

# Object-oriented Programming 2 - course description

| General information |                                                                           |
|---------------------|---------------------------------------------------------------------------|
| Course name         | Object-oriented Programming 2                                             |
| Course ID           | 11.3-WK-II-EP-PO2-W-S14_pNadGenS36RL                                      |
| Faculty             | <a href="#">Faculty of Mathematics, Computer Science and Econometrics</a> |
| Field of study      | Informatics and Econometrics                                              |
| Education profile   | academic                                                                  |
| Level of studies    | First-cycle studies leading to Bachelor's degree                          |
| Beginning semester  | winter term 2018/2019                                                     |

| Course information  |                                                                                          |
|---------------------|------------------------------------------------------------------------------------------|
| Semester            | 4                                                                                        |
| ECTS credits to win | 6                                                                                        |
| Course type         | optional                                                                                 |
| Teaching language   | polish                                                                                   |
| Author of syllabus  | <ul style="list-style-type: none"><li>dr hab. inż. Andrzej Kasperski, prof. UZ</li></ul> |

| Classes forms  |                                |                            |                                |                            |                    |
|----------------|--------------------------------|----------------------------|--------------------------------|----------------------------|--------------------|
| The class form | Hours per semester (full-time) | Hours per week (full-time) | Hours per semester (part-time) | Hours per week (part-time) | Form of assignment |
| Lecture        | 30                             | 2                          | -                              | -                          | Exam               |
| Laboratory     | 30                             | 2                          | -                              | -                          | Credit with grade  |

## Aim of the course

Poznanie teoretyczne i praktyczne nowoczesnych języków programowania obiektowego C# i Python. Zapoznanie ze środowiskiem programistycznym Microsoft Visual Studio i poznanie jego najważniejszych funkcjonalności. Przedstawienie wybranych zastosowań obiektowych języków programowania.

## Prerequisites

Podstawowa znajomość programowania.

## Scope

### Wykład

1. Wprowadzenie do platformy programistycznej .Net Framework.
2. Podstawy programowania w obiektowym języku programowania C#.
3. Deklaracja klasy, tworzenie obiektów, przekazywanie parametrów do metod, dziedziczenie w C#.
4. Zdarzenia, delegaty, obsługa wyjątków w C#.
5. Własności, polimorfizm, kolekcje i listy obiektów w C#.
6. Klasy abstrakcyjne, statyczne, anonimowe, zamknięte i interfejsy w C#.
7. Przeciążanie operatorów w C#.
8. Operacje plikowe oraz odczytanie konfiguracji/zasobów komputera w C#.
9. Operacje graficzne w C#.
10. Wprowadzenie do obiektowego języka programowania Python, wirtualna maszyna Pythona, interpreter Pythona,
11. Podstawy programowania w Pythonie.
12. Deklaracja klasy, tworzenie obiektów, dziedziczenie (w tym dziedziczenie wielobazowe) w Pythonie.
13. Standardowe biblioteki Pythona oraz wprowadzenie do biblioteki BioPython.
14. Obiektowa analiza kodu genetycznego.
15. Dostęp do zasobów bioinformatycznych z BioPythona.

### Laboratorium

1. Wprowadzenie do środowiska Microsoft Visual Studio.
2. Pisanie i uruchamianie prostych programów w C#.
3. Deklaracja klas, tworzenie obiektów w C# oraz wywołania metod na rzecz utworzonych obiektów.
4. Obsługa zdarzeń, wyjątków oraz delegaty w C#.
5. Implementacja własności, testowanie polimorfizmu, tworzenie list obiektów w C#.
6. Definiowanie różnych typów klas, w tym klas abstrakcyjnych oraz interfejsów w C#.
7. Testowanie przeciążania operatorów w C#.
8. Implementacja podstawowych operacji plikowych w C#.
9. Odczytywanie konfiguracji/zasobów komputera w C#.
10. Implementacja wybranych operacji graficznych w C#.
11. Pisanie oraz uruchamianie prostych programów w Pythonie.
12. Deklaracja klas, tworzenie obiektów w Pythonie oraz wywołania metod na rzecz utworzonych obiektów.
13. Testy dziedziczenia oraz dziedziczenia wielobazowego.

## Teaching methods

**Wykład:** wykład konwencjonalny.

**Laboratorium:** samodzielna praca przy komputerze. Przerabiany materiał według opracowanych instrukcji, które każdy student dostaje na początku zajęć. Dyskusje prowadzące do pogłębienia wiedzy i lepszego zrozumienia przerabianego materiału.

## Learning outcomes and methods of theirs verification

| Outcome description                                                                     | Outcome symbols                                                                                        | Methods of verification                                                                                                                                                             | The class form                                               |
|-----------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| Student zna podstawy programowania obiektowego w językach C# i Python.                  | <ul style="list-style-type: none"> <li><a href="#">K_W10</a></li> <li><a href="#">K_W11</a></li> </ul> | <ul style="list-style-type: none"> <li>an exam - oral, descriptive, test and other</li> </ul>                                                                                       | <ul style="list-style-type: none"> <li>Lecture</li> </ul>    |
| Student potrafi napisać przykładowy kod programu zawierający dziedziczenie wielobazowe. | <ul style="list-style-type: none"> <li><a href="#">K_U25</a></li> <li><a href="#">K_U26</a></li> </ul> | <ul style="list-style-type: none"> <li>a discussion</li> <li>an evaluation test</li> <li>sprawdzanie stopnia przygotowania studentów oraz ich aktywności w trakcie zajęć</li> </ul> | <ul style="list-style-type: none"> <li>Laboratory</li> </ul> |
| Student zna podstawowe informacje dotyczące platformy programistycznej .NET Framework.  | <ul style="list-style-type: none"> <li><a href="#">K_W10</a></li> <li><a href="#">K_W11</a></li> </ul> | <ul style="list-style-type: none"> <li>an exam - oral, descriptive, test and other</li> </ul>                                                                                       | <ul style="list-style-type: none"> <li>Lecture</li> </ul>    |
| Student zna pojęcie delegaty oraz przeciążania operatorów                               | <ul style="list-style-type: none"> <li><a href="#">K_W10</a></li> <li><a href="#">K_W11</a></li> </ul> | <ul style="list-style-type: none"> <li>an exam - oral, descriptive, test and other</li> </ul>                                                                                       | <ul style="list-style-type: none"> <li>Lecture</li> </ul>    |

## Assignment conditions

Na ocenę z przedmiotu składa się ocena z laboratorium (40%) i ocena z egzaminu (60%). Warunkiem przystąpienia do egzaminu jest pozytywna ocena z laboratorium. Warunkiem zaliczenia przedmiotu jest pozytywna ocena z laboratorium i z egzaminu.

## Recommended reading

1. Beazley D.M., Programowanie: Pyton, RM, 2002.
2. Lis M., C# praktyczny kurs, Helion, 2007.
3. Lutz M., Ascher D., Python. Wprowadzenie, Helion, 2002.
4. Perry S.C., C# i .NET, Helion, 2006.

## Further reading

1. Allen J., C# księga przykładów, APN Promise, 2006.
2. Fehily C., Po prostu Python, Chris Fehily, Helion, 2002.

## Notes

Modified by dr Robert Dylewski, prof. UZ (last modification: 04-05-2018 19:24)