KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu
Programowanie systemowe i współbieżne [N1Inf1>PSW]

Przedmiot

Kierunek studiów
Informatyka

Rok/Semestr
2/3

Studia w zakresie (specjalność)
–

Profil studiów
ogólnoakademicki

Poziom studiów
pierwszego stopnia

Język oferowanego przedmiotu
polski

Forma studiów
niestacjonarne

Wymagalność
obligatoryjny

Liczba godzin

Wykład
20

Laboratorium
20

Inne (np. online)
0

Ćwiczenia
0

Projekty/seminaria
0

Liczba punktów ECTS

5,00

Koordynatorzy

dr inż. Dariusz Wawrzyniak
dariusz.wawrzyniak@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający ten moduł powinien posiadać podstawową wiedzę z zakresu funkcjonowania komputera i programowania imperatywnego. Powinien posiadać umiejętność rozwiązywania podstawowych problemów z zakresu implementacji i oceny kosztu działania prostych algorytmów oraz umiejętność pozyskiwania informacji ze wskazanych źródeł. Powinien również rozumieć konieczność poszerzania swoich kompetencji i wykazywać gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.

Cel przedmiotu

Celem przedmiotu jest:

- przekazanie studentom podstawowej wiedzy teoretycznej związanej z usługami jądra systemu operacyjnego i przetwarzaniem współbieżnym,
- zaznajomienie studentów z praktycznymi aspektami realizacji przetwarzania współbieżnego,
- rozwijanie u studentów umiejętności rozwiązywania problemów związanych z przetwarzaniem współbieżnym w systemach komputerowych.

Przedmiotowe efekty uczenia się

Wiedza
1. ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną w zakresie programowania systemowego i współbieżnego, oraz wiedzę szczegółową w zakresie zakresu funkcjonowania systemów operacyjnych
2. ma podstawową wiedzę o cyklu życia systemów operacyjnych, a w szczególności o zasadach zarządzania procesami, mechanizmach synchronizacji i przeciwdziałania zakleszczeniom
3. zna podstawowe techniki, metody oraz narzędzia wykorzystywane w procesie rozwiązywania zadań informatycznych, głównie o charakterze inżynierskim, z zakresu kluczowych zagadnień programowania systemowego i współbieżnego
Umiejętności
1. potrafi, formułując i rozwiązując zadania informatyczne, zastosować odpowiednio dobrane metody programowania systemowego i współbieżnego, w tym metody analityczne
2. potrafi ocenić złożoność obliczeniową algorytmów i problemów współbieżnych
3. potrafi - zgodnie z zadaną specyfikacją - zaprojektować (sformułować specyfikację funkcjonalną i wymagania pozafunkcjonalne dla wybranych charakterystyk jakościowych) oraz zrealizować szeroko rozumiany system informatyczny, dobierając język programowania odpowiedni do danego zadania programistycznego oraz używając właściwych metod, technik i narzędzi programowania współbieżnego
4. ma umiejętność formułowania algorytmów współbieżnych i ich implementacji z użyciem przynajmniej jednego z popularnych narzędzi
Kompetencje społeczne
1. rozumie, że w informatyce wiedza i umiejętności z zakresu programowania współbieżnego bardzo szybko stają się przestarzałe
2. ma świadomość znaczenia wiedzy z zakresu programowania systemowego i współbieżnego w rozwiązywaniu problemów inżynierskich oraz zna przykłady i rozumie przyczyny wadliwie działających systemów informatycznych, które doprowadziły do poważnych strat finansowych i społecznych

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:
1.
2. W zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
   -
   - odpowiedzi na pytania dotyczące materiału omawianego na wykładach.
   -
3.
4. W zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:
   -
   - ocenę przygotowania studenta do poszczególnych zajęć laboratoryjnych,
   -
   - ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych.
   -

Ocena podsumowująca:
1.
2. W zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
   -
   - ocenę wiedzy i umiejętności wykazanych na zaliczeniu o charakterze testowym, składającym się z kilku pytań/zadań otwartych lub zamkniętych, z możliwością uzyskania w sumie 100 punktów z progiem 50 punktów na ocenę pozytywną,
   -
   - omówienie wyników zaliczenia.
   -

3. W zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:
   -
   - ocenę wiedzy i umiejętności związanych z treściami przekazywanymi na laboratoriach poprzez kolokwium końcowe,
   -
   - zestawienie ocen wystawionych w trakcie semestru w postaci średniej.
   -

Aktywność podczas zajęć premiowana jest dodatkowymi punktami, w szczególności za:
-
- omówienie dodatkowych aspektów zagadnienia,
-
- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,
-
- uwagi prowadzące do udoskonalenia materiałów dydaktycznych lub procesu dydaktycznego.
-

Treści programowe

Program wykładu obejmuje następujące zagadnienia:
1.
2. Abstrakcja programowania współbieżnego: wprowadzenie pojęcia operacji atomowych i ich przeplotu.
3.
4. Problem wzajemnego wykluczania oraz jego rozwiązanie oparte na atomowych operacjach odczytu i zapisu współdzielonych zmiennych (np. algorytmy: Petersona, Lamporta), pojęcia bezpieczeństwa i żywotności.
5.
6. Złożone operacje atomowe, np.: test-and-set, exchange.
7.
8. Mechanizmy synchronizacji wspierane przez system operacyjny: semafory binarne i zliczające, mechanizmy standardu POSIX (zamki i zmienne warunkowe).
9.
10. Klasyczne problemy synchronizacji: producent-konsument, czytelnicy i pisarze, pięciu filozofów, śpiący fryzjerzy.
11.
12. Wsparcie językowe dla mechanizmów synchronizacji: warunkowe regiony krytyczne, monitory.
13.
14. Problem zakleszczenia: definicja zakleszczenia, warunki konieczne i dostateczne zakleszczenia, przeciwdziałanie zakleszczeniom (zapobieganie, unikanie, detekcja i likwidacja).
15.

Program zajęć laboratoryjnych obejmuje następujące zagadnienia na poziomie interfejsu usług systemowych w języku C:
1.
2. Dostęp do plików.
3.
4. Zarządzanie procesami.
5.
6. Obsługa sygnałów.
7.
8. Komunikacja międzyprocesowa i synchronizacja za pośrednictwem łączy.
9.
10. Komunikacja międzyprocesowa i synchronizacja za pośrednictwem mechanizmów IPC.
11.

Metody dydaktyczne

Wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, rozwiązywanie zadań.

Ćwiczenia laboratoryjne: zajęcia w laboratorium komputerowym polegające na implementacji programów (w tym współbieżnych) opartych na funkcjach jądra systemu operacyjnego, dyskusja.

Literatura

Podstawowa:

1.
2. Mordechai Ben-Ari, Podstawy programowania współbieżnego i rozproszonego, WNT, W-wa, 2009.
3.
4. A. Silberschatz, J.L. Peterson, G. Gagne, Podstawy systemów operacyjnych, WNT, W-wa, 2006.
5.

Uzupełniająca:

1.
2. Z. Weiss, T. Gruźlewski, Programowanie współbieżne i rozproszone w przykładach i zadaniach, WNT, W-wa, 1993.
3.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	42	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwiów/egzaminu, wykonanie projektu)	83	3,00