



3. KARTA PRZEDMIOTU

| | |
|--------------------------------------|------------------------------|
| Kod przedmiotu | Z-IDN-U-406 |
| Nazwa przedmiotu | Programowanie obiektowe JAVA |
| Nazwa przedmiotu w języku angielskim | Object Programing JAVA |
| Obowiązuje od roku akademickiego | 2019/2020 |

USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

| | |
|----------------------------------|---|
| Kierunek studiów | INŻYNIERIA DANYCH |
| Poziom kształcenia | I stopień |
| Profil studiów | Praktyczny |
| Forma i tryb prowadzenia studiów | Studia niestacjonarne |
| Zakres | Wszystkie specjalności |
| Jednostka prowadząca przedmiot | Katedra Informatyki i Matematyki Stosowanej |
| Koordinator przedmiotu | Dr inż. Marcin Detka |
| Zatwierdził | Dr hab. inż. Artur Bartosik, prof. PŚk |

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

| | |
|---|--|
| Przynależność do grupy/bloku przedmiotów | Przedmiot kierunkowy |
| Status przedmiotu | Obowiązkowy |
| Język prowadzenia zajęć | Polski |
| Usytuowanie modułu w planie studiów - semestr | Semestr IV |
| Wymagania wstępne | Podstawy informatyki, Algorytmy i struktury danych |
| Egzamin (TAK/NIE) | Nie |
| Liczba punktów ECTS | 3 |

| Forma prowadzenia zajęć | wykład | ćwiczenia | laboratorium | projekt | inne |
|---------------------------|--------|-----------|--------------|---------|------|
| Liczba godzin w semestrze | 9 | | 18 | | |

EFEKTY UCZENIA SIĘ

| Kategoria | Symbol efektu | Efekty kształcenia | Odniesienie do efektów kierunkowych |
|-----------------------|---------------|---|-------------------------------------|
| Wiedza | W01 | Student zna i rozumie zasady funkcjonowania programów komputerowych zaimplementowanych z użyciem języka JAVA, z zastosowaniem paradygmatu obiektowego i sposobów jego uruchamiania w środowisku maszyny wirtualnej. | ID1_W11 |
| | W02 | Student ma podstawową wiedzę na temat typów danych i doboru tych typów do rozwiązania określonego zadania programistycznego. | ID1_W09 ID1_W10 |
| | W03 | Student ma wiedzę w zakresie syntaktyki i semantyki instrukcji sterujących języka JAVA oraz programowania w paradygmacie obiektowym. | ID1_W09 ID1_W10 |
| | W04 | Student rozumie budowę programu komputerowego oraz rolę funkcji wbudowanych i własnych w programie. | ID1_W10 ID1_W11 |
| | W05 | Student zna podstawowe komponenty do budowy graficznego interfejsu programu komputerowego. | ID1_W10 ID1_W11 |
| | W06 | Student ma podstawową wiedzę z zakresu diagnozowania i poprawy błędów w programie komputerowym. | ID1_W10 ID1_W11 |
| Umiejętności | U01 | Student posiada umiejętność konstruowania algorytmów służących rozwiązaniu różnych zadań programistycznych. | ID1_U09 |
| | U02 | Student potrafi wykorzystać środowisko programistyczne do zaprojektowania i zbudowania programu komputerowego, kompilacji, konsolidacji i testowania programu. | ID1_U09 |
| | U03 | Student potrafi zaprojektować i zbudować program komputerowy, korzystając z gotowych komponentów środowiska programistycznego oraz własnych rozwiązań programistycznych. | ID1_U09 |
| | U04 | Student potrafi wykonać analizę prostego kodu źródłowego, zdiagnozować błędy oraz wprowadzić modyfikacje w treść istniejącego programu. | ID1_U09 |
| | U05 | Student posiada umiejętność definiowania własnych funkcji oraz wykorzystania ich w tworzonym przez siebie programie komputerowym. | ID1_U09 |
| Kompetencje społeczne | K01 | Student rozumie potrzebę stałego uzupełniania i rozwijania wiedzy z zakresu programowania. | ID1_K01 |
| | K02 | Student potrafi pracować samodzielnie i w grupie, przyjmując w niej różne role. | ID1_K04 |

