

# Platforma .NET

## Zajęcia laboratoryjne

### Ćwiczenie nr 2

1. (a) Napisać klasę Zegar, która zgłasza zdarzenie co 1 sekundę.  
Wskazówka: W klasie zgłaszającej zdarzenie zdefiniować delegat `delZgłoszenieZegara` i utworzyć obiekt delegata zdarzenie. Funkcja delegata nie zwraca niczego, przyjmuje jako parametry referencję na klasę zgłaszającą (Zegar) i referencję na obiekt klasy `TimeEventArgs`.  
W klasie Zegar zaimplementować funkcję `Run`, która po uruchomieniu wykonuje się w pętli nieskończonej, odczytuje czas systemowy (`DateTime.Now`) i co sekundę zgłasza zdarzenie, tzn. uruchamia obiekt delegata (zdarzenie). Należy sprawdzić, czy obiekt delegata nie jest null!
- (b) Napisać klasę `TimeEventArgs` dziedziczącą po klasie `EventArgs`, przechowującą informację o czasie (godzinę, minutę i sekundę).
- (c) Napisać klasę `Sluchacz`, która nasłuchuje na zdarzenie generowane przez Zegar. Po wystąpieniu zdarzenia słuchacz wypisuje swoją nazwę i czas na konsoli.  
Wskazówka: Napisać funkcję `subskrybuj`, która przyjmuje jako parametr referencję na obiekt Zegara; funkcja przypisuje delegatowi klasy zgłaszającej zdarzenie metodę obsługi zdarzenia.
- (d) W klasie `Sluchacz` napisać funkcję obsługi zdarzenia, która przyjmuje jako parametry referencję na obiekt zgłaszającą zdarzenie (Zegar) i referencję na obiekt klasy `TimeEventArgs`.
- (e) W funkcji głównej uruchomić zegar i kilka słuchaczy
2. (a) Napisać klasę `Asynchroniczna`, której zadaniem jest uruchomienie kilku długotrwałych funkcji w sposób asynchroniczny.  
Wskazówka: W klasie tej zdefiniować obiekt delegata (zdefiniowanego poza klasą) jako zdarzenie (event). Jest to delegat na funkcję zwracającą `int`. Napisać także publiczną funkcję `dodajDelegata`, która dodaje delegata do obiektu delegata klasy.
- (b) W klasie `Asynchroniczna` napisać funkcję uruchamiającą delegaty. Funkcja uruchamia wszystkie funkcje przypisane do delegata (`foreach`, `GetInvocationList()`) z wykorzystaniem mechanizmu `BeginInvoke`. Mechanizm ten wymaga podania delegata funkcji, która jest uruchamiana po zakończeniu pracy funkcji wywołanej. Dalej: napisać funkcję obsługi zakończenia pracy delegata, jako parametr funkcja przyjmuje obiekt klasy `IAsyncResult`. W obiekcie tym zawarty jest uruchomiony delegat, to co zwrócił delegat można pobrać wywołując dla tegoż delegata metodę `EndInvoke`.
- (c) W funkcji głównej zaprezentować działanie.