

Karta (sylabus) modułu/przedmiotu
Mechatronika
 Studia pierwszego stopnia

Przedmiot:	Mechatroniczne systemy diagnostyczne
Rodzaj przedmiotu:	Obieralny
Kod przedmiotu:	MT 1 S 0 7 47-2_1
Rok:	4
Semestr:	7
Forma studiów:	Studia stacjonarne
Rodzaj zajęć i liczba godzin w semestrze:	60
Wykład	30
Ćwiczenia	-
Laboratorium	30
Projekt	-
Liczba punktów ECTS:	4
Sposób zaliczenia:	Zaliczenie
Język wykładowy:	Język polski

Cel przedmiotu	
C1	Zapoznanie z ze środowiskiem programowania aplikacji realizujących funkcje systemów pomiarowo-diagnostycznych dla układów mechatronicznych.
C2	Przedstawienie struktur przetwarzania danych pomiarowych oraz przetwarzania danych.
C3	Praktyczne przygotowanie do tworzenia aplikacji diagnostyczno-pomiarowych.
C4	Nabywanie umiejętności dopasowania systemu do warunków pracy aplikacji i wymagań użytkownika.

Wymagania wstępne w zakresie wiedzy, umiejętności i innych kompetencji	
1	Umiejętność podstawowej obsługi systemu operacyjnego MS Windows w zakresie przeszukiwania zasobów lokalnych (dyski), sieciowych i internetowych zdobywana w zakresie takich przedmiotów jak Informatyka, Technologia informacyjna.
2	Wiadomości podstawowe z zakresu miernictwa zdobywana w zakresie takich przedmiotów jak np. Metrologia wielkości elektrycznych.

Efekty kształcenia	
	W zakresie wiedzy:
EK 1	Student identyfikuje zróżnicowane typy danych, sposób wyróżniania ich w kodzie programu oraz wyjaśnia potencjalne problemy wynikające ze stosowania różnych typów zmiennych.
EK 2	Student rozróżnia techniki i metody odnoszące się do poprawnego wykorzystania struktur zarządzania kodem i wyjaśnia sposób minimalizowania wymagań tworzonej aplikacji w stosunku do systemu operacyjnego, platformy sprzętowej oraz spełniania wymagań użytkownika.
	W zakresie umiejętności:
EK 3	Student konstruuje proste jednozadaniowe programy, bardziej skomplikowane systemy złożone, tworzy pliki wykonywalne i instalacyjne

	systemu.
EK 4	Student analizuje wykonywanie kodu programu, identyfikuje występujące błędy i ocenia poprawność stosowania wybranej architektury programu.
EK 5	Student konstruuje aplikacje umożliwiające zdalne i grupowe użytkowanie systemu.
	W zakresie kompetencji społecznych:
EK 6	Student pracując w zespole określa wymagania wzajemnie współpracujących programów.
EK 7	Student jest odpowiedzialny za minimalizowanie wymagań tworzonych aplikacji w stosunku do systemu operacyjnego i platformy sprzętowej tak, aby system pracował jak najwydajniej.

Treści programowe przedmiotu	
Forma zajęć – wykłady	
	Treści programowe
W1	System diagnostyczny jako rozbudowany system pomiarowo-kontrolny. Charakterystyka środowiska programistycznego - instalacja, panel czołowy, schemat blokowy, palety narzędzi.
W2	Typy danych - występowanie, rozpoznawanie na podstawie symboli i kolorystyki obiektów, zmiana typu danych. Analiza/usuwanie błędów. Metody śledzenia kodu.
W3	Modularyzacja - tworzenie, wstawianie, wywoływanie podprogramów.
W4	Pętle while i for - zastosowanie, sposób obsługi, tunele danych pętli.
W5	Zmienne binarne. Funkcjonowanie przełączników logicznych (mechanical action). Rejestr przesuwany - obsługa obiektu w kodzie programu, zalecane ustawienia.
W6	Prezentacja wyników - wskaźniki graficzne (obiekty wykresów waveform chart, waveform graph, XY graph, intensity plot).
W7	Macierze /funkcje macierzowe. Klastry /funkcje klastrowe. Rozgałęzianie kodu struktura wyboru. Wymuszanie kolejności wykonywania kodu – struktura sekwencyjna.
W8	Dane łańcuchowe - wprowadzanie i wyświetlanie danych tekstowych. Operacje plikowe we/wy z wykorzystaniem zaawansowanych i prostych węzłów środowiska.
W9	Indywidualizacja aplikacji i środowiska programistycznego: zasady projektowania panelu czołowego, wprowadzanie klawiszy skrótów.
W10	Przyspieszanie pracy programisty dzięki indywidualizacji środowiska programistycznego - modyfikacja właściwości programów, palet, tworzenie własnych obiektów.
W11	Planowanie aplikacji – dobór struktury kodu, projektowanie i wdrażanie mechanizmów obsługi błędów, unikanie nadmiernego wykorzystania procesora i pamięci.
W12	Projektowanie panelu czołowego – zagadnienia podstawowe, klastry logiczne, programowa obsługa obiektów za pomocą węzłów właściwości.
W13	Zdalne sterowanie panelem. Udostępnianie danych za pomocą wbudowanego serwera WWW. Konfiguracja klienta zdalnego dostępu do aplikacji.
W14	Techniki zarządzania danymi w zakresie jednego programu, wymiany danych w zakresie pojedynczej jednostki, sieciowa wymiana danych – zmienne lokalne i globalne, protokół datasocket.

