Uniwersytet Gdański

Karta przedmiotu

Nazwa i kod przedmiotu	Warsztat programisty, PG_00204155		
Kierunek studiów	Informatyka (P)		
Data rozpoczęcia studiów	październik 2026 r.	Rok akademicki realizacji przedmiotu	2026/2027
Poziom kształcenia	I stopnia - licencjackie	Grupa zajęć	Grupa zajęć obowiązkowych z zakresu kierunku studiów Grupa zajęć powiązanych z praktycznym przygotowaniem zawodowym - profil praktyczny
Forma studiów	stacjonarne	Sposób realizacji	na uczelni
Rok studiów	1	Język wykładowy	polski
Semestr studiów	1	Liczba punktów ECTS	4.0
Profil kształcenia	praktyczny	Forma zaliczenia	zaliczenie
Jednostka prowadząca	Rektor -> Wydział Matematyki, Fizyki i Informatyki -> Instytut Informatyki -> Zakład Optymalizacji Kombinatorycznej		
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot	mgr Michał Zakrzewski	
	Prowadzący zajęcia z przedmiotu		

Formy zajęć							
	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	0.0	0.0	30.0	0.0	0.0	30
	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	30	0.0	70.0	100

Cel przedmiotu	Rozwinięcie praktycznych umiejętności niezbędnych do efektywnej pracy w środowisku systemów Unix/ Linux. Zostaną przekazane techniki zarządzania plikami i katalogami za pomocą terminala, a także podstawowe operacje systemowe. Moduł poświęcony pisaniu skryptów w powłoce Linux umożliwi automatyzację często powtarzalnych zadań. W ramach przedmiotu omówione zostaną także umiejętności związane z systemem kontroli wersji git, w tym tworzenie i zarządzanie gałęziami, comitowanie zmian oraz pobieranie aktualizacji z repozytorium zdalnego.

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[INFPL3_U03] potrafi współdziałać z innymi osobami w ramach prac zespołowych, zarządzać swoim czasem oraz podejmować zobowiązania, porozumiewać się przy użyciu różnych technik w środowisku zawodowym w tym z wykorzystaniem dedykowanych narzędzi; umie przedstawiać różne opinie i alternatywne rozwiązania techniczne w zespole projektowym, wyjaśniając ich podstawy, konsekwencje oraz wpływ na realizację projektu	Student potrafi korzystać z podstawowych poleceń powłoki Linux Student potrafi napisać w edytorze i uruchomić prosty skrypt w Bash'u Student umie utworzyć repozytorium i nim zarządzać Student potrafi pracować w zespole informatyków, zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminy, porozumiewać się przy użyciu różnych technik w środowisku zawodowym w tym z wykorzystaniem dedykowanych narzędzi	[SU4] test/egzamin - ustny lub pisemny [SU5] realizacja zadania problemowego
	[INFPL3_K02] jest gotów do uznawania znaczenia wiedzy w rozwiązywaniu problemów poznawczych i praktycznych oraz zasięgania opinii ekspertów w przypadku trudności z samodzielnym rozwiązywaniem problemu	Student potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania. Jest gotów do korzystania z dokumentacji technicznej, literatury fachowej oraz źródeł eksperckich	[SK4] test/egzamin - ustny lub pisemny [SK5] realizacja zadania problemowego
	[INFPL3_W04] zna i rozumie w zaawansowanym stopniu zagadnienia w zakresie programowania, algorytmów i złożoności, języków i paradygmatów programowania oraz złożone zależności między nimi	Student zna pojęcia związane z podstawowymi komendami terminalu Linux. Student zna zasady tworzenia i wykonywania skryptów w Bash. Student zna podstawy obsługi systemu kontroli wersji git, takie jak tworzenie i usuwanie repozytorium, dodawanie, usuwanie i commitowanie zmian oraz rozwiązywanie konfliktów. Student zna pojęcie wyrażenia regularnego i ich składnię oraz umiejętność ich wykorzystywania do przetwarzania plików tekstowych.	[SW4] test/egzamin - ustny lub pisemny [SW5] realizacja zadania problemowego
	[INFPL3_W08] zna i rozumie w zaawansowanym stopniu fakty oraz metody w zakresie wykorzystania narzędzi i środowisk wytwarzania, testowania i utrzymania oprogramowania; stosuje tę wiedzę do tworzenia wydajnych, skalowalnych i bezpiecznych aplikacji	Student zna pojęcia związane z podstawowymi komendami terminalu Linux Student zna zasady tworzenia i wykonywania skryptów w Bash. Student zna podstawy obsługi systemu kontroli wersji git, takie jak tworzenie i usuwanie repozytorium, dodawanie, usuwanie i commitowanie zmian oraz rozwiązywanie konfliktów. Student zna pojęcie wyrażenia regularnego i ich składnię oraz umiejętność ich wykorzystywania do przetwarzania plików tekstowych.	[SW4] test/egzamin - ustny lub pisemny [SW5] realizacja zadania problemowego

