

EFEKTY UCZENIA SIĘ I TREŚCI PROGRAMOWE ZAJĘĆ

Kierunek: **Technologie komputerowe**

Poziom studiów: **studia inżynierskie pierwszego stopnia**

Nazwa zajęć: **Wprowadzenie do studiowania**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna i rozumie poznawcze i pozapoznawcze uwarunkowania uczenia się oraz rozwija umiejętności wspierające efektywne uczenie się

w zakresie umiejętności:

1. potrafi identyfikować zasoby i ograniczenia, własne oraz otoczenia, wspierające i utrudniające studiowanie

w zakresie kompetencji społecznych:

1. jest gotów do podejmowania refleksji nad znaczeniem samodzielnego, odpowiedzialnego i zaangażowanego studiowania oraz kształtowania własnej ścieżki edukacyjnej i zawodowej

2. jest gotów do identyfikowania sytuacji trudnych w trakcie studiowania oraz poszukiwania adaptacyjnych strategii radzenia z nimi

3. jest gotów wykorzystywania wiedzy na temat możliwości rozwoju w ramach obligatoryjnej i fakultatywnej oferty uniwersytetu i wydziału

Treści programowe dla zajęć:

Studiowanie – nowe wyzwanie, jak mu sprostać. Podstawowe umiejętności niezbędne w procesie rozpoznawania swoich potrzeb, pasji, zainteresowań i zasobów. Rozpoznawanie zasobów otoczenia. Poznawcze i pozapoznawcze uwarunkowania procesu uczenia się i nauczania (strategie poznawcze, metapoznawcze, motywacyjne, zarządzanie zasobami, zarządzanie sobą w czasie). Samoregulacja w procesie uczenia się.

Czynniki ryzyka i wspierające w procesie studiowania, sytuacje trudne i radzenie sobie z nimi, stres i wypalenie zawodowe studentów: obciążenia (związane ze studiowaniem), adaptacja (adaptacyjne radzenie sobie), zasoby (co pomaga przystosować się do sytuacji studiowania).

Formalno-organizacyjno-prawne aspekty studiowania (studencki *savoir-vivre*, regulamin studiów – najważniejsze zagadnienia, obsługa systemu USOS). Obligatoryjna i fakultatywna oferta i możliwości jej wykorzystania.

Nazwa zajęć: **Elektronika analogowa i cyfrowa: Elektronika analogowa, Elektronika cyfrowa**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. Ma podstawową wiedzę niezbędną do zaprojektowania cyfrowego układu kombinacyjnego i sekwencyjnego

2. Zna podstawy teorii automatów

3. Zna podstawy fizyczne działania podzespołów elektronicznych i potrafi je zastosować do analizy układów elektronicznych

w zakresie umiejętności:

1. Potrafi wyznaczyć funkcję przełączającą układu kombinacyjnego i dokonać jej redukcji,

2. Potrafi zakodować funkcję przełączającą używając języka VHDL

3. Potrafi dokonać implementacji projektu układu kombinacyjnego w układzie FPGA, jego symulacji i sprawdzić działanie w rzeczywistym układzie

4. Potrafi zaimplementować prosty układ sekwencyjny wykorzystując konstrukcję procesu w języku VHDL lub edytor maszyny stanów skończonych i dokonać jego weryfikacji

5. Potrafi dokonać identyfikacji problemu i sformułować specyfikację projektową prostego analogowego układu elektronicznego. Potrafi zaprojektować i zrealizować prosty analogowy układ elektroniczny oraz zmierzyć jego parametry

6. Potrafi pozyskiwać informacje z literatury i baz danych oraz innych źródeł w języku polskim lub angielskim; potrafi integrować uzyskane informacje, dokonywać ich interpretacji, wyciągać wnioski oraz wykorzystywać je przy projektowaniu prostych układów elektronicznych

Treści programowe dla zajęć:

Rachunek zdań i rachunek zbiorów

Algebry Boole'a

Abstrakcyjna i strukturalna teoria automatów; opis w postaci maszyny stanów

Synteza układu kombinacyjnego
Kombinacyjne bloki funkcjonalne
Synteza układu sekwencyjnego
Sekwencyjne bloki funkcjonalne
Wprowadzenie do języka opisu sprzętu VHDL
Elementy elektroniczne czynne i bierne
Podstawy teorii obwodów i sygnałów
Półprzewodniki i przyrządy półprzewodnikowe (dioda, tranzystor bipolarny i unipolarny, elementy światłoczułe, dioda Led)
Wzmacniacze operacyjne
Zasilacze, prostowniki, filtry
Podstawy metrologii układów elektronicznych