W15	Profilowanie aplikacji. Tworzenie plików wykonywalnych. Generowanie pakietów instalatora.
------------	---

Forma zajęć – laboratoria	
	Treści programowe
L1	Zajęcia wstępne. Przedstawienie sposobu pracy i zasad obowiązujących w laboratorium. Utworzenie i przetestowanie kont użytkowników. Indywidualne kształtowanie środowiska pracy przez Studenta. Sprawdzenie dostępności zasobów sieciowych.
L2	Zapoznanie ze środowiskiem tworzenia systemów przez stworzenie przyrządu wirtualnego do generowania sygnału i jego prezentacji na panelu czołowym. Korzystanie z gotowych szablonów.
L3	Edycja elementów panelu czołowego. Korzystanie z węzłów typu Express VI. Ćwiczenie technik usuwania błędów z programu.
L4	Pętla While (sposób funkcjonowania, sposób przekazywania danych przez tunele pętli). Rejestr przesuwany (Shift register). Prezentacja danych za pomocą wykresu Waveform Chart. Stosowanie pętli For.
L5	Tablice tworzenie tablic oraz zapoznanie z funkcjami działania na tablicach. Korzystanie z wykresów XY (XY graph). Zapoznanie z korzystaniem z wykresów natężenia (intensity plot). Klastry - tworzenie obiektów klastrów na panelu czołowym oraz korzystanie z funkcji do łączenia i rozłączania danych o charakterze klastrowym.
L6	Zapoznanie z wykorzystaniem struktur wyboru (case structure). Struktura sekwencyjna (sequence structure) – przykładowe zastosowanie. Budowa przyrządu wirtualnego wykorzystującego węzły formuły do wykonywania złożonych działań matematycznych i wyświetlania ich na wykresie.
L7	Odrabianie zajęć / Wyrównywanie zaległości.
L8	Zmienne łańcuchowe - poznanie funkcji: formatowania do postaci łańcuchowej (Format Into String), łączenia łańcuchów (Concatenate Strings) i określania długości łańcuchów (String Length). Zapoznanie z mechanizmem obsługi plików z danymi (zapis i odczyt z pliku, zapisywanie tablicy dwuwymiarowej (2D) do pliku tekstowego w postaci arkusza danych).
L9	Deklaracja sposobu funkcjonowania podprogramów. Deklarowanie klawiszy skrótu dla funkcji panelu czołowego i konfigurowanie sposobu wyświetlania okien podprogramów inicjowanych za pomocą klawiszy skrótu. Obsługa klastrów za pomocą klawiszy skrótu. Zapoznanie z metodą edycji gotowych programów o konfiguracji utrudniającej modyfikację schematu blokowego.
L10	Program (instrument wirtualny) generujący, analizujący i wyświetlający serie danych, wykorzystujący standardowy mechanizm obsługi błędów. Utworzenie programu kontrolującego dane o użytkowniku bazującego na prostym modelu architektury. Zapoznanie z obsługą szablonów dostarczanych ze środowiskiem LabVIEW oraz obsługą szablonów tworzonych samodzielnie.
L11	Konfiguracja (optymalizacja) panelu czołowego. Stosowanie kontrolki zakładkowej (tab control). Menu bazujące na klastrze logicznym. Węzły właściwości.
L12	Wykorzystanie zmiennych lokalnych do inicjacji, modyfikowania wskaźników i kontrolki panelu czołowego programu. Używanie zmiennych globalnych do wymiany danych pomiędzy programami. Wymiana danych za pomocą mechanizmu datasocket.