<table>
<tr><td>Treści przedmiotu</td><td>1.
Wprowadzenie do systemu operacyjnego Linux: historia systemu operacyjnego Linux oraz podstawowe komendy w terminalu Linux

2.
Obsługa terminala w Linuxie: praca z konsolą, zarządzanie plikami i katalogami, przekierowanie strumieni wejścia/wyjścia, tworzenie i edycja plików w terminalu

3.
Podstawowa obsługa polecenia git: wprowadzenie do systemu kontroli wersji git, tworzenie repozytorium git,podstawowe operacje na repozytorium git (dodawanie, usuwanie, klonowanie), commitowanie zmian w repozytorium git, rozwiązywanie konfliktów w repozytorium git

4.
Wyrażenia regularne: wprowadzenie do wyrażeń regularnych, składnia i zasady pisania wyrażeń regularnych, użycie wyrażeń regularnych w terminalu Linux, zastosowanie wyrażeń regularnych

5.
Operacje na plikach tekstowych: przetwarzanie plików tekstowych w terminalu Linux, filtrowanie zawartości plików tekstowych, sortowanie i łączenie plików tekstowych

6.
Pisanie skryptów w Bash: wprowadzenie do Bash, zmienne i stałe w Bash, pętle i warunki w Bash, funkcje w Bash oraz interakcja z użytkownikiem w Bash

7.
Gałęzie w systemie kontroli wersji git: wprowadzenie do gałęzi w systemie kontroli wersji git, tworzenie i usuwanie gałęzi w repozytorium git, praca na gałęziach w repozytorium git, łączenie gałęzi w repozytorium git (merge), rozwiązywanie konfliktów połączenia gałęzi w repozytorium git,pobieranie i aktualizacja zmian z repozytorium git</td></tr>
<tr><td>Wymagania wstępne i dodatkowe</td><td>Brak wymagań wstępnych</td></tr>
<tr><td>Sposoby i kryteria oceniania osiąganych efektów uczenia się</td><td>
<table>
<tr><th>Sposób oceniania (składowe)</th><th>Próg zaliczeniowy</th><th>Składowa oceny końcowej</th></tr>
<tr><td>Sprawdziany praktyczne z pytaniami otwartymi</td><td>51.0%</td><td>40.0%</td></tr>
<tr><td>Kolokwium teoretyczne z pytaniami zamkniętymi</td><td>51.0%</td><td>60.0%</td></tr>
</table>
</td></tr>
<tr><td>Zalecana lista lektur</td><td>
<table>
<tr><td>Podstawowa lista lektur</td><td>• S. Chacon, B. Straub, Pro Git, Wydawnictwo Apress, 2014.
• W. E. Shotts, The Linux Command Line: A Complete Introduction, 2nd ed. San Francisco, CA: No Starch Press, 2019.
• Ł. Sosna, Linux. Komendy i polecenia, Wydawnictwo Helion, 2023.
• M. G. Sobell, Linux. Programowanie w powłoce. Praktyczny przewodnik, Wydawnictwo Helion, 2013.</td></tr>
<tr><td>Uzupełniająca lista lektur</td><td>• D. J. Barrett, Efficient Linux at the Command Line: Boost Your Command-Line Skills, O'Reilly Media, 2022.
• R. Blum and C. Bresnahan, Linux Command Line and Shell Scripting Bible, 3rd ed. Hoboken, NJ: John Wiley & Sons, 2015.</td></tr>
<tr><td>Adresy eZasobów</td><td></td></tr>
</table>
</td></tr>
<tr><td>Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania</td><td>Brak</td></tr>
<tr><td>Praktyki zawodowe w ramach przedmiotu</td><td>Nie dotyczy</td></tr>
</table>

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.