

KARTA PRZEDMIOTU (SYLABUS)

Opis przedmiotu

| | | | | |
|---|--------------------|--|-----------------------------------|---------------------|
| Kod przedmiotu | | Nazwa przedmiotu | PODSTAWY PROGRAMOWANIA | |
| IT/P/1/ST/B1-2 | | | PROGRAMMING FUNDAMENTALS | |
| Język wykładowy | | język polski | | |
| Rok akademicki | | 2024/2025 | | |
| Kierunek | | Informatyka techniczna | | |
| w zakresie | | | | |
| Poziom studiów | | studia pierwszego stopnia | | |
| Profil studiów | | praktyczny | | |
| Forma studiów | | studia stacjonarne | | |
| Semestr / semestry | | 1 | | |
| Przynależność do grupy zajęć | | B1. Grupa zajęć kierunkowych – obowiązkowych | | |
| Status przedmiotu | | obowiązkowy | | |
| Formy realizacji zajęć dydaktycznych, wymiar, punkty ECTS | | Forma zajęć | Liczba godzin zajęć dydaktycznych | Liczba punktów ECTS |
| | | Wykład | 30 [h] | 6 ECTS |
| | | Laboratorium | 45 [h] | |
| Powiązanie przedmiotu | z profilem studiów | związany z kształtowaniem umiejętności praktycznych | | 3,5 ECTS |
| | z uprawnieniami | służy do zdobywania przez studenta kompetencji inżynierskich | | 4 ECTS |
| | z dyscypliną | Informatyka techniczna i telekomunikacja | | 6 ECTS |
| Forma nauczania | | tradycyjna – zajęcia zorganizowane w Uczelni i/lub zajęcia z wykorzystaniem metod i technik kształcenia na odległość (max. 1,2 ECTS) | | |
| Wymagania wstępne | | | | |
| Jednostka prowadząca | | Katedra Informatyki i Teleinformatyki | | |
| Koordynator | | dr Artur Bartoszewski | | |
| Adres strony internetowej pjo | | www.wteii.uniwersytetradom.pl | | |
| Adres e-mail, telefon koordynatora | | artur.bartoszewski@urad.edu.pl | | |

EFEKTY UCZENIA SIĘ, TREŚCI PROGRAMOWE, REALIZACJA ZAJĘĆ DYDAKTYCZNYCH, WERYFIKACJA EFEKTÓW UCZENIA SIĘ

| | |
|--------------------|---|
| Cel kształcenia: | Poznanie możliwie szerokiego zakresu zagadnień podstawowych związanych z programowaniem komputerów, strukturami danych i algorytmami oraz związanych z nimi technik tworzenia programów. Wykłady oraz ćwiczenia o charakterze laboratoryjnym mają na celu opanowanie umiejętności formułowania problemu, tworzenia algorytmów i implementacji w języku programowania, na przykładzie języka C++ |
| Treści programowe: | <p>Wykład [W1, W2, U1, U2, U3, K1]:</p> <ol style="list-style-type: none"> 1. Języki programowania i ich klasyfikacja. Środowiska programistyczne, kompilacja i interpretacja. Uruchamianie i testowanie programu. Problem poprawności i złożoności programu. 2. Programowanie proceduralne: struktura programu, funkcje, moduły i biblioteki. 3. Pojęcia podstawowe: problem, algorytm, program, metody zapisu algorytmów. 4. Podstawowe konstrukcje języka C/C++: identyfikatory, typy standardowe, typy wyliczeniowe, zmienne i stałe, wyrażenia, funkcje standardowe. Instrukcje warunkowe. 5. Pojęcie iteracji i rekurencji, pętle programowe. 6. Praca z tablicami jedno i wielowymiarowymi. Algorytmy sortowania. 7. Funkcje, parametry formalne i aktualne, przekazywanie parametrów do funkcji. Zasięg widoczności zmiennych (lokalny i globalny), przeciążanie funkcji. 8. Przykłady znanych algorytmów: (między innymi: algorytm Euklidesa, rozwiązywanie równań kwadratowych itp.). Przykłady wykorzystania poznanych struktur programistycznych do rozwiązywania problemów. 9. Łańcuch znaków i klasa String. 10. Obsługa plików tekstowych. 11. Złożone typy danych (struktury, unie, pola bitowe). 12. Wskaźniki i zmienne dynamiczne, dynamiczne struktury danych. <p style="text-align: right;">Suma: 30 [h]</p> |

