



Elementy programowania gier komputerowych

Karta opisu przedmiotu

Informacje podstawowe

Kierunek studiów Tworzenie Przestrzeni Wirtualnych i Gier	Cykl dydaktyczny 2026/2027	
Specjalność -	Kod przedmiotu WIPPWGS.II.16860.26	
Jednostka organizacyjna Wydział Informatyki	Języki wykładowe polski	
Poziom kształcenia Studia licencjackie I stopnia	Obligatoryjność Obowiązkowy	
Forma studiów Stacjonarne	Blok zajęciowy Przedmioty kierunkowe	
Profil studiów Ogólnoakademicki	Przedmiot powiązany z badaniami naukowymi Nie	
Koordynator przedmiotu	Witold Alda	
Prowadzący zajęcia	Witold Alda	
Okres Semestr 1	Forma zaliczenia Egzamin	Liczba punktów ECTS 4
	Forma prowadzenia i godziny zajęć Wykład: 14 Ćwiczenia laboratoryjne: 28	

Cele kształcenia dla przedmiotu

C1	Celem jest jest zapoznanie studentów z podstawami programowania w języku wysokiego poziomu. Przedmiot ma charakter ogólny, jednak przykłady powinny zwracać uwagę na interakcję i zastosowanie w grach komputerowych. Przewidywanym językiem jest Python, ze względu na łatwość rozpoczęcia w nim pracy, popularność i częste zastosowanie w zagadnieniach gier.
----	--

Efekty uczenia się dla przedmiotu

Kod	Efekty w zakresie	Kierunkowe efekty uczenia się	Metody weryfikacji
Wiedzy - Student zna i rozumie:			
W1	Student zna i rozumie podstawowe pojęcia z programowania strukturalnego w Pythonie.	PPWG1A_W03	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
W2	Student ma wiedzę na temat struktur danych w Pythonie.	PPWG1A_W03	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
W3	Student zna i rozumie zasady podstawowe algorytmy w typowych zadaniach informatycznych.	PPWG1A_W01, PPWG1A_W03	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
W4	Student ma elementarną wiedzę z zakresu biblioteki PyGame.	PPWG1A_W01, PPWG1A_W03	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
Umiejętności - Student potrafi:			
U1	Student potrafi zanalizować prosty problem informatyczny i zrealizować go w postaci działającego programu.	PPWG1A_U01	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
U2	Student potrafi korzystać efektywnie ze środowiska do edytowania i uruchamiania programów.	PPWG1A_U01	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
U3	Student potrafi napisać prostą grę w Pythonie.	PPWG1A_U01, PPWG1A_U07	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
U4	Student potrafi użyć narzędzi dostarczonych przez PyGame i inne moduły.	PPWG1A_U01, PPWG1A_U07	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium, Egzamin
Kompetencji społecznych - Student jest gotów do:			
K1	Student rozumie potrzebę stałego uzupełniania swojej wiedzy i umiejętności programowania w zakresie środowiska Python.	PPWG1A_K01	Aktywność na zajęciach

Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Student zapoznaje się z podstawami języka Python, środowiska w jakim się go używa oraz podstawowymi algorytmami informatycznymi. Nabiera umiejętności w pisaniu i uruchamianiu programów.

Nakład pracy studenta

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane aktywności
Wykład	14
Ćwiczenia laboratoryjne	28
Egzamin lub kolokwium zaliczeniowe	2
Przygotowanie do zajęć	28
Samodzielne studiowanie tematyki zajęć	28
Dodatkowe godziny kontaktowe	5
Łączny nakład pracy studenta	Liczba godzin 105
Liczba godzin kontaktowych	Liczba godzin 42

* godzina (lekcyjna) oznacza 45 minut

Treści programowe

Lp.	Treści programowe	Efekty uczenia się dla przedmiotu	Formy prowadzenia zajęć
1.	Podstawowe pojęcia związane z językiem i programowaniem: zmienne i ich deklarowanie, słowa kluczowe, najważniejsze instrukcje, operatory i wyrażenia. Operacje wejścia. Elementy obiektowe i strukturalne w Pythonie. Klasy w Pythonie.	W1, W2	Wykład
2.	Pojęcie funkcji, jej użycie i zastosowanie. Zmienne lokalne i globalne.	W1, W2	Wykład
3.	Instrukcje warunkowe, instrukcje pętli.	W1, W2	Wykład
4.	Tworzenie programów interaktywnych.	W1, W2, W3, U3, K1	Wykład, Ćwiczenia laboratoryjne
5.	Dołączanie modułów systemowych, w tym biblioteki PyGame	W1, W2, W3, W4, U3, U4, K1	Wykład, Ćwiczenia laboratoryjne
6.	Ćwiczenia laboratoryjne obejmują: 1. Zapoznanie się ze środowiskiem Pythona 2. Implementację prostych programów 3. Przygotowanie prostego programu interaktywnego w stylu gry.	U1, U2, U3, U4, K1	Ćwiczenia laboratoryjne

Informacje rozszerzone

Metody i techniki kształcenia :

Dyskusja, Metoda problemowa (ang. Problem Based Learning), Metoda projektowa (ang. Project Based Learning), Metoda warsztatowa (ang. workshop), Wykład

Rodzaj zajęć	Metody zaliczenia	Warunki zaliczenia przedmiotu
Wykład	Egzamin	Warunkiem zaliczenia wykładu jest zdanie egzaminu pisemnego.
Ćwiczenia laboratoryjne	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Kolokwium	Warunkiem zaliczenia ćwiczeń laboratoryjnych jest wykonanie zadań programistycznych na laboratorium i zdanie kolokwium końcowego.

