Scripting languages - opis przedmiotu

Informacje ogólne

Nazwa przedmiotu	Scripting languages
Kod przedmiotu	11.3-WE-INFP-ScripLang
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	Program Erasmus pierwszego stopnia
Semestr rozpoczęcia	semestr zimowy 2020/2021

Informacje o przedmiocie

Semestr	4
Liczba punktów ECTS do zdobycia	6
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	• dr inż. Piotr Witczak

Formy zajęć

Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	30	2	-	-	Egzamin
Laboratorium	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

- Zapoznanie studentów z językami skryptowymi (w szczególności z językiem Python)
- Ukształtowanie umiejętności praktycznego zastosowania języków skryptowych
- Ukształtowanie umiejętności wykorzystywania gotowych bibliotek i frameworków

Wymagania wstępne

- Podstawy programowania
- Programowanie obiektowe

Zakres tematyczny

- paradygmat języków skryptowych i interpretowanych
- różnice pomiędzy językami interpretowanymi a kompilowanymi (na przykładzie języków Python i C++)
- zastosowanie języków skryptowych do tworzenia stron www
- wykorzystanie języków skryptowych w tworzeniu nowoczesnego oprogramowania
- wykorzystanie języków skryptowych do obliczeń inteligentnych

Metody kształcenia

- wykład: wykład konwencjonalny
- laboratorium: ćwiczenia lab., praca w grupach, programowanie w parach,

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Potrafi zastosować techniki skryptowe do realizacji zadania będącego częścią większego projektu lub systemu informatycznego		• obserwacje i ocena umiejętności praktycznych studenta	• Laboratorium
Ma wiedzę na temat trendów rozwojowych dyscypliny oraz potrzeby usprawniania codziennych czynności informatycznych poprzez stosowanie języków skryptowych		• egzamin - ustny, opisowy, testowy i inne	• Wykład
Potrafi zastosować języki skryptowe do rozwiązania problemu naukowego oraz inżynierskiego		• obserwacje i ocena umiejętności praktycznych studenta	• Laboratorium
Rozumie paradygmat języków skryptowych oraz interpretowanych oraz różnice w stosunku do języków kompilowanych		• egzamin - ustny, opisowy, testowy i inne	• Wykład
Rozumie potrzebę stosowania języków skryptowych		• dyskusja	• Wykład

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Dokonuje wyboru właściwych narzędzi oraz rozwiązań skryptowych w zależności od wymagań projektowyc		• obserwacje i ocena umiejętności praktycznych studenta	• Laboratorium

Warunki zaliczenia

Wykład - warunkiem zaliczenia jest uzyskanie pozytywnej oceny z egzaminu

Laboratorium - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich ćwiczeń laboratoryjnych, przewidzianych do realizacji w ramach programu laboratorium

Składowe oceny końcowej = wykład: 50% + laboratorium: 50%

Literatura podstawowa

1. Learning Python: Powerful Object-Oriented Programming, Mark Lutz, "O'Reilly Media, Inc.", 2013, 9781449355692
2. Python Crash Course: A Hands-On, Project-Based Introduction to Programming, Eric Matthes, No Starch Press, 2015, ISBN: 9781593276034
3. Deep Learning with TensorFlow: Explore neural networks and build intelligent systems with Python, 2nd Edition, Giancarlo Zaccone, Md. Rezaul Karim, Packt Publishing Ltd, 2018, ISBN: 9781788831833
4. Beginning Django: Web Application Development and Deployment with Python, Daniel Rubio, Apress, 2017 ISBN: 978148422787
5. Practical Python Design Patterns: Pythonic Solutions to Common Problems, Wessel Badenhorst, Apress, 2017, ISBN: 9781484226803

Literatura uzupełniająca

1. Python Cookbook, Alex Martelli, Anna Ravenscroft, David Ascher, "O'Reilly Media, Inc.", 2005, ISBN: 9780596554743
2. Practical Django 2 and Channels 2: Building Projects and Applications with Real-Time Capabilities, Federico Marani, Apress, 2018, ISBN: 9781484240991
3. Pro Python 3: Features and Tools for Professional Development, J. Burton Browning, Marty Alchin, Apress, 2019, ISBN: 9781484243855
4. Fluent Python: Clear, Concise, and Effective Programming, Luciano Ramalho, "O'Reilly Media, Inc.", 2015, ISBN: 9781491946251
5. Python Continuous Integration and Delivery: A Concise Guide with Examples, Moritz Lenz, Apress, 2018,9781484242810
6. Clean Architecture: A Craftsman's Guide to Software Structure and Design, *Martin, Robert C,* Prentice Hall, 2018, 9780134494166
7. Clean Python: Elegant Coding in Python, Sunil Kapil, Apress, 2019, ISBN: 9781484248782
8. Effective Python: 59 Specific Ways to Write Better Python, Brett Slatkin, Addison-Wesley Professional, 2015, ISBN: 9780134034409
9. Scripting Languages: Automating the Web, Rohit Khare, O'Reilly, 1997, ISBN: 9781565922655
10. Software Architecture: A Comprehensive Framework and Guide for Practitioners, Oliver Vogel, Ingo Arnold, Arif Chughtai, Timo Kehrer, Springer Science & Business Media, 2011, ISBN: 9783642197369

Uwagi

Zmodyfikowane przez dr inż. Piotr Witczak (ostatnia modyfikacja: 27-04-2020 11:53)

Wygenerowano automatycznie z systemu SylabUZ