

Napisz aplikację, która będzie służyła do obsługi logistycznej morskiego portu w zakresie terminalu przeładunkowego dla statków transportowych przystosowanych do przewozu kontenerów. Aplikacja będzie służyła do rozładunku i załadunku kontenerów z/na statek. W przypadku rozładunku kontenery mogą trafić do magazynu lub bezpośrednio na wagon transportowy.

Każdy statek ma inną pojemność i nośność definiowaną przez:

- maksymalną liczbę kontenerów z ładunkiem toksycznym lub wybuchowym możliwych do załadowania w ramach statku
- maksymalną liczbę ciężkich kontenerów
- maksymalną liczbę kontenerów wymagających podłączenia do sieci elektrycznej
- maksymalną liczbę wszystkich kontenerów
- maksymalną ładowność wagową statku

Dodatkowo każdy statek posiada podstawowe informacje o sobie (nazwę, port macierzysty, lokalizacja źródłowa transportu oraz docelowa). Każdy statek również posiada swój unikatowy numer identyfikacyjny nadawany automatycznie podczas tworzenia obiektu.

W przypadku ładowności kontenerów toksycznych i wybuchowych - wartość ta jest wspólna dla obu typów kontenerów, co oznacza, iż jeśli w statku zdefiniowano np. trzy miejsca na kontenery z materiałem wybuchowym lub toksycznym, to miejsca te mogą być załadowane odpowiednio przez:

- 3 kontenery z materiałami wybuchowymi
- 2 kontenery z materiałami wybuchowymi i 1 kontener z materiałami toksycznymi
- 1 kontener z materiałami wybuchowymi i 2 kontenery z materiałami toksycznymi
- 3 kontenery z materiałami toksycznymi

Każdy z typów kontenerów posiada inny zestaw cech (np. nadawca, tara, zabezpieczenia, waga netto, waga brutto, informacje o certyfikatach, itp.). W przypadku każdego typu kontenera należy wymyślić co najmniej jedną unikalną cechę na wskroś przez wszystkie typy kontenerów. Dodatkowo każdy kontener posiada swój unikatowy numer identyfikacyjny nadawany automatycznie podczas tworzenia obiektu.

Mamy do dyspozycji między innymi:

- kontener podstawowy,
- kontener ciężki, będący rodzajem kontenera podstawowego,
- kontener chłodniczy, będący rodzajem kontenera ciężkiego, wymagający podłączenia do sieci elektrycznej statku
- kontener na materiały ciekłe, będący rodzajem kontenera podstawowego
- kontener na materiały wybuchowe, będący rodzajem kontenera ciężkiego
- kontener na materiały toksyczne, będący rodzajem kontenera ciężkiego, który dostępny jest w dwóch wersjach: kontener na toksyczne materiały sypkie oraz kontener na ciekłe materiały toksyczne, który nie tylko jest rodzajem kontenera ciężkiego, ale również posiada cechy kontenera na materiały ciekłe.

W ramach statku jesteśmy ograniczeni nie tylko maksymalną liczbą kontenerów, ale również maksymalną ładownością wagową statku, w związku z czym przed załadowaniem kolejnego kontenera musimy być pewni, czy kolejny ładunek nie przekroczy dopuszczalnej bezpiecznej nośności statku. Nie ma potrzeby tworzenia wyspecjalizowanego algorytmu załadunkowego (np. czy statek się nie zacznie przechylać w związku z nierównomiernym załadunkiem).

W ramach aplikacji musimy mieć m.in. możliwość z poziomu menu stworzenia nowego statku oraz kontenerów każdego typu, załadowania kontenera na statek oraz rozładowania kontenera na wagon lub do magazynu.

W sytuacji, gdy rozładujemy kontener na wagon kolejowy, zakładamy, że jeden skład może pomieścić 10 kontenerów. Po zapełnieniu tego, należy odczekać 30 sekund na odjazd aktualnego składu i przyjazd kolejnego.

Magazyn przeładunkowy posiada pewną ograniczoną liczbę kontenerów jakie może pomieścić, zdefiniowaną podczas tworzenia obiektu typu *Magazyn*. W ramach magazynu wolno przechowywać każdy typ kontenera, jednak z zastrzeżeniem, iż kontenery z materiałami wybuchowymi mogą być magazynowane jedynie przez 5 dni, kontenery z płynnymi materiałami toksycznymi przez 10 dni, zaś kontenery z sypkimi materiałami toksycznymi maksymalnie przez 14 dni.

W związku z powyższym, należy zaimplementować mechanizm upływającego czasu za pośrednictwem wątków. Wątek powinien co 5 sekund przesuwac datę o 1 dzień do przodu, symulując upływ czasu. Równolegle powinny być sprawdzane kwestie magazynowe i w przypadku, gdy magazynowany kontener przekraczałby maksymalny czas magazynowania, zostawałby on w związku z decyzjami celnymi dot. bezpieczeństwa składowania i magazynowania towarów utylizowany poprzez wywiezienie z terenu portu, a następnie zapomniany z punktu widzenia aplikacji. Należy również zaimplementować 30-sekundowe odliczanie po zapełnieniu składu kolejowego do przyjazdu kolejnego składu, umożliwiającego dalsze rozładowywanie kontenerów na wagony.

