

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Inżynieria oprogramowania dla Syst. Wbudowanych i Mobilnych		Kod 1010542321010510037
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 1 / 2
Ścieżka obieralności/specjalność Mikrosystemy informatyczne	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: II stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 15 Ćwiczenia: - Laboratoria: - Projekty/seminaria: -		Liczba punktów 1
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy		(ogólnouniversytecki, z innego kierunku) z danego kierunku
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne		Podział ECTS (liczba i %) 1 100% 1 100%
Odpowiedzialny za przedmiot / wykładowca: dr inż. Szymon Szczesny email: szymon.szczesny@put.poznan.pl tel. 616652297 Informatyki Piotrowo 3A, 60-965 Poznań dr inż. Mariusz Naumowicz email: mariusz.naumowicz@put.poznan.pl tel. 616652297 Informatyki Piotrowo 3A, 60-965 Poznań		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę w zakresie architektury systemów wbudowanych, strategii modelowania i analizy mikrosystemów zarówno w obszarze warstwy fizycznej jak i warstwy oprogramowania oraz znać wykorzystywane w mikrosystemach protokoły transmisji przewodowej i bezprzewodowej.
2	Umiejętności:	Wymagana jest również umiejętność programowania oraz znajomość języków opisu sprzętu. Student powinien również posiadać elementarną wiedzę na temat technologii rekonfigurowalnych, np. na przykład obwodów programowalnej logiki typu FPGA oraz na temat układów programowalnych typu mikrokontrolery.
3	Kompetencje społeczne	Ponadto powinien prezentować postawę zainteresowania pogłębianiem posiadanej wiedzy w obszarze systemów wbudowanych, ze szczególnym uwzględnieniem metodologii tworzenia oprogramowania dla urządzeń programowalnych, hybrydowych i mobilnych.
Cel przedmiotu: 1. Poszerzenie wiedzy teoretycznej w zakresie metodologii rozwijania oprogramowania dla systemów wbudowanych, przekazanie informacji o głównych problemach dziedziny 2. Zaznajomienie z literaturą poświęconą projektowaniu, rozwijaniu i użytkowaniu mikrosystemów 3. Przekazanie wiedzy w zakresie metod i strategii projektowania i modelowania oprogramowania dla systemów wbudowanych wykorzystywanymi w przemysłowych rozwiązaniach 4. Przedstawienie różnic między tworzeniem oprogramowania dla systemów komputerowych a systemów wbudowanych z omówieniem poszczególnych klasy systemów 5. Pokazanie znaczenia obranych metod w zakresie rozwijania i testowania oprogramowania, scharakteryzowanie narzędzi wykorzystywanych w procesie rozwijania oprogramowania dla mikrosystemów 6. Zapoznanie z ograniczeniami w rozwoju oprogramowania, z zagadnieniami dotyczącymi bezpieczeństwa oraz regułami użytkowania licencji systemów wbudowanych 7. Omówienie specyfiki rynku systemów wbudowanych, głównych trendów jego rozwoju, a także scharakteryzowanie rynku pracy we wskazanej tematyce w skali kraju i świata. 8. Uświadomienie ograniczeń prawnych, ekonomicznych i technicznych w rozwoju dziedziny		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		

1. Absolwent zna podstawy teoretyczne projektowania i rozwijania dużych systemów informatycznych - [K2st_W1]
2. Absolwent zna techniki rozwijania oprogramowania dla branży systemów wbudowanych - [K2st_W2]
3. Absolwent zna i rozumie zagadnienia rozwijania systemów interdyscyplinarnych - [K2st_W4]
4. Absolwent zna cykle życia oprogramowania oraz narzędzia wspomagające jego rozwijanie - [K2st_W5]
5. Absolwent zna techniki modelowania złożonych systemów informatycznych - [K2st_W6]
Umiejętności:
1. Absolwent potrafi formułować wymagania funkcjonalne i нефункционалне systemów informatycznych oraz jest świadomy problemów komunikacyjnych pomiędzy zespołami ekspertów a zespołem deweloperów oprogramowania - [K2st_U2]
2. Absolwent potrafi integrować wiedzę dotyczącą warstwy sprzętowej i warstwy oprogramowania w celu realizacji systemu dedykowanego również dla pozainformatycznych zastosowań - [K2st_U5]
3. Absolwent potrafi wykorzystywać metody oceny pracochłonności projektu oraz zarządzać ryzykiem przy realizacji projektu informatycznego - [K2st_U7]
4. Absolwent potrafi wybrać narzędzia wspierające rozwijanie projektów dla branży systemów wbudowanych - [K2st_U9]
5. Absolwent potrafi ocenić złożoność zadań przewidzianych przy rozwijaniu oprogramowania dla branży systemów wbudowanych, w tym również projektów interdyscyplinarnych, cechujących się potencjałem badawczym - [K2st_U10]
Kompetencje społeczne:
1. Absolwent jest świadomy konieczności ciągłego poszerzania wiedzy, zwłaszcza w zakresie narzędzi i technik rozwijania oprogramowania - [K2st_K1]
2. Absolwent jest gotów na zdobywanie wiedzy niezbędnej przy realizacji projektów informatycznych, kompatybilnych z aktualnie obowiązującymi standardami rozwijania oprogramowania i komunikowania jego modułów - [K2st_K2]

