Opis **zajęć (sylabus)**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Nazwa zajęć: | | Advanced Programming | | | | | | | **ECTS** | **2** |
| Nazwa zajęć w j. polskim: | | Programowanie zaawansowane | | | | | | | | |
| Zajęcia dla kierunku studiów: | | **Informatyka i ekonometria** | | | | | | | | |
|  | |  | | | | | | | | |
| Język wykładowy: | | angielski | | | | Poziom studiów: | | studia II stopnia | | |
| Forma studiów: | stacjonarne  niestacjonarne | Status zajęć: | podstawowe  kierunkowe | obowiązkowe  do wyboru | | Numer semestru: ……**1**….. | |  semestr zimowy   semestr letni | | |
|  |  | Rok akademicki, od którego obowiązuje opis (rocznik): | | | | 2019/2020 | Numer katalogowy: | **ZIM-IE-BDA-2S-01Z-8** | | |
|  | | | | | | | | | | |
| Koordynator zajęć: | |  | | | | | | | | |
| Prowadzący zajęcia: | |  | | | | | | | | |
| Jednostka realizująca: | |  | | | | | | | | |
| Jednostka zlecająca: | |  | | | | | | | | |
| Założenia, cele i opis zajęć: | | The purpose of the course is knowing the ways of writing programs using advanced programistic techniques and procedural and object – oriented paradigms with the usage of chosen programming language and two ways of data presentation: text and graphical.Plan of the course:1. Introduction to programming: The structure of the program, Standard input and output, The conditional instruction IF – ELSE, Basic operators and mathematical functions.2. Some more practical aspects of basic programming in chosen language: Ternary operator, Random values, Formatted output. 3. WHILE and DO-WHILE iteration loops: WHILE loop, DO – WHILE loop.  4. FOR iteration loop and next useful instructions: FOR loop, BREAK and CONTINUE instructions, SWITCH instruction.  5. Arrays: The structure of one–dimensional array, FOR EACH loop, Advantages and disadvantages of arrays, Multi-dimensional arrays.  6. Lists: The structure of a list, Most important operations, Multi-dimensional lists.  7. Strings: String as an object, Function charAt(), Exceptions, TRY-CATCH instruction.  8. Connection between characters and integers. ASCII table: Connection between characters and integers, Most useful operations.  9. Functions: The types of the functions, Recurrent functions.  10. Programs with positional arguments.  11. Basic file operations: File object., Writing data to a text file, Reading data from a text file.  12. Basic file operations part II: Adding new data to a text file.  13. Introduction to object-oriented programming: Object, field, method, class, The usage of parametric constructors, Keyword THIS.  14. Introduction to object-oriented programming part II: Extension of a class, The basic concepts connected with  inheritance, Keyword SUPER.  15. Programs with graphical user interface (GUI): Philosophy of graphical user interface, Basic objects and first window, Events service, JAR archives. | | | | | | | | |
| Formy dydaktyczne, liczba godzin: | | ćwiczenia laboratoryjne; liczba godzin ...30… | | | | | | | | |
| Metody dydaktyczne: | | dyskusja problemu, rozwiązywanie problemu, studium przypadku, projekt grupowy, konsultacje | | | | | | | | |
| Wymagania formalne  i założenia wstępne: | | Wymagana jest elementarna wiedza dotycząca podstaw matematyki oraz programowania | | | | | | | | |
| Efekty uczenia się: | | Wiedza:  1 – zna zasady tworzenia i cykl życia programów i aplikacji komputerowych  2 – zna i rozumie klasyfikację współczesnych technik, narzędzi i języków programowania  3 – zna podstawowe struktury danych wykorzystywane w programowaniu strukturalnym i obiektowym | | | Umiejętności:  1 – potrafi pisać programy komputerowe z wykorzystaniem różnych narzędzi i zaawansowanych technik programistycznych  2 - umie rozwiązywać zadania i problemy przy użyciu samodzielnie i zespołowo tworzonych programów i aplikacji komputerowych | | | | Kompetencje: | |
| Sposób weryfikacji efektów uczenia się: | | test wiedzy dotyczący znajomości paradygmatów programowania proceduralnego i obiektowego w wybranym języku programowania (efekty wiedzowe)  kolokwium praktyczne polegające na samodzielnym napisaniu programów komputerowych rozwiązujących przedstawione problemy (efekty umiejętności 1)  projekt grupowy polegający na stworzeniu aplikacji z wykorzystaniem interfejsu GUI oraz poznanych technik obiektowych , która rozwiązuje problem przedstawiony w temacie i opisie aplikacji (efekt umiejętności 2) | | | | | | | | |
| Forma dokumentacji osiągniętych efektów uczenia się: | | Ocenione testy wiedzy.  Zarchiwizowane kolokwium praktyczne w formie elektronicznej z ocenami.  Ocenione pliki projektu grupowego w formie elektronicznej.  Ocenione testy wiedzy. | | | | | | | | |
| Elementy i wagi mające wpływ  na ocenę końcową: | | **test wiedzy 25%**  **kolokwium praktyczne 50%**  **projekt grupowy 25%** | | | | | | | | |
| Miejsce realizacji zajęć: | | laboratorium komputerowe | | | | | | | | |
| Literatura podstawowa i uzupełniająca:  Literatura podstawowa:  M. Ziółkowski.: Advanced JAVA programming – lectures and presentations for BDA students, SGGW, Warsaw, 2018.  J. Bloch: Java. Efektywne programowanie, Helion, 2018.  C.S. Horstmann: Java. Podstawy, Helion, 2018.  Literatura uzupełniająca: | | | | | | | | | | |
| UWAGI  Minimalna liczba punktów konieczna do zaliczenia: 50% z testu wiedzy, 50% z kolokwium praktycznego oraz 50% z projektu grupowego | | | | | | | | | | |

Wskaźniki ilościowe charakteryzujące moduł/przedmiot:

|  |  |
| --- | --- |
| Szacunkowa sumaryczna liczba godzin pracy studenta (kontaktowych i pracy własnej) niezbędna dla osiągnięcia zakładanych dla zajęć efektów uczenia się - na tej podstawie należy wypełnić pole ECTS: | **60 h** |
| Łączna liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich lub innych osób prowadzących zajęcia: | **1 ECTS** |

Tabela zgodności kierunkowych efektów uczenia się z efektami przedmiotu:

|  |  |  |  |
| --- | --- | --- | --- |
| kategoria efektu | Efekty uczenia się dla zajęć: | Odniesienie do efektów dla programu studiów dla kierunku | Oddziaływanie zajęć na efekt kierunkowy\*) |
| Wiedza 1 | zna zasady tworzenia i cykl życia programów i aplikacji komputerowych | K\_W06 / P7S\_WG | 3 |
| Wiedza 2 | zna i rozumie klasyfikację współczesnych technik, narzędzi i języków programowania | K\_W08 / P7S\_WG | 3 |
| Wiedza 3 | zna podstawowe struktury danych wykorzystywane w programowaniu strukturalnym i obiektowym | K\_W12 / P7S\_WG | 2 |
| Umiejętności 1 | potrafi pisać programy komputerowe z wykorzystaniem różnych narzędzi i zaawansowanych technik programistycznych | K\_U13 / P7S\_UW | 1 |
| Umiejętności 2 | umie rozwiązywać zadania i problemy przy użyciu samodzielnie i zespołowo tworzonych programów i aplikacji komputerowych | K\_U13 / P7S\_UW | 1  1 |

\*)

3 – zaawansowany i szczegółowy,

2 – znaczący,

1 – podstawowy,