W sytuacji, gdy kontener zostanie zutylizowany, nadawca otrzymuje ostrzeżenie w postaci obiektu wyjątku typu *IrresponsibleSenderWithDangerousGoods* zawierającego informacje dot. kontenera, tj. daty przybycia do magazynu, daty utylizacji oraz unikalnego numeru identyfikacyjnego). W przypadku, gdy nadawca uzbiera w ramach dostaw dwa ostrzeżenia, to w przypadku kolejnych kontenerów wymagających magazynowania, takowy towar nie zostanie przyjęty i zostanie na statku transportowym z nakazem odesłania do nadawcy.

Nadawca posiada ponad wymienione wcześniej informacje takie dane jak imię, nazwisko, pesel, adres, datę urodzenia (data urodzenia nie jest przechowywana specjalnie w ramach pola klasy - realizujemy ją w postaci metody, która wyświetli datę urodzenia jako obiekt klasy *LocalDate* na podstawie numeru PESEL).

W ramach każdego statku musimy zapewnić możliwość rozładowania i załadowania kontenera, jak i wypuszczenia statku w dalszą drogę z portu. Analogicznie w przypadku magazynu musimy zapewnić co najmniej metody umożliwiające magazynowanie kontenera, wyjęcie kontenera z magazynu i załadowanie na wskazany statek, sprawdzenie aktualnego stanu magazynowego wyświetlenie zawartości magazynu oraz manualną utylizację wskazanego kontenera.

Należy zapewnić trwałość aplikacji, tj. umożliwić zapis aktualnego stanu całego portu do pliku tekstowego (z uwzględnieniem wszystkich informacji z aplikacji, takich jak m.in. dane nadawców, statków, magazynu, kontenerów oraz kompletu informacji o nich). Zapisane w pliku powinny być zapisane przejrzystie i czytelnie dla człowieka z zachowaniem poniższych reguł:

- Kontenery znajdujące się na statkach powinny być posortowane rosnąco według wagi.
- Statki powinny być posortowane malejąco względem nazwy statku
- Kontenery magazynowane powinny być posortowane malejąco względem daty rozpoczęcia magazynowania (od najdłużej magazynowanych do najkrócej), a w sytuacji, gdy czas magazynowania będzie taki sam, to kolejnym kryterium sortującym jest nazwa nadawcy.

### *Nie wolno korzystać z serializacji!*

Zapisane dane do pliku muszą być wczytywane podczas kolejnego uruchomienia aplikacji, aby można było rozpocząć pracę od ostatniego zapisanego stanu. Można posłużyć się wieloma plikami w tym celu.

W programie należy obsłużyć wszystkie powstające wyjątki i odpowiednio zakomunikować użytkownikowi następującą sytuację bez potrzeby przerywania i uruchamiania na nowo programu.

Zasada działania programu:

- W metodzie *main* należy utworzyć port z co najmniej pięcioma statkami różnego rodzaju załadowanymi różnymi kontenerami. Dodatkowo należy w magazynie umieścić kilka kontenerów różnego typu i rodzaju.
- Po uruchomieniu programu, użytkownik powinien mieć możliwość za pośrednictwem konsoli poleceń wywołania wszystkich wspomnianych funkcjonalności. Są to m.in (lista niekompletna):
  - Zakończenia programu w dowolnym momencie
  - Zładowania wskazanego kontenera ze statku na wagon kolejowy/do magazynu
  - Załadowania na wskazany statek wskazanego kontenera
  - Czytelnego wypisania informacji o statku i znajdujących się na nim kontenerach
  - Czytelnego wyświetlenia stanu magazynowego wraz z czasami pozostałymi do utylizacji kontenerów przechowujących materiały wybuchowe lub materiały toksyczne
  - Utylizacji wskazanego kontenera z magazynu
  - Stworzenia nowego statku o wskazanych parametrach
  - Stworzenia nowego kontenera wskazanego typu o wskazanych parametrach
  - Wykonania polecenia zapisującego aktualny stan portu

*Projekt opiera się o materiał z zakresu PPJ oraz GUI.*

*Uwaga:*

- *W przypadku otrzymania projektu ze znacznymi brakami w implementacji lub niekompilującego się, wynikiem za taki projekt będzie 0 pkt.*
- *Brak znajomości dowolnej linii kodu lub plagiat skutkować będzie niezaliczeniem przedmiotu bez możliwości poprawy.*
- *W ocenie projektu poza praktyczną i merytoryczną poprawnością będzie brana również pod uwagę optymalność, jakość i czytelność napisanego przez Państwa kodu.*
- *Ważną częścią projektu jest wykorzystanie między innymi: dziedziczenia, kolekcji, interfejsów lub klas abstrakcyjnych, lambda-wyrażeń, typów generycznych, dodatkowych funkcjonalności lub struktur oraz innych elementów charakterystycznych*