

- I. [1 pkt.] Utwórz klasę `Drzewo` z polami `boolean wiecznieZielone`, `int wysokosc` oraz `String przekrojDrzewa`, następnie zaimplementuj konstruktory inicjujące pola i metodę `String toString()` zwracającą informacje o drzewie.
- II. [1 pkt.] Utwórz klasę `DrzewoIglaste` dziedziczącą po klasie `Drzewo`, oraz definiującą pola `int iloscIgiel` i `double dlugoscSzyszki`. Następnie przygotuj konstruktor inicjujący wszystkie pola oraz metodę `String toString()` zwracającą informacje o drzewie wraz z wartościami pól opisujących drzewo iglaste.
- Utwórz klasę `DrzewoLisciaste` dziedziczącą po klasie `Drzewo`, oraz definiującą pole `int ksztaltLiscia` i metodę `String toString()` zwracającą informacje o drzewie.
- Utwórz klasę `DrzewoOwocowe` dziedziczącą po klasie `DrzewoLisciaste`, oraz definiującą pole `String nazwaOwoca`, podobnie jak w poprzednich zadaniach uzupełnij klasę o konstruktor i funkcjonalną metodę `toString`.
- III. [1 pkt.] Zdefiniuj trzy metody, z których każda podniesie inny wyjątek:
- `podniesArrayIndexOutOfBoundsException()`;
 - `podniesFileNotFoundException()`;
 - `podniesException()`, z komunikatem *zła jednostka*.
- Obsłuż wszystkie możliwe wyjątki w stosownej klauzuli `catch`
- IV. [1 pkt.] Utwórz klasę `Rakieta`, zbudowaną z pól: `String nazwa` i `int wagaPaliwa`, konstruktora inicjującego pola oraz następujących metod: `zatankuj` - ładującą losową ilość paliwa, oraz `start` - rozpoczynającą procedurę startową. W ramach kontroli przedstartowej metoda `start` sprawdzi stan paliwa i jeżeli jest on mniejszy niż 1000, podniesiony zostanie wyjątek `Exception` z komentarzem 'start anulowany - za mało paliwa'.