Sposoby sprawdzenia efektów kształcenia

Ocena formująca:

- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,
- na podstawie prezentacji rozwiązania problemów zadanych na poprzednich wykładach

Ocena podsumowująca:

Weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę wiedzy i umiejętności wykazanych na sprawdzianie pisemnym o charakterze problemowym
- omówienie wyników sprawdzianu zaliczeniowego,

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

- omówienia dodatkowych aspektów zagadnienia,
- uwagi związane z udoskonaleniem materiałów dydaktycznych,
- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.

Treści programowe

Program wykładu obejmuje omówienie następujących zagadnień:

Różne klasy systemów wbudowanych i porównanie ich funkcjonalności do systemów komputerowych; wymagania dla systemów wbudowanych pod kątem efektywnego wykorzystania zasobów sprzętowych oraz wyboru właściwego rozwiązania technologicznego dyktowanego funkcjonalnymi i ekonomicznymi aspektami; metody projektowania, modelowania i analizy oprogramowania; osadzanie aplikacji w istniejącym systemie zależnie od jego wersji, wraz z omówieniem specyfiki pracy z systemami uniksowymi i systemem eCos; zaznajomienie z analizą BOM (Bill Of Materials); przedstawienie scenariuszy testowania i metod automatyzacji testów; omówienie narzędzi wykorzystywanych w procesie rozwijania oprogramowania: gcc, qemu, gdb, JTAG oraz narzędzi kontroli wersji: GIT, SVN; przedstawienie istniejących emulatorów platform sprzętowych i strategii rozwoju wielkosystemowych projektów; omówienie metod budowania interfejsu użytkownika wraz z opracowaniem mechanizmu interakcji z użytkownikiem i otoczeniem (UX ? User eXperience); uwzględnienie w procesie projektowania oprogramowania warunków środowiskowych (warunków klimatycznych, mechanicznych, czasu pracy); wyszczególnienie zasad tworzenia dokumentacji na różnych poziomach rozwoju oprogramowania (na etapie tworzenia specyfikacji, opracowywania architektury, implementacji, rekonfiguracji, testowania, serwisowania); porównanie rozwiązań komercyjnych i Open Source; omówienie zagrożeń i zasad bezpieczeństwa; przedstawienie różnic w obszarze mikrosystemów mikroprocesorowych, mikrosystemów rekonfigurowalnych, hybrydowych; nakreślenie specyfiki rynku systemów wbudowanych i trendów jego rozwoju, a także rynku pracy w tematyce systemów wbudowanych ? zarówno w skali kraju, jak i w skali świata; omówienie ograniczeń prawnych, ekonomicznych i technicznych w rozwoju dziedziny; podsumowanie głównych wyzwań stojących na drodze rozwoju systemów wbudowanych i mikrosystemów; omówienie efektywnego wykorzystania języków programowania i skryptowych typowych dla systemów wbudowanych (C/C++, bash, Python); podanie przykładowych mechanizmów rozbudowy systemów wbudowanych, szczególnie pod kątem tworzenia sterowników dla urządzeń peryferyjnych oraz całego procesu przygotowania portu istniejącego systemu na nowe platformy sprzętowe.

Literatura podstawowa:

1. UML: przewodnik użytkownika, Grady Booch, James Rumbaugh, Ivar Jacobson, WTN 2001
2. The industrial electronics handbook Wilanowski B, Irwin D., Taylor & Francis, 2011

Literatura uzupełniająca:

1. Real-Time Systems Design and Analysis: Tools for the Practitioner, P. A. Laplante, S. J. Ovaska, Wiley, 2012

Bilans nakładu pracy przeciętnego studenta

Czynność	Czas (godz.)
1. Udział w wykładach	15
2. Udział w konsultacjach związanych z realizacją procesu kształcenia	2
3. Zapoznanie się ze wskazaną literaturą	2
4. Przygotowanie do zaliczenia wykładów	6

Obciążenie pracą studenta

forma aktywności	godzin	ECTS
Łączny nakład pracy	25	1
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	17	1
Zajęcia o charakterze praktycznym	0	0