wrzucony jest do kolejki **B**.

W main zademonstruj działanie programu, w którym po przetworzeniu 100 ramek danych ma nastąpić zatrzymanie. Wszystkie dane z kolejki **B** mają być zapisane w katalogu o nazwie *processed* w formacie png. Kod źródłowy zadania proszę umieścić w wybranym przez siebie systemie kontroli wersji. Dozwolone jest wykorzystanie bibliotek.