| | |
|--|---|
| | <p>Laboratorium [PP, U1, U2, U3, K1, K2]:</p> <ol style="list-style-type: none"> 1. Rozgałęzienia w programie, instrukcje wyboru oraz pętle programowe, iteracyjne rozwiązywanie problemów numerycznych (algorytmy działające na ciągach liczbowych itp.). 2. Operacje na tablicach, tablicowanie funkcji, działania na znakach i łańcuchach znaków, wyznaczanie min/max n elementowego ciągu liczbowego, sortowania (sortowanie bąbelkowe, łączenie dwóch ciągów uporządkowanych itp.). 3. Podział programu na funkcje, przekazywanie parametrów do funkcji, przeciążanie funkcji. 4. Złożone typy danych (łańcuchy znaków i struktury). 5. Operacje na plikach tekstowych. 6. Rekurencja (silnia, potęga, itp.). 7. Wskaźniki i zmienne dynamiczne. 8. Listy jednokierunkowe, listy monotoniczne, kolejki LIFO i FIFO. <p style="text-align: right;">Suma: 45 [h]</p> |
| Metody dydaktyczne (kształcenia): | <ul style="list-style-type: none"> – metody podające (wykład informacyjny) – metody problemowe (wykład problemowy, wykład konwersatoryjny) – metody aktywizujące (metoda przypadków, dyskusja dydaktyczna), – metody praktyczne (pokaz, ćwiczenia laboratoryjne). |
| Rygor zaliczenia, kryteria oceny osiągniętych efektów uczenia się, sposób obliczania oceny końcowej: | <p>Warunkiem zaliczenia przedmiotu jest osiągnięcie wszystkich wymaganych efektów uczenia się określonych dla danego przedmiotu. Uzyskanie pozytywnych ocen ze wszystkich form zajęć wchodzących w skład danego przedmiotu jest równoznaczne z jego zaliczeniem i zdobyciem przez studenta liczby punktów ECTS przyporządkowanej temu przedmiotowi. Sposób obliczenia oceny końcowej z przedmiotu określa regulamin studiów. Sposób obliczania oceny z poszczególnych form zajęć przedstawia się następująco:</p> <p>Na ocenę z laboratorium składa się: średnia ocen z dwóch kolokwium praktycznych.</p> <p>Na ocenę z wykładu składa się wynik egzaminu pisemnego.</p> <p>Ocena z egzaminu - zdobyte punkty przeliczane zostają na ocenę wg skali:</p> <p>Ocena 2 poniżej 51%</p> <p>Ocena 3 od 51%</p> <p>Ocena 3,5 od 61%</p> <p>Ocena 4 od 71%</p> <p>Ocena 4,5 od 81%</p> <p>Ocena 5 od 91%</p> |

