



Karta przedmiotu

Nazwa i kod przedmiotu	Wstęp do informatyki, PG_00058877						
Kierunek studiów	Nanotechnologia						
Data rozpoczęcia studiów	październik 2022 r.	Rok akademicki realizacji przedmiotu			2023/2024		
Poziom kształcenia	I stopnia - inżynierskie	Grupa zajęć			Grupa zajęć obowiązkowych z zakresu kierunku studiów		
Forma studiów	stacjonarne	Sposób realizacji			na uczelni		
Rok studiów	2	Język wykładowy			polski		
Semestr studiów	3	Liczba punktów ECTS			2.0		
Profil kształcenia	ogólnoakademicki	Forma zaliczenia			zaliczenie		
Jednostka prowadząca	Wydział Fizyki Technicznej i Matematyki Stosowanej -> Katedra Fizyki Ciała Stałego						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot	dr inż. Marek Augustyniak					
	Prowadzący zajęcia z przedmiotu	dr hab. Maciej Bobrowski dr inż. Marek Augustyniak					
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	30.0	0.0	0.0	45
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów	Udział w konsultacjach	Praca własna studenta	RAZEM		
	Liczba godzin pracy studenta	45	2.0	3.0	50		
Cel przedmiotu	Część 1: (Arkusz/Python) Trening funkcji arkusza kalkulacyjnego Wprowadzenie do języka skryptowego Python (programowanie strukturalne i wstęp do programowania obiektowego) Przegląd i demonstracja wybranych bibliotek. Część 2: (Linux/C/Latex): Nauczenie umiejętności pracy z systemem składu dokumentów Latex: kompilacja, preambuła, wzory matematyczne. Nauczenie podstaw programowania w języku C: zmienne, instr. warunkowe, pętle, tablice jedno i dwuwymiarowe. Nauczenie umiejętności pisania podstawowych programów strukturalnych.						
Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu			Sposób weryfikacji i oceny efektu		
	[K6_U01] Potrafi uczyć się samodzielnie, pozyskiwać informacje z literatury, baz danych oraz innych właściwie dobranych źródeł.	Student potrafi samodzielnie rozwiązywać zadania rozsyłane przez prowadzącego zajęcia na podstawie danych literaturowych, skryptu udzielonego przez prowadzącego oraz z wskazanych książek.			[SU1] Ocena realizacji zadania [SU4] Ocena umiejętności korzystania z metod i narzędzi [SU3] Ocena umiejętności wykorzystania wiedzy uzyskanej w ramach przedmiotu		
	[K6_W04] Ma podstawową wiedzę o narzędziach informatycznych (procesorach tekstu, arkuszach kalkulacyjnych, itd.), tworzeniu prezentacji multimedialnych oraz programowaniu i grafice komputerowej.	Student potrafi praktycznie tworzyć i obsługiwać arkusze kalkulacyjne, utworzyć prezentacje multimedialne oraz dokumenty w procesorach tekstu.			[SW1] Ocena wiedzy faktograficznej [SW2] Ocena wiedzy zawartej w prezentacji		
	[K6_U03] Posiada umiejętność programowania w wybranym języku oraz stosowania podstawowych pakietów oprogramowania.	Student potrafi wykorzystać elementy programowania proceduralnego i pisać samodzielnie programy.			[SU1] Ocena realizacji zadania [SU4] Ocena umiejętności korzystania z metod i narzędzi		

Treści przedmiotu	<p>Część 1:1. Informatyka z lotu ptaka; m.in.: - kierunki ewolucji software i hardware- poziomy systemów i języków programowania- technologie aktualnie potrzebne na rynku pracy- test wstępny - poziom dotychczas zdobytych umiejętności informatycznych2. Arkusz kalkulacyjny - repetytorium i rozszerzenie umiejętności, m.in.: - Podobieństwa i różnice w wariantach software (evolucja interfejsu i konwencji w Excel, specyfika LibreOffice i OpenOffice)- Umiejętności podstawowe (praca z ograniczoną ilością danych i formuł)- Praca ze zmiennymi, macierzami i formularzami - Zadania