

Rząd 3, Kolej.

**Pamiętaj: podstawą do rozpoczęcia sprawdzania jest poprawna kompilacja programu i przesłanie go na Githuba.**

0. Stwórz projekt aplikacja konsolowa, zapisz go w folderze na Pulpicie, nazwij go Poprawa+swój numer indeksu np. poprawa130000.

1. Stwórz abstrakcyjną klasę PojazdSzynowy. W tej klasie wykonaj czynności:

a) dodaj pola marka (string) i masa (int) z mod. protected

b) stwórz konstruktor z parametrami(string, int), które pobrane parametry podstawy jako pola w klasie

c) dodaj deklarację dwóch abstrakcyjnych metod:

- ObliczKoszt() z typem zwracanym double

- Info() z typem zwracanym void

2. Stwórz klasę Lokomotywa dziedziczącą z klasy PojazdSzynowy. W nowoutworzonej klasie wykonaj czynności:

a) przesłoń metodę ObliczKoszt() tak aby zwracała liczbę równą  $masa * 34$ ;

b) przesłoń metodę Info() tak, aby wyświetlała na konsoli informacje o lokomotywie np. Lokomotywa, model: WZ2, masa: 55000

3. Stwórz klasę Wagon dziedziczącą z klasy PojazdSzynowy. W nowoutworzonej klasie wykonaj czynności:

a) dodaj prywatne pole ładunek

b) przesłoń metodę ObliczKoszt() tak aby zwracała liczbę 3000 jeśli ładunek jest pusty (jest równy null) albo  $3000 + masa$  jeśli ładunek jest różny od pustego (null)

c) przesłoń metodę Info() tak, aby wyświetlała na konsoli informację o wagonie np. Wagon, model MZ23, masa: 22000.

4. W klasie Program i metodzie Main wykonaj czynności:

a) stwórz listę ciuchcia na obiekty typu PojazdSzynowy

b) na listę dodaj 10 różnych obiektów

c) wypisz za pomocą metody Info() elementy z listy ciuchcia na konsoli

d) oblicz łączny koszt wszystkich elementów z listy ciuchcia i wypisz go na konsoli.

5. Stwórz interfejs IKoszt i dodaj w nim deklarację metody ObliczKoszt() zwracającą typ double.

6. Podepnij interfejs IKoszt do klasy PojazdSzynowy (ew. popraw kod jeśli coś nie gra).

7. Stwórz klasę Stacja. W nowej klasie wykonaj czynności:

a) dodaj prywatne pole ileDzienniePociagow typu int

b) do klasy podepnij interfejs IKoszt() i zaimplementuj właściwą metodę tak, aby zwracała liczbę obliczoną wg wzoru:  $\text{ileDzienniePociagów} * 345.67$

c) dodaj prywatne pole nazwaStacji typu string

d) przestoń metodę ToString() tak, aby zwracała informacje o stacji np.

Stacja: Malbork, liczba pociągów dziennie: 24.

8. W klasie Program i metodzie Main wykonaj czynności:

a) stwórz kolejkę generyczną miasta przechowującą obiekty typu Stacja

b) na kolejkę dodaj 5 elementów oraz upewnij się, że każdy element ma ustawione pole nazwaStacji i ileDzienniePociagow

c) dla elementów z kolejki wywołaj kolejno metodę ObliczKoszt(), zwróć wartość wypisz na konsoli, po zwróceniu pozostaw elementy na kolejce

9. Do klasy Stacja podepnij interfejs IComparable<Stacja> tak, aby sortował elementy po nazwie alfabetycznie.

10. W klasie Program i metodzie Main stwórz listę na elementy typu Stacja, na listę skopiuj elementy z kolejki miasta. Następnie posortuj listę. Po sortowaniu wypisz elementy na konsoli używając ToString() + ew. inną metodę.

Punktacja

Polecenia 1-4 – łącznie 4 punktów

Polecenia 5-8 – łącznie 4 punkty

Polecenia 9-10 – łącznie 2 punkt

Aby otrzymać ocenę dostateczną na koniec, trzeba uzyskać min. 6 punktów.