Nazwa zajęć: **Grafika użytkowa**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Zna podstawowe etapy rozwoju grafiki użytkowej. Potrafi określić obszary zastosowań grafiki użytkowej.
2. Zna i potrafi omówić rozwój budowę i funkcjonowanie układu wzrokowego oraz konsekwencje różnic w percepcji istotne z punktu widzenia grafiki użytkowej (w tym grafiki www). Rozróżnia procesy leżące u podstaw postrzegania (fizyczne, fizjologiczne, psychofizyczne).
3. Zna podstawy teorii koloru i mieszania barw. Zna i rozumie podstawowe prawa widzenia barwnego i ich związek z przestrzenią barw. Rozumie różne aspekty percepcji barw. Zna zasady stosowane w komputerowym opisie barwy. Zna i potrafi opisać wzorniki, modele i przestrzenie barw (RGB, CMYK, HSV, CIEXYZ i CIELAB, CIELCH), wskazać różnice w zakresie metod konstrukcji i realizacji modelu, wymienić obszary ich zastosowań i podać przykłady. Zna metody separacji składowych i zapisu w odpowiedniej notacji. Zna zasady transformacji pomiędzy składowymi/współrzędnymi w systemach barwnych i potrafi wyliczyć składowe po transformacji. Zna metody reprezentacji stosowane w grafice www (standardy: DEC, heksadecymalny i heksadecymalny skrócony) oraz zasady konwersji. Potrafi skorzystać z odpowiednich narzędzi.
4. Zna i rozumie podstawowe pojęcia grafiki rastrowej (bitmapa, raster, głębia bitowa, rozdzielczość, wielkość pliku, poziomy kompresji, skalowalność, dithering, antialiasing, przeplot, przezroczystość) oraz związki pomiędzy nimi.
5. Zna cechy podstawowych formatów rastrowych (BMP, PNG, JPG, TIFF, GIF, LWF (JPG2000), DJVU, RAW) i wektorowych (SVG) z uwzględnieniem podstawowych pojęć. Zna zasady doboru odpowiedniego formatu w zależności od rodzaju zastosowań (www, grafika prezentacyjna i DTP), rozumie pojęcie ograniczeń i zniekształceń powstających w wyniku konwersji. Zna kwestie związane z bezpieczeństwem grafiki zamieszczanej na stronach www. Zna zasady optymalizacji grafiki dla potrzeb www.
6. Ma wiedzę z zakresu podstaw przetwarzania grafiki rastrowej. Zna i rozumie pojęcie operacji bezkontekstowych i kontekstowych na bitmapie. Potrafi opisać przekształcenia punktowe bezkontekstowe (operacje logiczne i arytmetyczne, liniowe i nieliniowe) oraz przekształcenia kontekstowe (filtry: liniowe oraz nieliniowe). Potrafi opisać podstawowe przekształcenia geometryczne w odniesieniu do grafiki rastrowej: przesunięcie (translacja), odbicie symetryczne, obrót, zniekształcenie. Zna zasady i ograniczenia związane ze zmianą rozdzielczości przestrzennej (metody interpolacyjne: najbliższego sąsiada, interpolacji dwuliniowej, interpolacji dwukubicznej). Rozumie pojęcie histogramu, normalizacja histogramu, operacje na histogramie. Zna pojęcie i rodzaje operacji kontekstowych. Zna podstawy filtracji i rozumie konstrukcję filtrów od strony matematycznej. Zna algorytmy stosowane do tworzenia filtrów graficznych. Potrafi wymienić cele filtracji i obszary zastosowań filtrów w grafice użytkowej. Zna najważniejsze algorytmy wykorzystywane w grafice rastrowej (kreślenie odcinków, łamanych, okręgów).
7. Zna podstawy przetwarzania grafiki wektorowej. Zna pojęcie prymitywu i potrafi wymienić prymitywy. Zna i rozróżnia podstawowe przekształcenia geometryczne w odniesieniu do grafiki wektorowej. Zna i potrafi scharakteryzować najważniejsze algorytmy wektorowe wykorzystywane w grafice (m.in. kreślenia krzywych Beziera, wypełniania (flood fill)). Rozumie ich zalety, wady i ograniczenia. Zna i rozumie różnice pomiędzy grafiką rastrową a wektorową oraz potrafi wymienić obszary ich zastosowań. Zna pojęcia wektoryzacji i rasteryzacji grafiki oraz obszary ich zastosowań. Zna podstawy typografii w kontekście grafiki rastrowej i wektorowej.
8. Zna pojęcie systemu identyfikacji wizualnej i jego roli w tworzeniu wizerunku firmy. Zna zasady projektowania loga i wie czym jest księga znaku.

