Zadania podstawowe:

1. Stwórz klasę Car.
2. Stwórz w niej nowe prywatne pola **double pojemonoscSilnika**, **string marka**
3. Stwórz obiekt typu Car, czy masz dostęp do jego pól? czy możesz zmienić wartości? (nie zmieniaj pól klasy na publiczne).
4. Do klasy Car dodaj ręcznie napisany konstruktor.
5. Przeciąż konstruktor tworząc nowy: **Car(double pojemonoscSilnika**, **string marka).** Niech ten parametryczny konstruktor ustawia wartości pól pojemność silnika oraz marka. (użyj inicjalizatora **this**).
6. Stwórz nowy obiekt za pomocą konstruktora parametrycznego.
7. Za pomocą debuggera sprawdź jakie warości na prywatnych polach ma obiekt numer 1, a jakie obiekt 2.
8. Zmień modyfikatory dostępu konstruktorów na private. Czy jest możliwe utworzenie nowego obiektu tej klasy? Co się stało z konstruktorem domyślnym?
9. W klasie Car stwórz metodę **Car Create(),** która będzie zwracać obiekt.
10. Stwórz obiekt dzięki metodzie Create() (pamiętałeś o dodaniu static do metody? )
11. Utwórz statyczne pole **int iloscKol**. Następnie stwórz statyczny konstruktor ustawiający wartość tego pola na 4.
12. Uruchom program i w debuggerze sprawdź jaką wartość pola iloscKol ma wczesniej stworzony obiekt.
13. Do klasy Car dodaj destruktor **~Car(),** wewnątrz dodaj następujący kod:
**Console.WriteLine(„Zwalniam pamięć”);
Console.ReadKey();**
14. Uruchom program i zwróć uwagę kiedy następuje zwolnienie pamięci.

Załóż konto na Github, stwórz repozytorium i dodaj dowolny projekt. Link wyślij na maila piojas@matman.uwm.edu.pl (jeśli robisz to na współdzielonym komputerze wyczyść lub zmień wprowadzony w ustawieniach adres e-mail – użyj takiego, który może być publicznie dostępny; ew. upewnij się i wyczyść poświadczenia w systemie windows).

Dodatkowe:

1. Stwórz klasę Konto z kilkoma polami, metodami i konstruktorami (saldo początkowe, końcowe, przelew miedzy kontami, przelew zewnętrzny, wplata, wyplata itd.) Następnie utwórz co najmniej 3 obiekty i wykonaj kilka metody na nich.
2. Stwórz klasę Telefon, która będzie zawierała parametry do obsługi bilingu (usługa prepaid).Dodatkowo ma zawierać metody, konstruktory (saldo początkowe, końcowe, doładowanie, wysłanie smsa, rozmowa, połączenie z internetem, usługi premium). Potem stwórz kilka obiektów i wykonaj na nich metody. Na ekranie wyświetl „bilingi” dla stworzonych obiektów.

Samodzielna lektura do opanowania w celu zarządzania plikami (jeśli zabraknie czasu na ćwiczeniach, to w domu)

<http://www.cs.put.poznan.pl/anstroinski/data/uploads/mipo/materials/mipo-struktury-i-dzialania-na-plikach-w-c.pdf>

To co trzeba poznać to: Czytanie i zapisywanie do plików tekstowych, Zarządzanie systemem plików, Pobieranie informacji o plikach i katalogach

(pominąć część o strukturach, Wykorzystanie protokołu TCP/IP)