

Uwaga:

- nie używaj kontenera *vector*

Tematyka: konstruktory, destruktor, akcesory, funkcje zaprzyjaźnione, przeciążanie operatorów

Stwórz klasy:

Pasazer	Autokar
<p><i>private:</i></p> <ul style="list-style-type: none"> • imię • nazwisko • rok urodzenia • licznik (<i>static</i>) <p><i>public:</i></p> <ul style="list-style-type: none"> • konstruktory (domyśl., param., kopiuj.) • destruktor • akcesory (getter, setter) • wypisz() <p>opis pól klas: licznik (<i>static</i>) – określa aktualnie istniejącą liczbę pasażerów</p>	<p><i>private:</i></p> <ul style="list-style-type: none"> • nazwa autokaru • cel podróży • cena biletu • limit miejsc • lista pasażerów <p><i>public:</i></p> <ul style="list-style-type: none"> • dodajPasażera() <p>opis pól klas: lista pasażerów – tablica wskaźników na obiekty klasy <i>Pasazer</i> o rozmiarze równym limitowi miejsc</p>

Opis wybranych metod:

Konstruktory – mają wyświetlać informację o tym, jaki konstruktor został wywołany i jakiej klasy obiekt utworzono. Dodatkowo, ma pokazywać licznik obiektów.

Konstruktor kopiujący – ma umożliwiać przypisanie dowolnego nazwiska (nowe nazwisko jako argument).

Destruktor – ma pokazywać licznik obiektów.

wypisz() – wyświetla dane obiektu.

dodajPasażera() – dodaje obiekt klasy *Pasazer* do tablicy (ustawia wskaźnik z tablicy listy pasażerów na obiekt *Pasazer*). Metoda ma sama wyszukiwać wolne miejsce, tzn. nie może przyjmować indeksu tablicy, jako argumentu.

- Utwórz obiekt domyślny klasy *Pasazer*. Uzupełnij i wypisz jego dane przy życiu akcesorów.
- Utwórz obiekt klasy *Pasazer* przy użyciu konstruktora parametryzującego. Wyświetl jego dane przy użyciu funkcji *wypisz()*.
- Utwórz kopię dowolnego obiektu przy użyciu konstruktora kopiującego. Wyświetl jego dane przy użyciu funkcji *wypisz()*.
- Utwórz kopię dowolnego obiektu, ale niech konstruktor kopiujący przypisze inne nazwisko (nazwisko podaj jako argument). Wyświetl jego dane przy użyciu funkcji *wypisz()*.
- Stwórz tablicę dynamiczną czterech obiektów klasy *Pasazer* o dowolnych polach. Wyświetl dane obiektów.
- Napisz zaprzyjaźnioną z klasą *Pasazer* funkcję *sortuj()*, która posortuje obiekty z tablicy względem wieku malejąco. Posortuj tablicę. Wyświetl posortowaną tablicę z wykorzystaniem przeciążonego operatora „<<”.
- Usuń tablicę dynamiczną obiektów i wyświetl licznik.
- Stwórz obiekt klasy *Autokar* o dowolnych polach. Wyświetl informacje o obiekcie (bez listy pasażerów).
- Przeciąż operator inkrementacji (pre i post) „++”. Niech użycie operatora „++” na obiekcie klasy *Autokar* spowoduje zwiększenie ceny biletu o 1. Zwiększ cenę biletu o 2 przy użyciu przeciążonego operatora (pre i post). Wyświetl cenę biletu przed i po podwyżce.
- Dodaj cztery utworzone wcześniej obiekty klasy *Pasazer* do listy pasażerów korzystając z metody *dodajPasażera()*. Wyświetl listę pasażerów autokaru.
- Stwórz funkcję **globalną**, która usunie wybranego pasażera z wybranego autokaru. Usuń dowolnego pasażera z listy pasażerów utworzonego wcześniej autokaru. Wyświetl listę pasażerów autokaru.

Punktacja:

[illegible]