9. Zna zasady i techniczne aspekty przygotowywania, optymalizacji grafiki dla potrzeb www, prezentacji oraz DTP (format, głębia bitowa, rozdzielczość, kompresja, wydajność: czas i sposób ładowania, wielkość pliku itp.)

w zakresie umiejętności:

1. Potrafi korzystać z narzędzi stosowanych do tworzenia grafiki użytkowej rastrowej i wektorowej.
2. Stosuje zasady i zalecenia doboru barwy w grafice (www, prezentacyjnej i DTP), potrafi skorzystać z odpowiednich narzędzi. Potrafi określić paletę zastosowaną w istniejącym projekcie.
3. Potrafi przygotować grafikę wektorową na potrzeby www (m.in. loga, banery reklamowe, infografikę).
4. Rozumie pojęcia: wektoryzacja i rasteryzacja grafiki. Zna zastosowania tych operacji. Potrafi przekształcić tekst na ścieżkę oraz zwektoryzować mapę bitową.
5. Potrafi dokonać obróbki obrazu, korekcji barwnej, optymalizacji i kompresji grafiki rastrowej.
6. Potrafi przygotowywać projekt zgodnie z zasadami DTP.
7. Potrafi dokonać separacji na kanały, redukcji kolorów, filtracji, zdefiniować własny filtr. Potrafi zinterpretować histogram.

Treści programowe dla zajęć:

Historia i ewolucja grafiki użytkowej. Obszary zastosowań grafiki użytkowej.

Rozwój, budowa i funkcjonowanie układu wzrokowego oraz konsekwencje różnic w percepcji istotne z punktu widzenia grafiki użytkowej (w tym grafiki www). Procesy leżące u podstaw postrzegania (fizyczne, fizjologiczne, psychofizyczne).

Podstawy teorii koloru i mieszania barw. Podstawowe prawa widzenia barwnego i ich związek z przestrzenią barw. Różne aspekty percepcji barw. Komputerowy opis barwy - modele i przestrzenie barw (RGB, CMYK, HSV, HSL, CIEXYZ i CIELAB, CIELCH), różnice w zakresie metod konstrukcji i realizacji modelu, obszary zastosowań, separacja składowych i zapis w odpowiedniej notacji. Zasady transformacji pomiędzy składowymi/współrzędnymi w systemach barwnych. Metody reprezentacji stosowane w grafice www (standardy: DEC, heksadecymalny i heksadecymalny skrócony) oraz zasady konwersji. Zasady i zalecenia doboru barwy w grafice (www, prezentacyjnej i dtp).

Podstawowe pojęcia grafiki rastrowej (bitmapa, raster, głębia bitowa, rozdzielczość, wielkość pliku, poziomy kompresji, skalowalność, dithering, antialiasing, przeplot, przezroczystość) oraz związki pomiędzy nimi.

Podstawowe formaty rastrowe (BMP, PNG, JPG, TIFF, GIF, LWF (JPG2000), DJVU, RAW) oraz wektorowe (SVG), z uwzględnieniem podstawowych pojęć (m.in. kompresja, przeplot). Zasady doboru odpowiedniego formatu w zależności od rodzaju zastosowań (www, grafika prezentacyjna i dtp), Ograniczenia zniekształcenia powstające w wyniku konwersji. Aspekty związane z bezpieczeństwem grafiki zamieszczanej na stronach www. Podstawowe zasady wyboru formatu dla potrzeb www.

