



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie obiektowe [S2Eltech2>PO2]

### Przedmiot

Kierunek studiów  
Elektrotechnika

Rok/Semestr  
1/2

Studia w zakresie (specjalność)  
Sieci i automatyka elektroenergetyczna

Profil studiów  
ogólnoakademicki

Poziom studiów  
drugiego stopnia

Język oferowanego przedmiotu  
polski

Forma studiów  
stacjonarne

Wymagalność  
obligatoryjny

### Liczba godzin

| Wykład    | Laboratorium       | Inne |
|-----------|--------------------|------|
| 0         | 30                 | 0    |
| Ćwiczenia | Projekty/seminaria |      |
| 0         | 0                  |      |

### Liczba punktów ECTS

2,00

### Koordynatorzy

dr hab. inż. Leszek Kasprzyk prof. PP  
leszek.kasprzyk@put.poznan.pl

### Wykładowcy

### Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu informatyki i programowania. Dodatkowo powinien również posiadać umiejętności myślenia abstrakcyjnego, pozyskiwania informacji ze wskazanych źródeł oraz wykazywać gotowość do podjęcia współpracy w zespole.

### Cel przedmiotu

Przekazanie studentom teoretycznych i praktycznych zagadnień programowania wysokopoziomowego z elementami programowania obiektowego, nabycie umiejętności tworzenia aplikacji w środowisku Microsoft Visual Studio (w języku C#). Rozwijanie u studentów umiejętności rozwiązywania problemów występujących podczas projektowania i implementacji aplikacji technicznych.

### Przedmiotowe efekty uczenia się

Wiedza:

Student zna zasady programowania wysokopoziomowego. Posiada wiedzę z zakresu programowania obiektowego przydatną podczas tworzenia aplikacji technicznych.

## Umiejętności:

Student umie stosować narzędzia służące do programowania z wykorzystaniem elementów programowania obiektowego. Potrafi programować w języku C#.

## Kompetencje społeczne:

Świadomość szybkości rozwoju technologii informatycznych i związanej z tym konieczności aktualizacji wiedzy. Potrafi samodzielnie wyszukiwać informacje w literaturze i Internecie, także w językach obcych.

## Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Umiejętności nabyte w ramach zajęć laboratoryjnych weryfikowane są na podstawie kolokwium zaliczeniowego lub wykonanej przez siebie aplikacji projektowej z uwzględnieniem aktywności indywidualnej na zajęciach. Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć (szczególnie za: omówienie dodatkowych aspektów zagadnienia; efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu; umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium; uwagi związane z udoskonaleniem materiałów dydaktycznych; staranność)

## Treści programowe

Implementacja w środowisko Visual Studio C# Express Edition przedstawionych na wykładzie zagadnień w praktycznych programach obiektowego wykorzystujących następujące elementy programowania: deklaracje typów obiektowych, pola i metody, pola typu readonly, elementy statyczne, konstruktory i destruktor, właściwości, przeciążenie metod, przeciążenia operatorów, hermetyzacja, dziedziczenie, polimorfizm i jego zastosowanie, klasy i metody abstrakcyjne, wykorzystanie kolekcji różnych typów, wykorzystanie delegat oraz interfejsów, programowanie aplikacji wykorzystującej wiele wątków, implementacja wybranych wzorców projektowych.

## Tematyka zajęć

1. Zajęcia organizacyjne, omówienie środowiska.
2. Podstawowe typy wartościowe, deklaracje funkcji, pętli, tworzenie wykresów.
3. Typ wyliczeniowy, konwersja jawna i niejawną, params.
- 4-6. Tworzenie klasy i obiektu, właściwości, metody, pola, hermetyzacja.
7. Konstruktory, destruktor, operator, elementy statyczne.
- 8-9. Dziedziczenie i polimorfizm, protected.
10. Kolekcje i Tuple.
- 11-12. Delegaty, interfejsy, elementy uzupełniające materiał.
13. Praca wielowątkowa.
14. Wzorzec projektowy.
15. Kolokwium zaliczeniowe.

## Metody dydaktyczne

Praca indywidualna w laboratorium komputerowym, polegająca na implementacji zadanych funkcji i programów komputerowych, dyskusja, demonstracje.

## Literatura

Podstawowa:

1. Troelsen, Andrew; Japikse, Phiplip. Język C# 6.0 i platforma .NET 4.6. Red. . Warszawa: Wydawnictwo Naukowe PWN, 2017, 1465 s. ISBN 978-83-01-19832-9
2. Chłosta, Paweł. Aplikacje Windows Forms .Net w C#. Red. . Warszawa: Wydawnictwo Naukowe PWN, 2006, 318 s. ISBN 83-01-14830-6
3. Rodenburg, Jort. Koduj jak profesjonalista C#. Red. . Warszawa: Promise, 2022, 466 s. ISBN 978-83-7541-459-2
4. Trey Nash, Accelerated C# 2010, Apress, 2010
5. C# language specification: <https://github.com/dotnet/csharpplang/tree/main/spec>
6. Internet

Uzupełniająca:

1. Sharp, John. Microsoft Visual C# 2017 Krok po kroku. Red. . Warszawa: Promise, 2019, 937 s. ISBN 978-83-7541-362-5

2. Stasiak, Andrzej; Dąbrowski, Włodzimierz; Wolski, Michał. Modelowanie systemów informatycznych w języku UML 2.1. Red. . Warszawa: Wydawnictwo Naukowe PWN, 2009, 196 s. ISBN 978-83-01-21108-0

### Bilans nakładu pracy przeciętnego studenta

|  | Godzin | ECTS |
|--|--------|------|
| Łączny nakład pracy  | 50     | 2,00 |
| Zajęcia wymagające bezpośredniego kontaktu z nauczycielem  | 30     | 1,50 |
| Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu) | 20     | 0,50 |