



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Techniki emulacji

Przedmiot

Kierunek studiów

Informatyka

Studia w zakresie (specjalność)

Przetwarzanie brzegowe

Poziom studiów

drugiego stopnia

Forma studiów

stacjonarne

Rok/semestr

2/3

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

15

Ćwiczenia

Laboratoria

15

Projekty/seminaria

Inne (np. online)

Liczba punktów ECTS

2

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Mariusz Naumowicz

email: mariusz.naumowicz@put.poznan.pl

tel. +48 61 665-2364

Wydział Informatyki i Telekomunikacji

ul. Piotrowo 3 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

Wymagania wstępne

Student rozpoczynający przedmiot powinien posiadać podstawową wiedzę z zakresu systemów operacyjnych i elektroniki. Powinien również rozumieć konieczność poszerzania swoich kompetencji oraz mieć gotowość do podjęcia współpracy w ramach zespołu.

Cel przedmiotu

- Przekazanie studentom wiedzy związanej z nowoczesnymi technikami emulacji systemów



wbudowanych.

- Zapoznanie studentów z nowoczesnymi metodami projektowania, testowania i prototypowania systemów wbudowanych z wykorzystaniem emulatorów.
- Rozwijanie u studentów umiejętności rozwiązywania złożonych problemów projektowych w zakresie budowania i testowania systemów wbudowanych.
- Kształtowanie u studentów umiejętności pracy zespołowej.

Przedmiotowe efekty uczenia się

Wiedza

1. Ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną związaną z kluczowymi zagadnieniami z zakresu informatyki - [K2st_W2]
2. Ma zaawansowaną wiedzę szczegółową dotyczącą wybranych zagadnień z zakresu informatyki - [K2st_W3]
3. Ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach informatyki i innych, wybranych, pokrewnych dyscyplin naukowych - [K2st_W4]
4. Ma zaawansowaną i szczegółową wiedzę o procesach zachodzących w cyklu życia systemów informatycznych sprzętowych lub programowych - [K2st_W5]

Umiejętności

1. Potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku polskim i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie - [K2st_U1]
2. Potrafi planować i przeprowadzać eksperymenty, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski oraz formułować i weryfikować hipotezy związane ze złożonymi problemami inżynierskimi i prostymi problemami badawczymi - [K2st_U3]
3. Potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne - [K2st_U4]
4. Potrafi — przy formułowaniu i rozwiązywaniu zadań inżynierskich — integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne - [K2st_U5]
5. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K2st_U6]

Kompetencje społeczne

1. Rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K2st_K1]
2. Rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu informatyki w rozwiązywaniu problemów badawczych i praktycznych - [K2st_K2]



Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- a) w zakresie wykładów: na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,
- b) w zakresie laboratoriów: na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

- a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez egzamin (test w postaci elektronicznej na platformie Moodle);
- b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez sprawdzian projektowy i ocenę zadań realizowanych w ramach każdego spotkania laboratoryjnego;

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

- omówienia dodatkowych aspektów zagadnienia,
- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,
- umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium.

Treści programowe

Program wykładu obejmuje następujące zagadnienia:

Programowe, sprzętowe, programowo-sprzętowe techniki emulacji. Nowoczesne narzędzia emulacji i ich możliwości. Zastosowanie emulatorów w przemyśle. Techniki optymalizacji emulatorów. Testowanie systemów wbudowanych z wykorzystaniem emulatora. Przystosowanie emulatorów do własnych potrzeb.

Zajęcia laboratoryjne prowadzone są w formie 2-godzinnych spotkań, odbywających się w laboratorium, poprzedzonych sesją instruktazową na początku semestru. Ćwiczenia realizowane są przez 2-osobowe zespoły studentów. Program laboratorium obejmuje następujące zagadnienia:

Przygotowanie środowiska programistycznego niezbędnego do uruchomienia wybranych emulatorów. Budowa własnego systemu wbudowanego w oparciu o istniejącą strukturę w emulatorze. Przygotowanie systemu operacyjnego pod emulowany system wbudowany. Analiza sygnałów systemu wbudowanego przy pomocy emulatora. Emulacja złożonego systemu informatycznego w oparciu o kilka urządzeń IoT. Automatyczne testowanie z wykorzystaniem emulatora.

Część wymienionych wyżej treści programowych realizowana jest w ramach pracy własnej studenta.

Metody dydaktyczne



1. wykład: prezentacja multimedialna uzupełniona przykładami podawanymi na tablicy.
2. ćwiczenia laboratoryjne: ćwiczenia praktyczne, dyskusja, praca w zespole, zawody projektowe.

Literatura

Podstawowa

1. John L. Hennessy, John L. Hennessy, Computer Architecture: A Quantitative Approach, 4th Edition. Elsevier, 2007. ISBN: 0123704901.
2. Eldad Eilam, Reversing: Secrets of Reverse Engineering, Wiley, 2011. ISBN: 0764574817.

Uzupełniająca

1. Noam Nisan, Shimon Schocken, The Elements of Computing Systems: Building a Modern Computer from First Principles, The MIT Press, 2005. ISBN: 0262640686.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	50	2,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	30	1,5
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych, przygotowanie do sprawdzianu/egzaminu, wykonanie projektu) ¹	20	0,5

¹ niepotrzebne skreślić lub dopisać inne czynności