Podstawy przetwarzania grafiki rastrowej. Przekształcenia punktowe bezkontekstowe (operacje logiczne i arytmetyczne, liniowe i nieliniowe) oraz przekształcenia kontekstowe (filtry: liniowe oraz nieliniowe). Podstawowe przekształcenia geometryczne w odniesieniu do grafiki rastrowej: przesunięcie (translacja), odbicie symetryczne, obrót, zniekształcenie. Zasady i ograniczenia związane ze zmianą rozdzielczości przestrzennej (metody interpolacyjne: najbliższego sąsiada, interpolacji dwuliniowej, interpolacji dwukubicznej). Histogram, normalizacja histogramu, podstawowe operacje na histogramie. Filtracja jako operacja kontekstowa. Podstawy filtracji i konstrukcja filtrów od strony matematycznej. Algorytmy stosowane do tworzenia filtrów graficznych. Cele filtracji i obszary zastosowań filtrów w grafice użytkowej. Zaawansowane filtry graficzne. Najważniejsze algorytmy wykorzystywane w grafice rastrowej (kreślenie odcinków, łamanych, okręgów).

Podstawy przetwarzania grafiki wektorowej. Pojęcie prymitywu i rodzaje prymitywów. Podstawowe przekształcenia geometryczne w odniesieniu do grafiki wektorowej. Najważniejsze algorytmy wektorowe wykorzystywane w grafice (m.in. kreślenie krzywych Beziery, flood fill). Ich zalety, wady i ograniczenia. Różnice pomiędzy grafiką rastrową a wektorową oraz obszary ich zastosowań. Wektoryzacja i rasteryzacja grafiki oraz obszary ich zastosowań. Podstawy typografii w kontekście grafiki rastrowej i wektorowej.

Pojęcie systemu identyfikacji wizualnej i jego roli w tworzeniu wizerunku firmy. Księga znaku. Zasady projektowania loga.

Zasady i techniczne aspekty przygotowywania, optymalizacji grafiki dla potrzeb www, prezentacji oraz dtp (format, głębia bitowa, rozdzielczość, kompresja, wydajność: czas i sposób ładowania, wielkość pliku itp.)

Nazwa zajęć: **Programowanie obiektowe i aspektowe**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zaawansowana znajomość zasad programowania obiektowego i zrozumienie koncepcji obiektów i interakcji pomiędzy obiektami.

w zakresie umiejętności:

1. zaawansowana umiejętność programowania używając wybranego przez siebie języka programowania. Tworzy własny rozbudowany projekt.

Treści programowe dla zajęć:

Omówienie (przypomnienie) zasad programowania obiektowego.

Omówienie koncepcji programowania aspektowego.

Dalej zajęcia mają charakter seminaryjny. Studenci tworzą swoje projekty i omawiają je w trakcie zajęć. Tematyka projektów dowolna do wyboru.

Nazwa zajęć: Fizyka i przetwarzanie danych w systemach wbudowanych: Mikrosterowniki, Mikroukłady komputerowe, Sterowniki PLC

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. Zna różnice pomiędzy systemem wbudowanym oraz systemem komputerowym

2. Identyfikuje elementy platformy Raspberry Pi i portu GPIO

3. Wie jaka jest struktura rejestrów wewnętrznych procesora ARM i definiuje ich funkcje w cyklach rozkazowych

4. Wyjaśnia w jaki sposób procesor ARM wykonuje procedury rozgałęzień, budowania stosu i komunikacji z pamięcią

w zakresie umiejętności:

1. Potrafi napisać i uruchomić prosty program w języku niskopoziomym (assembler), uruchomić go i zweryfikować poprawność jego działania.

2. Potrafi zaprojektować prosty algorytm sterowania i zaimplementować go w sterowniku PLC w co najmniej dwóch standardowych językach do tego służących (ladder diagram, instruction list, structured text, functional block chart, sequential function chart), uruchomić go i przetestować poprawność jego działania

3. Potrafi skonfigurować prostą wymianę danych pomiędzy PLC i PC

4. Wykorzystuje cykliczny rejestr przesuwany do implementacji efektywnych instrukcji wykonywanych przez procesor

5. Definiuje mody pinów portu GPIO, implementuje i uruchamia sterownik urządzeń zewnętrznych obsługiwanych przez magistrale portu GPIO

6. Potrafi wykorzystać operacje logiczne do negowania, ustawiania i zerowania wybranych bitów w bajcie.

7. Potrafi zaimplementować pętlę i rozgałęzienie w kodzie assemblerowym.

8. Obsługuje proste układy wejścia-wyjścia (klawiatura sekwencyjna, wyświetlacz 7-segmentowy) za pomocą programów zaimplementowanych w kodzie assemblerowym.