L13	Zapis i odczyt danych z plików binarnych. Przeglądanie i sterowanie programem ze zdalnego komputera z zainstalowanym środowiskiem LabVIEW. Zdalna obsługa programów za pośrednictwem protokołu HTTP i przeglądarki internetowej.
L14	Łączenie podprogramów ramach projektu. Zapoznanie się z wbudowanymi funkcjami środowiska LabVIEW ułatwiającymi obsługę projektów aplikacji. Tworzenie wykonywalnego pliku samodzielnej aplikacji - Application Builder.
L15	Odrabianie zajęć / Wyrównywanie zaległości. Wystawianie ocen.

Metody dydaktyczne	
1	Wykład z prezentacją multimedialną.
2	Dyskusja w trakcie zajęć wykładowych.
3	Laboratorium programistyczne.
4	Praca grupowa

Obciążenie pracą studenta	
Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny kontaktowe z wykładowcą, w tym:	60
Udział w wykładach	30
Udział w laboratoriach	30
Praca własna studenta, w tym:	40
Przygotowanie do laboratoriów	20
Samodzielne przygotowanie do zaliczenia wykładu	20
Łączny czas pracy studenta	100
Sumaryczna liczba punktów ECTS dla przedmiotu:	4
Liczba punktów ECTS w ramach zajęć o charakterze praktycznym (ćwiczenia, laboratoria, projekty)	2

Literatura podstawowa	
1	Chruściel M.: LabVIEW w praktyce, Wyd. BTC, Legionowo 2008.
2	Tłaczała W.: Środowisko LabVIEW w eksperymencie wspomaganym komputerowo, WNT, W-wa 2002.
Literatura uzupełniająca	
3	Materiały szkoleniowe firmy National Instruments LabVIEW Express Basics Interactive Training. CD. National Instruments 2008.
4	Dokumentacja firmy National Instruments: G Programming Reference Manual, BridgeVIEW and LabVIEW. National Instruments 2001.

Macierz efektów kształcenia					
Efekt kształcenia	Odniesienie danego efektu kształcenia do efektów zdefiniowanych dla całego	Cele przedmiotu	Treści programowe	Metody dydaktyczne	Metody oceny

	programu (PEK)				
EK 1	MT1A_W13+ MT1A_W15+ MT1A_W17+ MT1A_W21+	C1	W1, W2, W5, W8, L1, L2, L5, L8	1, 2, 3	O1, O3
EK 2	MT1A_W13+ MT1A_W15+ MT1A_W17+ MT1A_W21+	C1, C2	W2, W10, W11, W15, L3, L4, L11	1, 2, 3	O1, O3
EK 3	MT1A_W15+ MT1A_U15+ MT1A_U19++	C2, C3	W3, W4, W6, W7, L4, L5, L6, L10, L11	1, 2, 3	O1, O3
EK 4	MT1A_U02+ MT1A_U03+ MT1A_W15+ MT1A_U19++ MT1A_U21+	C3	W2, W9, W10, W15, L5, L6, L7, L9, L10, L11	1, 2, 3	O1, O3
EK 5	MT1A_W15+ MT1A_U19++ MT1A_U21+	C3, C4	W11, W12, W13, W14, L12, L13, L14	1, 2, 3	O1, O3
EK 6	MT1A_U21+ MT1A_K03+ MT1A_K01+	C4	W13, W14, W15, L7, 12, L13, L14	1, 2, 3, 4	O1, O2, O3
EK 7	MT1A_U19++ MT1A_K02+ MT1A_K04+	C3, C4	W11, W15, L11, L13, L15	1, 2, 3, 4	O1, O2, O3

Metody i kryteria oceny		
Symbol metody oceny	Opis metody oceny	Próg zaliczeniowy
O1	Kolokwium zaliczeniowe	51%
O2	Frekwencja na zajęciach laboratoryjnych w wyznaczonych terminach (systematyczność)	80%
O3	Wykonywanie zadań instrukcji laboratoryjnych połączona z uzyskiwaniem plików poprawnie działających aplikacji	100%

Autor programu:	dr inż. Andrzej Sumorek
Adres e-mail:	a.sumorek@pollub.pl
Jednostka organizacyjna:	Katedra Inżynierii Komputerowej i Elektrycznej, Wydział Elektrotechniki i Informatyki