TREŚCI PROGRAMOWE

| Forma zajęć | Treści programowe |
|-------------|---|
| wykład | 1. Wprowadzenie do programowania obiektowego, paradygmaty programowania obiektowego: abstrakcja, hermetyzacja, dziedziczenie i polimorfizm. |
| | 2. Podstawowe konstrukcje języka Java, deklaracje klas, pola i metody statyczne. |
| | 3. Zasady tworzenia i inicjalizacji obiektów, konstruktory, przeciążanie nazw metod. Hermetyzacja w języku Java: kwalifikatory dostępu, interfejs a implementacja, pakiety. |
| | 4. Wielokrotne wykorzystanie implementacji: dziedziczenie i kompozycja, zasady dziedziczenia w języku Java. Polimorfizm. |
| | 5. Klasy, metody i pola ostateczne. Klasy abstrakcyjne i interfejsy. Zastosowania polimorfizmu. Typy uogólnione, zasady deklaracji klas i metod sparametryzowanych typami. |
| | 6. Tablice i kolekcje obiektów, możliwości list, map, kolejek i zbiorów. |

| | |
|--------------|--|
| | 7. Wyjątki: zasady specyfikacji wyjątków w metodach i konstruktorach, obsługa wyjątków. |
| | 8. Strumienie wejścia/wyjścia w języku Java, zasady obsługi systemu plików. Serializacja obiektów, metody sterowania serializacją. |
| laboratorium | 1. Wprowadzenie do programowania obiektowego w języku programowania Java. |
| | 2. Podstawowe konstrukcje języka Java, deklaracje klas, pola i metody statyczne. |
| | 3. Operatory, instrukcje sterujące, pętle, instrukcje warunkowe. |
| | 4. Zasady tworzenia i inicjalizacji obiektów, konstruktory, czyszczenie pamięci. |
| | 5. Hermetyzacja w języku Java: kwalifikatory dostępu, interfejs a implementacja, pakiety. |
| | 6. Wielokrotne wykorzystanie implementacji: dziedziczenie i kompozycja. |
| | 7. Polimorfizm, przeciążanie i przesłanianie metod. Klasy, metody i pola ostateczne. Klasy abstrakcyjne i interfejsy. |
| | 8. Kolekcje obiektów, możliwości list, map, kolejek i zbiorów. |

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

| Symbol efektu | Metody sprawdzania efektów kształcenia <i>(zaznaczyć X)</i> | | | | | |
|---------------|---|-----------------|-----------|---------|--------------|------|
| | Egzamin ustny | Egzamin pisemny | Kolokwium | Projekt | Sprawozdanie | Inne |
| W01 | | | X | | X | |
| W02 | | | X | | X | |
| W03 | | | X | | X | |
| W04 | | | X | | X | |
| W05 | | | X | | X | |
| W06 | | | X | | X | |
| U01 | | | X | | X | |
| U02 | | | X | | X | |
| U03 | | | X | | X | |
| U04 | | | X | | X | |
| U05 | | | X | | X | |
| K01 | | | X | | X | |
| K02 | | | X | | X | |

FORMA I WARUNKI ZALICZENIA

| Forma zajęć | Forma zaliczenia | Warunki zaliczenia |
|--------------|--------------------|---|
| wykład | zaliczenie z oceną | Uzyskanie 50% punktów z pisemnej pracy zaliczeniowej, której zakres dotyczy zarówno wykładów jak i laboratoriów. |
| laboratorium | zaliczenie z oceną | Student zdobywa punkty za aktywność na laboratoriach, za wykonanie sprawozdań do wybranych laboratoriów (wg wskazań prowadzącego) oraz za dwa sprawdziany praktyczne przy komputerach. Warunkiem zaliczenia jest uzyskanie 50% punktów. |

NAKŁAD PRACY STUDENTA

| Bilans punktów ECTS | | | | | | | |
|---------------------|--|---------------------|---|----|---|---|-----------|
| Lp. | Rodzaj aktywności | Obciążenie studenta | | | | | Jednostka |
| | | W | C | L | P | S | |
| 1. | Udział w zajęciach zgodnie z planem studiów | 9 | | 18 | | | h |
| 2. | Inne (konsultacje, egzamin) | 2 | | 2 | | | h |
| 3. | Razem przy bezpośrednim udziale nauczyciela akademickiego | 31 | | | | | h |
| 4. | Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego | 1,2 | | | | | ECTS |
| 5. | Liczba godzin samodzielnej pracy studenta | 44 | | | | | h |
| 6. | Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy | 1,8 | | | | | ECTS |
| 7. | Nakład pracy związany z zajęciami o charakterze praktycznym | 50 | | | | | h |
| 8. | Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym | 2 | | | | | ECTS |
| 9. | Sumaryczne obciążenie pracą studenta | 75 | | | | | h |
| 10. | Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i> | 3 | | | | | ECTS |

LITERATURA

1. Booch G., Rumbaugh J., Jacobson I., *UML przewodnik użytkownika*, Helion, Gliwice 2012.
2. Eckel B., *Thinking in Java*, wydanie 4, Helion, Gliwice 2006.
3. Gamma E., Helm R., Johnson R., Vlissides J., *Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku*, Wydawnictwa Naukowo-Techniczne, Warszawa 2005.
4. Gosling J., Bill J., Steele G., Bracha G., Buckley A., *The Java® Language Specification. Java SE 8 Edition*, <https://docs.oracle.com/javase/specs/jls/se8/jls8.pdf>, 2015.