Treści programowe dla zajęć:

Podstawy systemów wbudowanych

Podstawy sterowników PLC, norma IEC61131

Środowisko developerskie (SDK) CoDeSys

Język programowania PLC ladder diagram (LD)

Język programowania PLC instruction list (IL)

Język programowania PLC structured text (ST)

Język programowania PLC functional block diagram (FBD)

Język programowania PLC sequential functional chart (SFC)

Funkcje standardowe środowiska CoDeSys

Assembler 8-bitowych procesorów Intel

Assembler 16-bitowych procesorów Intel

Assembler ARM- podstawowy architektury procesora ARM, etapy kompilacji kodu niskiego poziomu, struktura kodu niskiego poziomu w kontekście architektury ARM

Efektywne wykorzystanie rejestrów w trybie rozkazowym (instrukcje LDR/ STR),

Wykorzystanie debuggера do deasemblacji kodu i analizy wykonywania rozkazów przez procesor

Assembler ARM- obsługa funkcji wejścia i wyjścia

Assembler ARM- wykorzystanie cyklicznego rejestru przesuwanego

Assembler ARM- praca w trybie ARM i Thumb, stos, wywoływanie i obsługa procedur

Tworzenie wstawek assemblerowych ARM w programowaniu wysokopoziomym (język C)

Mapa pinów GPIO, numeracja w notacji BCM, zasoby biblioteki WiringPi, sterownik urządzenia obsługiwane przez pin GPIO w modzie wyjściowym
Napisanie sterownika urządzenia obsługiwane przez piny GPIO w modzie wyjściowym i wejściowym z wykorzystaniem magistrali I2C
Napisanie sterownika urządzenia obsługiwane przez piny GPIO w modzie wyjściowym, wejściowym oraz w modzie PWM
Cykl rozkazowy procesora
Organizacja pamięci i tryby adresowania
Układy wejścia-wyjścia i przerwania

Nazwa zajęć: **Fizyczne i informatyczne podstawy telekomunikacji**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna i rozumie podstawowe parametry związane z ruchem
2. zna zależności związane z transmisją bezprzewodową
3. zna techniki wielodostępu
4. zna podstawowe modulacje oraz potrafi zaprezentować przebiegi sygnałów w dziedzinie czasu i częstotliwości
5. rozumie podstawowe zasady fizyki dotyczące propagacji radiowej

w zakresie umiejętności:

1. potrafi wyjaśnić zasadę działania podstawowych systemów, sieci i usług telekomunikacyjnych w sieciach stacjonarnych
2. potrafi wyjaśnić zasadę działania podstawowych systemów, sieci i usług telekomunikacyjnych w sieciach radiowych
3. potrafi wykorzystać znane technologie radiowe przy budowie układów bezprzewodowego sterowania (np. dronami)

w zakresie kompetencji społecznych:

1. jest gotów tłumaczyć zasady działania systemów telekomunikacyjnych

Treści programowe dla zajęć:

Architektura i topologia telekomunikacyjnych sieci przewodowych i bezprzewodowych
Sieci telefoniczne
Rozległe sieci transmisji danych
Elementy optyczne w systemach światłowodowych
Komutacja łączy i pakietów
Łącza transmisyjne przewodowe i bezprzewodowe
System telefonii komórkowej (budowa, usługi, działanie)
Obliczenia propagacyjne i planowanie radiowe
Wymiarowanie traktów komunikacji rozmównej z komutacją łączy
Wielodostęp
Modulacje

Nazwa zajęć: **Sztuczna Inteligencja - Tworzenie Środowisk Rzeczywistości Cyfrowych**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. Zna i rozumie w zakresie rozszerzonym procesy integracji SI w systemach RC.
2. Zna i rozumie w zakresie rozszerzonym, wsparte SI, techniki tworzenia zasobów dla RC.
3. Zna i rozumie strategie optymalizacji zasobów (w tym generowanych SI) i scen SI/RC.
4. Zna i rozumie w zakresie rozszerzonym techniki materiałowe i tekstuowania RC, w tym rolę SI.
5. Zna i rozumie w zakresie rozszerzonym techniki renderowania wizualnego RC i optymalizacji SI.
6. Zna i rozumie zasady projektowania immersyjnego audio z użyciem SI.
7. Zna i rozumie metody tworzenia dynamicznych efektów środowiskowych.
8. Zna i rozumie w zakresie rozszerzonym systemy animacji i kontroli ruchu.
9. Zna i rozumie metody proceduralnego i generatywnego tworzenia treści.
10. Zna i rozumie w zakresie rozszerzonym techniki kodowania wspomaganego SI dla platform RC.
11. Zna i rozumie metody analizy wydajności, optymalizacji oraz przygotowania środowisk dla symulacji i pre-treningu agentów SI (Postaci Graczy (PG)/Non-PG).