| Efekty uczenia się dla przedmiotu w odniesieniu do efektów kierunkowych i formy zajęć | | | | Metody weryfikacji efektów uczenia się | |
|---|---|------------------------------------|------------------------------------|--|----------------------------|
| Numer efektu uczenia się | Opis efektów uczenia się dla przedmiotu (PEU) Student, który zaliczył przedmiot (W) zna i rozumie/ (U) potrafi /(K) jest gotów do: | Kierunkowy efekt uczenia się (KEU) | Forma zajęć | Forma weryfikacji (zaliczeń) | Metody sprawdzania i oceny |
| W1 | zagadnienia programowania komputerów, systemów operacyjnych, struktur danych i algorytmów oraz związanych z nimi technik tworzenia i uruchamiania programów | K_WG06 K_WG08 | wykład, ćwiczenia laboratoryjne | egzamin, zaliczenie | egzamin, kolokwium |
| W2 | składnię i polecenia języka C oraz elementy składni języka C++ (z wyłączeniem programowania obiektowego). | K_WG06 | wykład | egzamin, zaliczenie | egzamin, kolokwium |
| U1 | zapisywać i implementować algorytmy w językach programowania C i C++ | K_UW04 | wykład | egzamin, zaliczenie | egzamin, kolokwium |
| U2 | dobierać właściwe struktury danych oraz konstrukcje algorytmiczne do rozwiązywanego problemu oraz umie zaprogramować algorytm w języku wysokiego poziomu | K_UW05 | wykład, ćwiczenia laboratoryjne | egzamin, zaliczenie | egzamin, kolokwium |
| U3 | formułować problemy w języku C++ oraz tworzyć i uruchamiać programy w wybranym środowisku programistycznym | K_UW04 | wykład, ćwiczenia laboratoryjne | egzamin, zaliczenie | egzamin, kolokwium |
| K1 | poszerzać swoją wiedzę i doskonalić umiejętności potrzebne w wykonywaniu zawodu programisty | K_KK01 | wykład | egzamin | egzamin |
| K2 | zaplanować pracę pod kątem zakładanych rezultatów, określić priorytetowe zadania w oparciu o zasady skutecznego działania. | K_KO05 | ćwiczenia laboratoryjne | zaliczenie | kolokwium |

| Literatura i pomoce naukowe | |
|-----------------------------|--|
| 1. | Grębosz J. : Opus magnum C++. Programowanie w języku C++. Wydanie III poprawione, Helion 2024. |
| 2. | Stroustrup B.: Język C++. Kompendium wiedzy. Wydanie IV, Helion 2014 |
| 3. | Kerninghan B. W., Ritchie D. M.: język ANSI C, Wydawnictwo Naukowo Techniczne, Warszawa 1994. |
| 4. | Sysło M.M.: Piramidy, szyszki i inne konstrukcje algorytmiczne, WSiP, Warszawa 1998. |
| 5. | Wirth N.: Algorytmy + struktury danych = programy, WNT, Warszawa 2004. |
| 6. | Simon L. D., Drozdek A.: Struktury danych w języku C, WNT, Warszawa 1996. |

| Nakład pracy studenta potrzebny do osiągnięcia zakładanych efektów uczenia się – bilans punktów ECTS | | | |
|--|-----------------------------|---|---------------------|
| Udział w zajęciach, aktywność | Obciążenie studenta [h] | | |
| | Inne godz. kontaktowe (IGK) | Zajęcia bez nauczyciela-praca własna studenta (ZBN) | Zajęcia dydaktyczne |
| Udział w wykładach | X | X | 30 [h] |
| Udział w ćwiczeniach / laboratoriach / projektach / seminariach | X | X | 45 [h] |
| Udział w konsultacjach | 5 [h] | X | X |
| Przygotowanie do wykładów / ćwiczeń / laboratoriów / projektów / seminariów | X | 70 [h] | X |
| Przygotowanie do zaliczenia/egzaminu | | | |
| Sumaryczne obciążenie pracą studenta | 5 [h] / 0,2 ECTS | 70 [h] / 2,8 ECTS | 75 [h] / 3 ECTS |
| Punkty ECTS za przedmiot | 6 ECTS | | |

| Informacje dodatkowe, uwagi |
|---|
| <p>W przypadku studentów ze szczególnymi potrzebami, w tym: z niepełnosprawnością, przewlekle chorych, określone powyżej (w karcie) metody i formy weryfikacji efektów uczenia się dostosowuje się odpowiednio do indywidualnych potrzeb tych studentów.</p> <p>Szczegółowe zasady i formy wsparcia studentów ze szczególnymi potrzebami: w tym z niepełnosprawnością, przewlekle chorych podczas zajęć, zaliczeń i egzaminów określono w: Regulaminie Studiów, Zasadach Studiowania, Procedurze dotyczącej zapewnienia dostępności procesu kształcenia studentom ze szczególnymi potrzebami, w tym: z niepełnosprawnością, przewlekle chorych.</p> |