Dodatkowy opis

Wykłady z przedmiotu i pozostałe zajęcia będą odbywać się w salach. Dotyczy to także zaliczeń i egzaminów odbywających się w sesjach egzaminacyjnych.

Warunki i sposób zaliczenia poszczególnych form zajęć, w tym zasady zaliczeń poprawkowych, a także warunki dopuszczenia do egzaminu

Zaliczenie poszczególnych zajęć laboratoryjnych dokonuje się na podstawie obecności studenta i wykonania przez niego przeznaczonego na zajęcia zadania. Zaliczenie poprawkowe może odbyć się na zajęciach innej grupy lub wyjątkowo poza zajęciami po kontakcie z prowadzącym.

Sposób obliczania oceny końcowej

1. Aby uzyskać pozytywną ocenę końcową niezbędne jest uzyskanie pozytywnej oceny z laboratorium oraz z egzaminu z wykładu. 2. Obliczamy średnią ważoną sr z ocen z laboratorium (50%) i wykładów (50%) uzyskanych we wszystkich terminach. 3. Wyznaczymy ocenę końcową OK na podstawie zależności: $\text{if } sr > 4.75 \text{ then } OK \rightarrow 5.0 \text{ else if } sr > 4.25 \text{ then } OK \rightarrow 4.5 \text{ else if } sr > 3.75 \text{ then } OK \rightarrow 4.0 \text{ else if } sr > 3.25 \text{ then } OK \rightarrow 3.5 \text{ else } OK \rightarrow 3$. 4. Jeżeli pozytywną ocenę z laboratorium i z egzaminu uzyskano w pierwszym terminie oraz ocena końcowa jest mniejsza niż 5.0 to ocena końcowa jest podnoszona o 0.5

Sposób i tryb wyrównywania zaległości powstałych wskutek nieobecności studenta na zajęciach

Odrobienie zajęć laboratoryjnych w innej grupie, a jeśli to niemożliwe - indywidualne wykonanie ćwiczenia laboratoryjnego po umówieniu się z prowadzącym.

Wymagania wstępne i dodatkowe

Brak wymagań wstępnych.

Zasady udziału w poszczególnych zajęciach, ze wskazaniem, czy obecność studenta na zajęciach jest obowiązkowa

Wykład: Studenci uczestniczą w zajęciach poznając kolejne treści nauczania zgodnie z sylabusem przedmiotu. Studenci winni na bieżąco zadawać pytania i wyjaśniać wątpliwości. Rejestracja audiowizualna wykładu wymaga zgody prowadzącego. Wykład kończy się egzaminem. Ćwiczenia laboratoryjne: Studenci wykonują ćwiczenia laboratoryjne zgodnie z materiałami udostępnionymi przez prowadzącego. Student jest zobowiązany do przygotowania się w przedmiocie wykonywanego ćwiczenia, co może zostać zweryfikowane kolokwium w formie ustnej lub pisemnej. Zaliczenie zajęć odbywa się na podstawie zaprezentowania rozwiązania postawionego problemu. Zaliczenie modułu jest możliwe po zaliczeniu wszystkich zajęć laboratoryjnych. Laboratoria są obowiązkowe.

Literatura

Obowiązkowa

1. Rob S. Miles, Python: zacznij programować, Helion 2019.
2. Naomi R. Ceder: Python szybko i prosto, Helion 2019.
3. Materiały do kursu programowania w języku Python z Runestone Academy (<https://runestone.academy/runestone/books/published/fopp/index.html>).

Dodatkowa

1. Swaroop C. H., A byte of Python, 2005, free book.
2. Sweigart, A. Invent your own computer games with Python, 4th Edition, No Starch Press, Inc., 2017

Kierunkowe efekty uczenia się

Kod	Treść
PPWG1A_K01	Samodzielnie podejmuje niezależne prace, wykazując się umiejętnościami zbierania, analizowania i interpretowania informacji, rozwijania idei i formułowania krytycznej argumentacji oraz wewnętrzną motywacją i umiejętnością organizacji pracy.
PPWG1A_U01	Potrafi definiować oraz realizować projekty gier, wymagające twórczego i innowacyjnego podejścia , wybierając, wykorzystując i w razie potrzeb adaptując stosowne narzędzia programistyczne i sprzętowe.
PPWG1A_U07	Potrafi stosować narzędzia z zakresu projektowania graficznego, grafiki komputerowej i komunikacji wizualnej.
PPWG1A_W01	Ma wiedzę z matematyki obejmującą logikę elementarną, podstawy geometrii analitycznej, algebry, rachunku prawdopodobieństwa i statystyki w zakresie potrzebnym w projektowaniu gier.
PPWG1A_W03	Zna podstawy programowania w językach C#, JavaScript, Python i Kotlin oraz środowiska do tworzenia programów w tych językach, w zakresie użycia w grach i grafice. Zna podstawy popularnych systemów operacyjnych: Linux, Windows, Android i iOS oraz techniki pracy z nimi.