w zakresie umiejętności:

1. Potrafi konstruować w zakresie rozszerzonym procesy przetwarzania i integracji SI w RC.
2. Potrafi tworzyć złożone zasoby RC z wykorzystaniem wsparcia SI.
3. Potrafi optymalizować zasoby (w tym generowane SI) i sceny SI/RC różnymi metodami.

4. Potrafi implementować w zakresie rozszerzonym materiały i tekstury, wykorzystując SI.
5. Potrafi implementować w zakresie rozszerzonym renderowanie, optymalizację wizualną i wirtualną kinematografię.
6. Potrafi projektować i implementować immersyjne systemy audio przy wsparciu SI.
7. Potrafi tworzyć dynamiczne efekty środowiskowe.
8. Potrafi implementować w zakresie rozszerzonym systemy animacji i kontroli ruchu.
9. Potrafi stosować techniki manualne, proceduralne i generatywne do tworzenia treści RC.
10. Potrafi rozszerzać funkcjonalność platform RC poprzez kodowanie wspomagane SI.
11. Potrafi analizować wydajność systemów SI/RC, optymalizować je i przygotowywać środowiska dla agentów SI (PG/NPG), w tym pre-treningu NPG.

w zakresie kompetencji społecznych:

1. Jest gotów/gotowa do specjalistycznego rozwoju w zakresie integracji SI i RC.
2. Jest gotów/gotowa do pracy w zespole przy złożonych projektach SI/RC.
3. Jest gotów/gotowa do zarządzania zadaniami w projektach z elementami SI.
4. Jest gotów/gotowa do oceny i adaptacji nowych technologii SI w RC.
5. Jest gotów/gotowa do efektywnej pracy zespołowej w projektach SI/RC.
6. Jest gotów/gotowa do uzasadniania decyzji technicznych i artystycznych (w tym użycia SI).
7. Jest gotów/gotowa do kompromisów oraz adaptacji do tempa rozwoju technologicznego w projektowaniu złożonych systemów SI/RC.

Treści programowe dla zajęć:

Zasady przetwarzania i integracji SI w złożonych systemach RC.
Wsparte rozwiązaniami SI procesy tworzenia zasobów dla środowisk RC.
Techniki wspomaganego SI modelowania RC.
Techniki tworzenia materiałów, cieniowania i przetwarzania tekstur RC.
Integracja i optymalizacja zasobów generowanych przez SI.
Realizacje RC scen i renderowanie o wysokiej wierności wizualnej.
Wsparte SI projektowanie immersyjnych przestrzeni dźwiękowych.
Tworzenie dynamicznych efektów środowiskowych.
Metody animacji i kontroli ruchu.
Proceduralne a generatywne tworzenie treści dla środowisk RC.
Rozszerzanie platform RC – Kodowanie i tworzenie narzędzi wspomagane SI.
Techniki budowania i kompozycji światów SI/RC w dużej skali i na dużą skalę.
Wsparcie rozwiązaniami SI środowisk RC i postaci graczy (PG).
Kontrolowane przez SI Non-PG (NPG) i syntetyczny pre-trening modeli NPG w środowisku RC (min. na platformach społecznościowych RC).
Projekt: Wsparcie rozwiązaniami SI budowa środowiska SI/RC.

