**Opis modułu kształcenia / przedmiotu (sylabus)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Rok akademicki: | 2017/2018 | Grupa przedmiotów: |  | Numer katalogowy: | |  | |
|  | | | | | | | |
| Nazwa przedmiotu1): | | Programowanie obiektowe | | | | **ECTS** 2) | **6** |
| Tłumaczenie nazwy na jęz. angielski3): | | Object Oriented Programming | | | | | |
| Kierunek studiów4): | | **Informatyka** | | | | | |
| Koordynator przedmiotu5): | | **dr inż. Piotr Wrzeciono** | | | | | |
| Prowadzący zajęcia6): | | **dr inż. Piotr Wrzeciono, dr inż. Marcin Bator** | | | | | |
| Jednostka realizująca7): | | **Wydział Zastosowań Informatyki i Matematyki, Katedra Informatyki, Zakład Podstaw Informatyki** | | | | | |
| Wydział, dla którego przedmiot jest realizowany8): | |  | | | | | |
| Status przedmiotu9): | | a) przedmiot **kierunkowy** | b) stopień **1** rok **1** | | c) ~~stacjonarne~~ / niestacjonarne | | |
| Cykl dydaktyczny10): | | **Semestr letni** | Jęz. wykładowy11): **polski** | |  | | |
| Założenia i cele przedmiotu12): | | Przedstawienie studentom zasad programowania i modelowania obiektowego, zaprezentowanie podstaw programowania wielowątkowego (Java). W ramach nauki modelowania studenci mają również poznać diagram klas UML. | | | | | |
| Formy dydaktyczne, liczba godzin13): | | 1. wykład; liczba godzin 18; 2. laboratorium; liczba godzin 18; 3. projekt wykonywany w domu; liczba godzin 15; 4. konsultacje wykładowe 5. | | | | | |
| Metody dydaktyczne14): | | Prezentacja omawiająca konkretne zagadnienia (dostępna dla studentów przed zajęciami), wykład z prezentacją dużego przykładu (pisanego w całości podczas zajęć), laboratorium oraz zadania domowe. Podczas pisania programu na wykładzie, studenci aktywnie uczestniczą w zajęciach poprzez zadawanie pytań i proponowanie rozwiązań. | | | | | |
| Pełny opis przedmiotu15): | | Zajęcia obejmują następujące zagadnienia:  * Podstawy modelowania obiektowego – metoda analizy i syntezy filozoficznej szkoły platońskiej, * Modelowanie z użyciem diagramu klas UML, omówienie relacji dziedziczenia, agregacji i kompozycji * Naukę programowania w języka Java, z wykorzystaniem zarówno najprostszych narzędzi (kompilator, maszyna wirtualna i konsola) oraz zaawansowanego środowiska programistycznego (NetBeans), * Dziedziczenie, agregacja i kompozycja w Javie, * Metody i własności statyczne i dynamiczne, * Hermetyzacja metod i własności (pól), * Tablice jako obiekty, * Dziedziczenie, polimorfizm i metody wirtualne w Javie, * Stosowanie interfejsów, * struktury dynamiczne, typy generyczne oraz apliki archiwum Javy (JAR), * Strumienie, serializacja, obsługę wyjątków, tworzenie dokumentacji oprogramowania (javadoc), * Podstawy programowania okienkowego (awt, swing, stosowanie adapterów), * Podstawy wielowątkowości | | | | | |
| Wymagania formalne (przedmioty wprowadzające)16): | | Wstęp do Programowania, Matematyka Dyskretna 1 | | | | | |
