

# Programowanie zaawansowane – zaliczenie ćwiczeń (1/2)

prowadzący: dr inż. Piotr Bobiński, [piotr.bobinski@warszawa.merito.pl](mailto:piotr.bobinski@warszawa.merito.pl)

## Zasady realizacji projektów

- Projekty są realizowane w zespołach maksymalnie 3-osobowych
- Podstawą zaliczenia i ocenienia jest:
  - Przesłanie/udostępnienie plików źródłowych oraz dokumentacji w pliku README.md (archiwum lub link do **repozytorium na GitHub umieszczony w Moodle**)
  - **Rozwiązanie zamieszcza jedna osoba z zespołu**
- Technologia – **dowolna**
- **Termin (ostateczny) – 4 czerwca (23:59)**

## Zadanie projektowe

Napisz aplikację pracującą w modelu klient-serwer, spełniającą następujące założenia:

- Aplikacja definiuje przynajmniej trzy klasy, z których każda zawiera przynajmniej dwa pola i odpowiednie metody (konstruktor, toString, equals itp.).
- Serwer przy starcie tworzy po cztery obiekty każdej klasy, inicjalizując je innymi danymi i umieszcza te obiekty w mapie, gdzie kluczem będzie nazwa klasy oraz numer porządkowy, *np. dla klasy Kot i dwóch instancji mamy: kot\_1, kot\_2*.
- Serwer ma obsługiwać wielu klientów równocześnie, maksymalną liczbę klientów należy ograniczyć w postaci stałej, np. MAX\_CLIENTS, nadmiarowi klienci powinni otrzymać informację o odmowie obsługi, zadbaj o losowe opóźnienia przy obsłudze klientów.
- Klient łączy się z serwerem, wysyła swoje id (liczba) i otrzymuje status połączenia: OK lub REFUSED (w przypadku gdy została przekroczona maksymalna liczba obsługiwanych klientów), serwer rejestruje id klienta, którego obsługuje lub odrzuca i wypisuje tę informację na konsoli.
- W przypadku odpowiedzi REFUSED klient kończy działanie, a w przypadku odpowiedzi OK:
  - klient prosi o przesłanie kolekcji obiektów konkretnej klasy,
  - serwer pobiera obiekty z mapy, tworzy ich kolekcję i odsyła go do klienta w postaci zserializowanej oraz wypisuje na konsoli co (jakie obiekty) i komu (id) przesłał,
  - gdy obiektów nie ma w mapie, serwer odsyła dowolny obiekt, klient powinien obsłużyć błąd rzutowania przy odczycie obiektu, o który poprosił,
  - klient odbiera kolekcję i obsługuje ją strumieniowo, wypisując jej obiekty na konsoli wraz ze swoim id, gdy nie potrafi zrzutować, wypisuje o tym informację na konsoli,
  - klient powtarza powyższe punkty kilka razy dla różnych obiektów i kończy działanie.

Sprawdź powyższą funkcjonalność:

- Uruchom serwer.
- Utwórz kilka klientów, niech każdy ma inne id i pyta o inne obiekty i je uruchom, ręcznie albo w oddzielnej klasie jako wątki (ważne będą tu losowe opóźnienia w obsłudze aby zaobserwować co się dzieje w systemie).

Elementy wpływające na ocenę: obiekty, obsługa strumieni danych, wątki, komunikacja sieciowa, obsługa błędów, dokumentacja.