Nazwa zajęć: **Przetwarzanie sygnałów fizycznych**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna zasady próbkowania sygnałów

w zakresie umiejętności:

1. potrafi obliczać spłot przebiegów
2. potrafi reprezentować sygnał w dziedzinie czasu i częstotliwości
3. potrafi projektować filtry cyfrowe
4. potrafi modelować operacje przetwarzania sygnału z wykorzystaniem oprogramowania
5. potrafi wykorzystać języki programowania (Matlab, Python) do przetwarzania sygnałów

w zakresie kompetencji społecznych:

1. jest gotów wykorzystywać przetwarzanie sygnałów na potrzeby budowy systemów, rozwiązań i aplikacji informatycznych

Treści programowe dla zajęć:

Próbkowanie i kwantowanie sygnału
Spłot i przekształcenie Z
Transformata Fouriera
Filtracja cyfrowa
Modulacje
Symulacja procesów przetwarzania sygnałów

Nazwa zajęć: **Wprowadzenie do programowania WWW**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Zna strukturę i sposób kodowania stron webowych z uwzględnieniem różnic w poszczególnych wersjach HTML aż do 5.0 oraz CSS do 3.0. Potrafi wskazać różnice na przykładach.
2. Zna pojęcie responsywności strony i metody realizacji.
3. Zna podstawowe etapy tworzenia serwisów www. Potrafi określić obszary zastosowań i wymienić podstawowe narzędzia. Rozumie ideę i zastosowania Systemów Zarządzania Treścią (CMS).
4. Zna i potrafi opisać kierunek zmian środowiska CMS na przykładzie różnych wersji Joomla!
5. Posiada wiedzę na temat architektury aplikacji internetowych (na przykładzie CMS Joomla!).
6. Zna narzędzia administratora dla CMS Joomla!
7. Zna metody konfigurowania CMS i zarządzania użytkownikami. Zna i rozumie zakresy uprawnień.
8. Zna podstawowe zasady tworzenia dynamicznych stron internetowych. Zna i rozumie różnice pomiędzy artykułami statycznymi i dynamicznymi.
9. Zna zasady tworzenia, modyfikacji i instalacji szablonów dla CMS Joomla!
10. Zna konfigurację i metodę działania menedżera mediów.
11. Zna różnice pomiędzy komponentami, modułami i rozszerzeniami.
12. Zna budowę formularzy i sposób ich osadzania na stronach www.
13. Zna sposób instalacji i konfiguracji wielojęzycznego serwisu - strony www i e-sklepu (na przykładzie CMS Joomla!).

w zakresie umiejętności:

1. Potrafi zbudować prostą stronę w oparciu o język znaczników, potrafi osadzić arkusze stylów wewnętrzne i zewnętrzne, zapewnić dynamiczną zmianę wyglądu strony przez użytkownika za pomocą arkusza stylów.
2. Potrafi utworzyć responsywną stronę.
3. Potrafi utworzyć ikony na potrzeby strony www w oparciu o CSS oraz zastosować transformacje 2D/3D.
4. Zna i korzysta z narzędzi dla administratora. Potrafi skonfigurować CMS i zarządzać użytkownikami.
5. Potrafi stworzyć artykuły dynamiczne i statyczne, dodać elementy graficzne (utworzone na zajęciach z Grafiki użytkowej).
6. Potrafi przygotować pliki szablonu CMS Joomla! i paczkę instalacyjną zawierającą te pliki. Potrafi zainstalować i zmodyfikować szablon.
7. Potrafi skonfigurować i wykorzystać menedżera mediów.
8. Potrafi zainstalować, skonfigurować i zastosować komponent, moduł i rozszerzenie.
9. Potrafi skonfigurować i osadzić formularze na stronach www.
10. Potrafi zbudować stronę firmową oraz sklep internetowy w oparciu o System Zarządzania Treścią na przykładzie Joomla!
11. Potrafi przeprowadzić instalację i konfigurację wielojęzycznego serwisu - strony www i e-sklepu.

Treści programowe dla zajęć:

Struktura i sposób kodowania stron webowych z uwzględnieniem różnic w poszczególnych wersjach HTML aż do 5.0, CSS do 3.0 (na przykładach).

Podstawowe etapy tworzenia serwisów www, obszary zastosowań i podstawowe narzędzia. Idea i zastosowania Systemów Zarządzania Treścią (CMS). Kierunek zmian środowiska CMS na przykładzie różnych wersji Joomla!

Konfiguracja CMS, zarządzanie użytkownikami i zakresy ich uprawnień. Narzędzia dla administratora. Artykuły statyczne, dynamiczne.

Budowa szablonu. Tworzenie, instalacja i modyfikacja szablonu.

Menedżer mediów.

Komponent, moduł i rozszerzenie.

Formularze na stronach www.

