

Języki programowania - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Języki programowania
Kod przedmiotu	11.3-WE-EP-JP
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Elektrotechnika
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2017/2018

Informacje o przedmiocie	
Semestr	2
Liczba punktów ECTS do zdobycia	3
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	dr inż. Grzegorz Łabiak

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Laboratorium	30	2	18	1,2	Zaliczenie na ocenę
Wykład	15	1	9	0,6	Zaliczenie na ocenę

Cel przedmiotu

Celem przedmiotu jest nauka programowania w języku C/C++. Nauka polega w pierwszej kolejności na zapoznaniu się ze składnią i semantyką języka C i wybranych elementów języka C++ oraz na kształceniu umiejętności implementowaniu prostych algorytmów. Dodatkowo studenci zaznajamiani są z podstawowymi elementami programowania obiektowego, które pozwalają definiowanie obiektów programistycznych przydatnych w zakresie projektowania prostych obwodów elektrycznych.

Wymagania wstępne

Zagadnienia z matematyki z zakresu szkoły średniej.

Zakres tematyczny

- Wiadomości wstępne: program i jego składowe.
- Środowisko programistyczne. Pliki źródłowe. Kompilacja. Podstawowe elementy i budowa programu. Funkcja główna. Funkcje i procedury.
- Typy podstawowe. Zmienne. Operacje wejścia/wyjścia. Operatory.
- Pętle iteracyjne: for, while, do-while (repeat-unitl).
- Instrukcje decyzyjne.
- Tworzenie własnych funkcji.
- Tablice i łańcuchy znaków.
- Wskaźniki.
- Algorytmy sortowania: bąbelkowe, wybór prosty, qsort.
- Struktury listowe: tworzenie, dodawanie, usuwanie, sortowanie, wyświetlanie.
- Klasy, obiekty, dziedziczenie, przeciążanie funkcji operatorowych.

Metody kształcenia

- wykład:** wykład konwencjonalny, wykład problemowy, dyskusja
- laboratorium:** ćwiczenia laboratoryjne

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Potrafi opracowane rozwiązanie problemu algorytmicznego zaimplementować w języku C, uruchomić i testować powstały program komputerowy.	K_U11	- bieżąca kontrola na zajęciach - sprawdzian	Laboratorium
Student potrafi formułować problemy świata rzeczywistego jako typowe problemy algorytmiczne, dokonywać ich analizy pod kątem ich rozwiązania i oceny złożoności obliczeniowej.	K_U11	- bieżąca kontrola na zajęciach - sprawdzian	Laboratorium

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Podstawową nabytą umiejętnością jest umiejętność implementacji typowych algorytmów (np. wyszukiwania, problemy grafowe itp.) w języku C.	K_U11	kolokwium	Wykład
Rozumie różnicę pomiędzy programowaniem obiektowym i strukturalnym	K_W09	bieżąca kontrola na zajęciach	Laboratorium

Warunki zaliczenia

Wykład - egzamin - umiejętność rozwiązywania zadań; znajomość składni i semantyki języka C/C++

Laboratorium - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich ćwiczeń laboratoryjnych, przewidzianych do realizacji w ramach programu laboratorium

Składowe oceny końcowej = wykład: 50% + laboratorium: 50%

Literatura podstawowa

1. Kernighan B. W., Ritchie D. M.: Język Ansi C, WNT, Warszawa, 1994.
2. Sielicki A.: Laboratorium programowania w języku Pascal, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1994.
3. Stroustrup B.: Język C++, WNT 2002

Literatura uzupełniająca

1. Aho A. V., Hopcroft J. E., Ullman J. D.: Projektowanie i analiza algorytmów, Helion, Warszawa, 2003.
2. Banachowski L., Diks K., Rytter W.: Algorytmy i struktury danych, WNT Warszawa, 2001.
3. Roszkowski J.: Analiza i projektowanie strukturalne, Helion, Gliwice, 2002.
4. Wirth N.: Algorytmy + struktury danych = programy, WNT, Warszawa, 1989.

Uwagi

Zmodyfikowane przez dr hab. inż. Radosław Kłosiński, prof. UZ (ostatnia modyfikacja: 02-05-2017 18:31)

Wygenerowano automatycznie z systemu SylabUZ