wymagające operacji wieloetapowych- Test z umiejętności podstawowych oraz mini-projekt3. Python - składnia, algorytmy, użycie wybranych bibliotek- Python na tle innych języków skryptowych i nie-skryptowych; kompilacja vs interpretacja; strategie przydziału pamięci, struktury danych, pętle i instrukcje warunkowe, operacje wejścia/wyjścia- Wprowadzenie do IDE: PyCharm Community Edition; opcje uproszczone - interpretery online- Przecwiczenie składni, m.in.: if, for, while, try, tworzenie krotek ("tuples") oraz list- Elementy pracy z obiektami - możliwości i ograniczenia w Python- Test ze znajomości składni i konwencji języka- Przegląd wybranych bibliotek- Projekt: np. zaprogramowanie rozwiązania numerycznego problemu ruchów Browna ze stałą siłą wymuszającą ruch cząstki lub zamodelowanie prostego symulatora giełdy</p> <p>Część 2::1. System operacyjny Linux.Wstęp prowadzącego: Linux wśród innych systemów operacyjnych, historia, zastosowania, budowa, zalety/wady systemu, przyszłość. Tryb graficzny i tekstowy, obciążenie, odchudzanie systemu, optymalizacja pracy, urządzenia zewnętrzne: dyski, drukarki, skanery, itp. Polecenia/programy/procesy. Sposób wykonywania poleceń (opcje, parametry).Drzewo katalogów, nawigowanie po drzewie, tworzenie/usuwanie katalogów, kopiowanie plików i katalogów z opcjami, backup danych, listowanie plików z opcjami, znaki specjalne, nazwy plików, zmiana położenia i/lub zmiana nazwy plików/katalogów, bezpieczne przeglądanie zawartości plików, uprawnienia do plików/katalogów w systemie wielu użytkowników, usuwanie plików i katalogów z plikami, standardowe strumienie (STDOUT, STDERR, STDIN), przekierowywanie strumieni danych, przekierowywanie strumienia wejściowego, znak rury i łączenie poleceń w złożone kombajny poleceń, przeszukiwanie zawartości plików (grep), szukanie plików lub katalogów (find),praca z kolumnami danych (język awk ale tylko do tego wyłącznie zastosowania),tł, plan pierwszy: fg, bg, &, dodatki: kopiowanie myszą, historia poleceń (strzałki górna/dolna), tabulator (dokańczanie poleceń, wynajdywanie plików/katalogów) manuale elektroniczne do poleceń, Edycja plików tekstowych: edytory vi oraz vim:tryby pracy (edycja i polecenia), zachowywanie zmian/zawartości, poruszanie się poziomo i pionowo, buforowanie danych (yankowanie) wierszami i kolumnami, dodatki: (de)kapitalizacja, przeszukiwanie, łączenie linii, zamiana znaków/słów, cofanie/powtarzanie poleceń, plik ~/.vimrc oraz opcje pliku z konfiguracjami programu vim. Konfiguracja powłoki, zmienne powłoki, pliki konfiguracyjne, przykłady działań i efektów działań na zmiennych, procesy, działania na procesach, zasoby komputera, monitorowanie pracy. Praca w sieci: wstęp prowadzącego (topologia, urządzenia, rozwiązania sprzętowe), logowanie na zdalne komputery, sprawdzanie konfiguracji sieci, podgląd na innych użytkowników, kopiowanie danych pomiędzy komputerami, programy uruchamiane z innych komputerów, współpraca Windows/Linux.+ sprawdzian</p> <p>Przykłady problemów do rozwiązania na sprawdzianie: *****1. Przy pomocy komendy df -k oraz języka awk (w jednym poleceniu używającym strumieni) sprawdź, czy ilość zajętego miejsca na danej partycji dysku oraz ilość pustego miejsca na danej partycji dysku sumują się do całkowitej objętości odpowiedniej partycji dysku.2. Zadanie trudniejsze. Przy pomocy komendy ifconfig oraz narzędzi przeszukujących dane w plikach tekstowych, spróbuj odnaleźć przypisane adresy IP (w protokole TCP IP)do kart sieciowych, które oznaczone są symbolami interfejsów, np. interfejs sieciowy eth0, eth1.Ma to być jedno złożone polecenie, które wypisze jeden pod drugim, adresy IP.2. System składu dokumentów Latex.