

LISTA nr 3

Przedmiot:	Algorytmy i Struktury Danych, laboratorium
Prowadzący:	Anna Gilewska
Rok:	2018/2019, semest letni
Punkty do zdobycia:	10 pkt
Termin oddania:	dla grupy wtorek 15.15-16.45: <u>26 marzec 2019</u> dla grupy czwartek 7.30-9.00: <u>28 marzec 2019</u>

Zasady i wskazówki:

1. Każde rozwiązanie należy implementować w postaci osobnej klasy. Wskazane jest też zaimplementowanie metody, która będzie uruchamiała testy oraz prezentację wyników.
3. Rozwiązania powinny zwracać wyniki testów w postaci zmiennej, a być wypisywane na ekran tylko za pomocą funkcji `toString`, którą należy przeciążyć.
4. Wszystkie występujące w programie wyjątki (także własne) należy obsługiwać.
5. W ramach ćwiczenia dobrej praktyki nauczyć się uruchamiać pisane programy z linii komend.
6. Mile widziana możliwość wprowadzania parametrów w postaci pliku.
7. Po prezentacji, zadanie należy załączać jako 1 plik (zawierający wszystkie spakowane pliki niezbędne do działania rozwiązania) na ePortal. Plik proszę oznaczyć wedle schematu [ALG][LAB2]<numer indeksu><imię><nazwisko>
8. Należy przesłać cały projekt, spakowany, napisany w formie, w której kompilacja i uruchomienie funkcji `main()` zaprezentuje pełen zakres działania, tj. wykorzystane zostaną wszystkie funkcje. Np. dla funkcji usunięcia elementu z listy, należy udokumentować stan przed usunięciem i po usunięciu (wypisać).

ZADANIA – STOSY I KOLEJKI

Implementowane kolejki i stosy powinny implementować przedstawione na wykładzie (lub w skrypcie) interfejsy `Queue` lub `Stack`

Zadanie 1 (3 pkt)

Należy zaimplementować kolejkę ograniczoną przechowującą dowolną stworzoną przez siebie klasę, a dodatkowo przetestować działanie, zwracając uwagę na sytuacje brzegowe przepełnienie kolejki, pusta kolejka, itp.

Zadanie 2 (3 pkt)

Należy zaimplementować kalkulator korzystający z ONP oparty na stosie. Kalkulator ma wykonywać następujące operacje: dodawanie, odejmowanie, mnożenie, dzielenie, potęgowanie, pierwiastkowanie i logarytmowanie.

Zadanie 3 (4 pkt)

Należy zaimplementować stos tonący przechowujący dowolną stworzoną przez siebie klasę (może być z zadania 1). Struktura będzie przyjmować jako parametr liczbę naturalną N . Jeżeli liczba elementów w stosie będzie równa N to dodanie elementu do stosu powinno spowodować usunięcie najstarszego elementu.

Zastanów się i do implementacji wybierz strukturę, która będzie zapewniała efektywne działanie. Wybór uzasadnij.