**Nazwa przedmiotu:**

Programowanie niskopoziomowe

**Koordynator przedmiotu:**

dr inż. Sławomir Paśko

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Automatyka Robotyka i Informatyka Przemysłowa

**Grupa przedmiotów:**

Obowiązkowe

**Kod przedmiotu:**

PNP

**Semestr nominalny:**

6 / rok ak. 2019/2020

**Liczba punktów ECTS:**

3

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

1) Liczba godzin bezpośrednich – 32 godz., w tym:
• wykład: 15 godz.
• projekt: 15 godz.
• konsultacje: 2 godz.
2) Praca własna studenta – 55 godz., w tym:
• przygotowanie do dwóch testów - 20 godz.
• przygotowanie do zajęć projektowych: 15 godz.
• samodzielne wykonanie projektu – 20 godz.
Razem: 87 godz. (3 ECTS)

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

1 punkt ECTS – 32 godz., w tym:
• wykład – 15 godz.
• projekt - 15 godz.
• konsultacje - 2 godz.

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

1 punkt ECTS – 35 godz., w tym:
• projekt - 15 godz.
• samodzielne wykonanie projektu – 20 godz.

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 15h |
| Ćwiczenia: | 0h |
| Laboratorium: | 0h |
| Projekt: | 15h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Wymagana biegła umiejętność programowania w języku C, C++ lub Java

**Limit liczby studentów:**

brak limitu

**Cel przedmiotu:**

Przygotowanie studenta do tworzenia oprogramowania współdziałającego bezpośrednio z systemem operacyjnym za pośrednictwem jego API

**Treści kształcenia:**

Komunikacja zewnętrzna, najczęściej spotykane przypadki. Port szeregowy RS232 konfiguracja i przesył danych. Komunikacja z wykorzystaniem portu USB. Transmisja danych poprzez sieć z użyciem protokołów UDP oraz TCP.
Kamera wideo jako urządzenie przechwytujące obraz. Przykładowe SDK kamery, pokazanie możliwości nią sterowania. Biblioteka DirectX jako uniwersalne rozwiązanie pozwalające na pobierać wideo z dużej części kamer komunikujących się po USB i FireWire. Konwencje wołania funkcji języka C++ oraz sposoby przekazywania parametrów. Wybrane, najczęściej używane rozkazy asemblera. Tworzenie oprogramowania z wykorzystaniem technologii SSE2.Podstawowe operacje na grafice z wykorzystaniem interfejsu GDI, w tym fonty, palety, bitmapy. Możliwości GDI+. Idee leżące u podstaw DirectX, wybrane obiekty i struktury. Użycie asemblera w połączeniu z technologią MMX do przetwarzania obrazów. Podział pamięci komputerowej, rodzaje, wybrane parametry ją charakteryzujące. Pamięć fizyczna oraz pamięć wirtualna. Zarządzanie zasobami pamięci, przydzielanie, zwalnianie itp.. Dysk twardy, szybki zapis oraz odczyt danych z wykorzystaniem Windows API. Drivery do czego służą i gdzie są wykorzystywane. Sterowniki dla urządzeń rzeczywistych i wirtualnych. Struktura i zasada budowy driverów w Linuxie oraz Windowsie. Kompilator Microsoft a kompilator Intela, porównanie. Możliwości wspomnianych kompilatorów oraz wykorzystanie niestandardowych ich dyrektyw celem poprawy parametrów wynikowych modułów uruchamialnych.

**Metody oceny:**

Ocena końcowa z przedmiotu jest średnią ważoną ocen uzyskanych z kolokwium przeprowadzonego zdalnie w formie testu jednokrotnego i wielokrotnego wyboru na platformie Microsoft Frames (waga - 0.4) oraz z projektu wykonanego indywidualnie przez studenta lub w przypadku złożonego projektu w grupie nie większej niż trzyosobowa, w który w działaniu wykorzystuje elementy programowania niskopoziomowego(waga - 0.6).

**Egzamin:**

nie

**Literatura:**

1. A. Daniluk, „RS 232C – praktyczne programowanie. Od Pascala i C++ do Delphi i Buildera”,wydanie III, Helion
2. A. Daniluk, „USB Praktyczne programowanie z Windows API w C++”,Helion, 2009
3. R. Scrimger, P. LaSalle, C. Leitzke, M. Parihar, M. Gupta, „TCP/IP. Biblia”,Helion, 2002
4. S. Kruk, „Kurs programowania w języku Asembler - dla średnio zaawansowanych”, MIKOM, 2001
5. K. R. Fall, W. R. Stevens, "TCP/IP od środka. Protokoły", Wydanie II, Helion, 2013
6. D. Farbaniec, "Asembler. Programowanie", Helion, 2019
7. P. Yosifovich, A. Ionescu, M. E. Russinovich, D. A. Solomon, "Windows od środka. Architektura systemu, procesy, wątki, zarządzanie pamięcią i dużo więcej", Wydanie VII, Helion, 2018

**Witryna www przedmiotu:**

brak

**Uwagi:**

brak

## Charakterystyki przedmiotowe

### Profil ogólnoakademicki - wiedza

**Charakterystyka PNP\_Ist\_o \_W01:**

Ma podstawową wiedzę w zakresie stosowania bibliotek systemowych w tworzonym oprogramowaniu. Zna postawy assemblera, potrafi tworzyć w nim proste procedury.

Weryfikacja:

Samodzielne wykonanie projektu na bazie biblioteki lub bibliotek systemowych.

**Powiązane charakterystyki kierunkowe:** K\_W05, K\_W06

**Powiązane charakterystyki obszarowe:** P6U\_W, I.P6S\_WG.o, III.P6S\_WG

### Profil ogólnoakademicki - umiejętności

**Charakterystyka PNP\_Ist\_o\_U01:**

Ma umiejętność samokształcenia i pogłębiania kwalifikacji w zakresie tworzenia oprogramowania wykorzystującego biblioteki niskopoziomowe

Weryfikacja:

Samodzielne wykonanie projektu na bazie biblioteki lub bibliotek systemowych.

**Powiązane charakterystyki kierunkowe:** K\_U08, K\_U09

**Powiązane charakterystyki obszarowe:** I.P6S\_UW.o, III.P6S\_UW.o

### Profil ogólnoakademicki - kompetencje społeczne

**Charakterystyka PNP\_Ist\_o\_K01:**

Ma świadomość posiadanej wiedzy, umiejętność pracy samodzielnej oraz w zespole.

Weryfikacja:

Samodzielne wykonanie projektu na bazie biblioteki lub bibliotek systemowych.

**Powiązane charakterystyki kierunkowe:**

**Powiązane charakterystyki obszarowe:**