----- Wstęp prowadzącego: co to jest Latex i po co, historia, zastosowanie, możliwości, zalety/wady systemu. Źródło/kompilacja. Opis minimum wymagań do pisania wzorów matematycznych: układy stron, (pod)rozdziały, preambuła, pakiety, zmienne, kompilacja, uzyskiwanie plików dvi, ps, pdf.System składu dokumentu w latexu, ze względu na ograniczenia czasowe całość ograniczona do prawie wyłącznie wzorów matematycznych: tryby pisania wzorów (w linii tekstu, oddzielnie), dostępne środowiska, jednolinijkowce, wielolinijkowce (wyprowadzanie wzorów), symbole greckie zmiennych i symbole typowych funkcji matematycznych (próbowanie domyślania się oznaczeń latexowych),ułamki, całki, sumy, różniczki, wyznaczniki, macierze i inne symbole, w miarę możliwości czasowych.+ sprawdzian .Przykład problemów do rozwiązania na sprawdzianie:- Mając do dyspozycji ostateczny wygląd dokumentu w pliku pdf oraz plik-szablon do kodu źródła Latexa, uzyskaj taki sam dokument pdf jaki uzyskał prowadzący.</p>		
Wymagania wstępne i dodatkowe	Brak wymagań wstępnych. Przewiduje się przerobienie podstawowych informacji o działaniu systemów operacyjnych, i podstaw programowania.		
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe) sprawdziany na laboratoriach	Próg zaliczeniowy 51.0%	Składowa oceny końcowej 100.0%
Zalecana lista lektur	Podstawowa lista lektur	1. Brian. W. Kernighan, Dennis. M. Ritchie, „ANSI C”.	
	Uzupełniająca lista lektur	1. John S. Gray, „Komunikacja między procesami w Unixie”,Wydawnictwo RM, Warszawa, 1998. 2. Dale Dougherty, Arnold Robbins, sed i awk, Wydawnictwo O'Reilly, 2002, 3. William H. Press, Saul. A. Teukolsky, William T. Vetterling, Brian P. Flannery, Numerical recipes in C, Wydawnictwo: Cambridge Univ. Press, 1992, 4. Eleen Frisch, Unix, Administracja Systemu, Wydawnictwo O'Reilly, 1996,	
	Adresy eZasobów	Adresy na platformie eNauczanie: Wstęp do informatyki -- jesień 2023 - MA, MB - Moodle ID: 32057 https://enauczanie.pg.edu.pl/moodle/course/view.php?id=32057	

<p>Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania</p>	<p>Część 1: @ Arkusz kalkulacyjny: utwórz algorytm, który będzie porównywał listę osób faktycznie obecnych na np. konferencji do dostępnej listy wszystkich potencjalnych uczestników. @ Arkusz kalkulacyjny: wyznacz kąt między dwoma wektorami w przestrzeni 3D lub n-wymiarowej @ Python: zaprogramowanie rozwiązania numerycznego problemu ruchów Browna ze stałą siłą wymuszającą ruch cząstki lub zamodelowanie prostego symulatora giełdy</p> <p>Część 2: @ Mając do dyspozycji ostateczny wygląd dokumentu w pliku pdf oraz plik-szablon do kodu źródła latexa, uzyskaj taki sam dokument pdf jaki uzyskał prowadzący. @ Przy pomocy komendy df -k oraz języka awk (w jednym poleceniu używającym strumieni) sprawdź, czy ilość zajętego miejsca na danej partycji dysku oraz ilość pustego miejsca na danej partycji dysku sumują się do całkowitej objętości odpowiedniej partycji dysku. @ Zadanie trudniejsze. Przy pomocy komendy ifconfig oraz narzędzi przeszukujących dane w plikach tekstowych, spróbuj odnaleźć przypisane adresy IP (w protokole TCP IP) do kart sieciowych, które oznaczone są symbolami interfejsów, np. interfejs sieciowy eth0, eth1. Ma to być jedno złożone polecenie, które wypisze jeden pod drugim, adresy IP.</p>
<p>Praktyki zawodowe w ramach przedmiotu</p>	<p>Nie dotyczy</p>