| Założenia wstępne17): | | Konieczna podstawowa umiejętność programowania imperatywnego w językach strukturalnych oraz znajomość podstawowych algorytmów i umiejętność projektowania prostych struktur danych | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Efekty kształcenia18): | Student zna:   |  |  | | --- | --- | | 01 | Podstawy modelowania obiektowego | | 02 | Zasady stosowania hermetyzacji | | 03 | Dziedziczenie, agregację i kompozycję | | 04 | Pojęcia metody i własności (pola) statyczne i dynamiczne | | 05 | Interfejsy i polimorfizm | | 06 | Pojęcie typów generycznych | | 07 | 0bsługa wyjątków | | 08 | Obsługę strumieni oraz serializację | | 09 | Podstawowe struktury dynamiczne | | 10 | Pojęcie adaptera wykorzystywanego w programowaniu z użyciem okienek | | 11 | Podstawy programowania wielowątkowego | | Student potrafi:   |  |  | | --- | --- | | 12 | Zamodelować proste zagadnienie z wykorzystaniem diagramu klas UML | | 13 | Stosować prawidłowo hermetyzację | | 14 | Używać sensownie dziedziczenia, agregacji oraz kompozycji | | 15 | Wykorzystać i tworzyć metody i pola statyczne i dynamiczne | | 16 | Używać polimorfizmu i interfejsów | | 17 | Stosować typy generyczne | | 18 | Prawidłowo tworzyć obsługę wyjątków | | 19 | Wykorzystywać różnego rodzaju strumienie oraz serializację | | 20 | Tworzyć i zarządzać strukturami dynamicznymi | | 21 | Wykorzystywać adaptery | | 22 | Napisać prosty program wykorzystujący wielowątkowość | |
| Sposób weryfikacji efektów kształcenia19): | Efekty 12, 13, 14, 15, 16, 18, 19 – egzamin pisemny  Efekty 01, 02, 03, 04, 05, 07, 08 – ocena zadań wykonanych w czasie zajęć laboratoryjnych  Efekty 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22 – ocena projektu przygotowanego w domu przez studenta | |
| Forma dokumentacji osiągniętych efektów kształcenia 20): | Zadania bieżące wykonywane podczas laboratorium wraz z krótkim kolokwium  Sprawozdanie projektowe (kody + dokumentacja).  Egzaminy pisemne z ocenami | |
| Elementy i wagi mające wpływ na ocenę końcową21): | **Ćwiczenia laboratoryjne – 30%, egzamin pisemny – 60%, projekt – 10%** | |
| Miejsce realizacji zajęć22): | Wykład + ćwiczenia laboratoryjne – laboratorium komputerowe | |
| Literatura podstawowa i uzupełniająca23):   1. Bruce Eckel – „Thinking in Java” Edycja polska – Helion 2002 2. Cay S. Horstmann, Gary Cornell - “Java. Podstawy” - Helion 2013 | | |
| UWAGI24):  Liczba punktów do zdobycia za laboratoria: 7  Liczba punktów do zdobycia za sprawdzian praktyczny: 15  Liczba punktów do zdobycia za projekt: 10  Przeliczniki ocen:   * Laboratorium: 0,1,2 punkty – ndst, 3 punkty – dst, 4 punkty – dst+, 5 punktów – db, 6 punktów – db+, 7 punktów – bdb * Projekt: poniżej 5,1 punkta: ndst. <5,1;6,0) – dst, <6,0;7,0) – dst+, <7,0;8,0) – db, <8,0;9,0) – db+, <9,0; 10,0> - bdb * Egzamin: <0,0;7,5> – ndst, <7,6;8,9> - dst, <9,0;10,4> - dst+, <10,5;11,9> - db, <12,0;13,4> - db+, <13,5;15,0> - bdb   średnia ważona = 0,3 \* ocena z laboratorium + 0,1 \* ocena za projekt + 0,6 \* ocena z egzaminu  Przelicznik średniej ważonej na ocenę końcowa: <2,0;2,74> - ndst, <2,75;3,49> - dst, <3,5;3,74> - dst+, <3,75;4,49> - db, <4,5;4,74> - db+, <4,75;5,0> - bdb | | |

