

Zadanie 1.

Zdefiniuj klasę `TPunkt` przedstawiającą punkt na płaszczyźnie. W klasie zdefiniuj metodę `Odleglosc`, która zwraca odległość od punktu będącego parametrem tej metody. Klasa ta nie posiada konstruktora domyślnego. Pozostałe metody tej klasy określ samodzielnie.

Zdefiniuj klasę `TKolo`, dziedziczącą po klasie `TPunkt`. Pola odziedziczone reprezentują środek koła, punkt na okręgu jest dodatkowym polem tej klasy. Wśród metod tej klasy umieść metody wyznaczające pole i obwód koła.

Do konstruktorów każdej klasy dołącz instrukcje pozwalające śledzić kolejność ich wywołania. Konstruktory klas powinny umożliwiać następujące deklaracje:

```
TPunkt p1(1,1), p2(0,3);
```

```
TKolo k1(0,0,2,2), k2(p1,1,1), k3(p1,p2);
```

Jeśli parametrem (parametrami) konstruktora jest punkt, to należy wykorzystać konstruktor kopiujący.

Zadanie 2.

Napisz klasę `TSamochod` posiadającą następujące pola chronione:

`marka`

`pB` (pojemność baku)

`pM` (prędkość maksymalna)

`zP` (zużycie paliwa na 100 km)

Posiada ona następujące funkcje publiczne:

- konstruktor ustawiający wszystkie pola na podstawie swoich parametrów,
- `jedz(float jakSzybko, float jakDaleko)` – wyświetla komunikat mówiący jak szybko samochód pojedzie (nie szybciej niż prędkość maksymalna), ile litrów paliwa zostanie zużytych na trasie, ile razy po drodze będzie musiał tankować i ile czasu zajmie podróż (każde tankowanie zajmuje 10 minut). Na początku podróży bak samochodu jest pełny. Jeśli samochód jedzie za wolno (poniżej 30% prędkości maksymalnej) lub za szybko (powyżej 80% prędkości maksymalnej), to zużycie paliwa wzrasta o 20%.
- w rozwiązaniu można wykorzystać dodatkowe metody (np. dla wyznaczenia chwilowego zużycia paliwa).

Następnie zdefiniuj klasę `TKabriolet` dziedziczącą po klasie `TSamochod`. Dodatkowo posiada ona pole logiczne `dach_otwarty` ustawiane w konstruktorze na `false` oraz funkcje:

- `otworz_dach`,
- `zamknij_dach`.

W zmodyfikowanej funkcji `jedz` wyświetlamy informację o tym, czy dach kabrioletu jest otwarty. Dodatkowo należy wziąć pod uwagę, że z otwartym dachem kabriolet pali o 15% więcej.

Zadanie 3.

Zdefiniuj klasę `TaxiM` (taksówka miejska) z polami chronionymi:

`start` (opłata za rozpoczęcie jazdy taksówką w groszach),

`km` (odległość wyrażona w kilometrach – liczba całkowita),

`stawka` (cena jednego kilometra w groszach)

oraz konstruktorem (ustawia pola na podstawie wartości swoich parametrów) i metodami `wczytaj` i `wyswietl` (inne metody wg potrzeb).

Metoda `wyswietl` wykorzystuje zewnętrzną metodę o nagłówku

`string formatuj(int pom)`, która zwraca napis będący liczbą całkowitą przekazaną przez parametr zapisaną w złotych i groszach.

Jeśli na przykład pole `start = 500`, `km=10` a `stawka=40` oznacza to, że za rozpoczęcie jazdy taksówką płacimy 5 złotych, jedziemy 10 kilometrów, a za przejazd płacimy $500 + 40 * 10 = 900$ gr = 9 pln.

Zdefiniuj klasę pochodną `Taxi` – część trasy pokonujemy poza miastem. W klasie tej zdefiniuj dodatkowe pola:

`km_strefa` (odległość poza miastem wyrażona w kilometrach),

`stawka_strefa` (cena za jeden kilometr poza miastem wyrażona w groszach).

W metodzie `wyswietl` uwzględnij jazdę samochodem poza miastem.

Inne metody należy zdefiniować samodzielnie.

Napisz fragment programu, w którym zdefiniujemy obiekt klasy `Taxi`, wczytamy wartości jego pól i wyświetlimy rachunek postaci:

Rachunek za przejazd taksówką:

Do zapłaty.....ZZ.ZZ pln

W tym:

`start`.....ZZ.ZZ pln

`miasto`.....XX km * YY gr/km = ZZ.ZZ pln

`poza miastem`.....XX km * YY gr/km = ZZ.ZZ pln