Wskaźniki ilościowe charakteryzujące moduł/przedmiot25) :

|  |  |
| --- | --- |
| Szacunkowa sumaryczna liczba godzin pracy studenta (kontaktowych i pracy własnej) niezbędna dla osiągnięcia zakładanych efektów kształcenia18) - na tej podstawie należy wypełnić pole ECTS2: | **180 H** |
| Łączna liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich: | **1.5 ECTS** |
| Łączna liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym, takich jak zajęcia laboratoryjne, projektowe, itp.: | **4.5 ECTS** |

Tabela zgodności kierunkowych efektów kształcenia efektami przedmiotu 26)

|  |  |  |
| --- | --- | --- |
| Nr /symbol efektu | Wymienione w wierszu efekty kształcenia: | Odniesienie do efektów dla programu kształcenia na kierunku |
| 01 | Zna podstawy modelowania obiektowego | W06, W08, W10,W17 |
| 02 | Zna zasady stosowania hermetyzacji | W06, W08, W10 |
| 03 | Zna dziedziczenie, agregację i kompozycję | W06, W08, W10, W17 |
| 04 | Zna pojęcia metody i własności (pola) statyczne i dynamiczne | W06, W08, W10 |
| 05 | Zna interfejsy i polimorfizm | W06, W08, W10, W17 |
| 06 | Zna pojęcie typów generycznych | W06, W08 |
| 07 | Zna obsługa wyjątków | W06, W08 |
| 08 | Zna obsługę strumieni oraz serializację | W06, W08 |
| 09 | Zna podstawowe struktury dynamiczne | W06, W08 |
| 10 | Zna pojęcie adaptera wykorzystywanego w programowaniu z użyciem okienek | W08 |
| 11 | Zna podstawy programowania wielowątkowego | W06,W10, W17 |
| 12 | Potrafi zamodelować proste zagadnienie z wykorzystaniem diagramu klas UML | U04, U12, U27 |
| 13 | Potrafi stosować prawidłowo hermetyzację | U04, U10, U27 |
| 14 | Potrafi używać sensownie dziedziczenia, agregacji oraz kompozycji | U04, U10, U27 |
| 15 | Potrafi wykorzystać i tworzyć metody i pola statyczne i dynamiczne | U04, U10, U27 |
| 16 | Potrafi używać polimorfizmu i interfejsów | U04, U10, U27 |
| 17 | Potrafi stosować typy generyczne | U04 |
| 18 | Potrafi prawidłowo tworzyć obsługę wyjątków | U10, U27 |
| 19 | Potrafi wykorzystywać różnego rodzaju strumienie oraz serializację | U04, U21 |
| 20 | Potrafi tworzyć i zarządzać strukturami dynamicznymi | U10, U21 |
| 21 | Potrafi wykorzystywać adaptery | U10 |
| 22 | Potrafi napisać prosty program wykorzystujący wielowątkowość | U10, U21 |

*Całkowity nakład czasu pracy - przyporządkowania ECTS2):*

|  |  |
| --- | --- |
| *Wykłady* | *18h* |
| *Ćwiczenia laboratoryjne* | *18h* |
| *Udział w konsultacjach* | *5h* |
| *Obecność na egzaminie* | *2h* |
| *Realizacja zadań domowych z laboratorium* | *5h x 9 = 45h* |
| *Praca własna studenta w domu (w tym projekt)* | *56h* |
| *Przygotowanie do egzaminu* | *36h* |
| *Razem:* | ***180 h*** |
|  | ***6 ECTS*** |

*W ramach całkowitego nakładu czasu pracy studenta - łączna liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:*

|  |  |
| --- | --- |
| *Wykłady* | *18h* |
| *Ćwiczenia laboratoryjne* | *18h* |
| *Udział w konsultacjach* | *5h* |
| *Egzamin* | *2h* |
| *Razem:* | *41 h* |
|  | ***1,5 ECTS*** |

*W ramach całkowitego nakładu czasu pracy studenta - łączna liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:*

|  |  |
| --- | --- |
| *Wykład (tu ma charakter praktyczny)* | *18h* |
| *Ćwiczenia laboratoryjne* | *18h* |
| *Praca własna studenta w domu (w tym projekt)* | *56h* |
| *Realizacja zadań domowych* | *5h x 9 = 45h* |
| *Udział w konsultacjach* | *5h* |
| *Razem:* | *142h* |
|  | ***4,5 ECTS*** |