Budowa strony firmowej oraz sklepu internetowego w oparciu o System Zarządzania Treścią na przykładzie Joomla!

Instalacja i konfiguracja wielojęzycznego serwisu - strony www. Instalacja, konfiguracja i uruchomienie e-sklepu w ramach tworzonego serwisu.

Nazwa zajęć: Efektywne algorytmy dla rzeczywistych problemów programistycznych

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Rozróżnia funkcje rekurencyjne i pierwotnie rekurencyjne.
2. Zna metody optymalizacji algorytmów (iteracyjne, rekurencyjne, rekurencja ogonowa, tablicowa).

3. Zna wieloargumentowe funkcje rekurencyjne (Ackermana, Mac Carthy'ego, Sudana) i ich zastosowania.
4. Zna kod Graya, kod binarny i heksadecymalny i jego podstawowe zastosowania.
5. Rozumie pojęcie systemu funkcji iterowanych (IFS).
6. Zna pojęcie systemu Lindenmayera (L-systemu), rozróżnia systemy kontekstowe i bezkontekstowe, potrafi operować zestawem reguł w celu generowania obiektów o budowie fraktalnej (również z czynnikiem losowym).
7. Zna definicje, własności i metody konstrukcji podstawowych obiektów fraktalnych 1D, 2D oraz 3D.

w zakresie umiejętności:

1. Potrafi zaimplementować rekurencję w odniesieniu do różnych obiektów matematycznych oraz zbadać złożoność zastosowanego algorytmu.
2. Potrafi zastosować metody optymalizacji algorytmów (iteracyjne, rekurencyjne, rekurencja ogonowa, tablicowa)
3. Potrafi zaimplementować wieloargumentowe funkcje rekurencyjne (Ackermana, Mac Carthy'ego, Sudana)
4. Potrafi zaimplementować kod Graya, kod binarny i heksadecymalny i zastosować go do prostych zagadnień praktycznych.
5. Potrafi zastosować metodę przekształceń zwężających (IFS) w ujęciu Hutchinsona do iterowania obiektów matematycznych (funkcji i fraktali samopodobnych Cantora, Kocha, Hilberta, Sierpińskiego)
6. Potrafi operować zestawem reguł w celu generowania obiektów o budowie fraktalnej (również z czynnikiem losowym).
7. Potrafi zaimplementować algorytmy fraktalne.

Treści programowe dla zajęć:

Algorytmy iteracyjne, rekurencyjne, rekurencja ogonowa, tablicowa. Metody optymalizacji.

Funkcje rekurencyjne i pierwotnie rekurencyjne.

Wieloargumentowe funkcje rekurencyjne (Ackermana, Mac Carthy'ego, Sudana, Bucka, Dwyera (D-funkcja)) i ich zastosowania w teorii obliczeń.

Kod Graya, kod binarny i heksadecymalny i ich zastosowania w robotyce i grafice.

System funkcji iterowanych (IFS) oraz metoda przekształceń zwężających w ujęciu Hutchinsona. Iterowanie obiektów matematycznych - funkcji i fraktali samopodobnych (Cantora, Kocha, Peano, Hilberta, Lévy'ego, Sierpińskiego, Heighway'a, Hartera) oraz ich zastosowanie w grafice komputerowej. Podstawy systemu Lindenmayera (L-systemu), gramatyka systemu, systemy kontekstowe i bezkontekstowe, deterministyczne i stochastyczne. Metody generowania obiektów o budowie fraktalnej. L-systemy 2D i 3D.

Fraktale. Definicje, własności i metody konstrukcji podstawowych obiektów fraktalnych 1D, 2D oraz 3D (zbiory Mandelbrota, Julii, Mandelbulb, kostka Menger, piramida Sierpińskiego etc.)

Nazwa zajęć: **Algorytmy i struktury danych**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. objaśnia rodzaje i typy złożoności obliczeniowej algorytmów, wyraża je w sposób matematyczny i rozumie metody tworzenia algorytmów
2. definiuje podstawowe algorytmy i struktury danych, zna typowe dla nich operacje, implementacje i zastosowania

w zakresie umiejętności:

1. implementuje podstawowe algorytmy sortowania, rozumie różnice między nimi, wskazuje przykłady ich zastosowań, zna ich złożoność obliczeniową
2. realizuje operacje wyszukiwania i selekcji elementów w ciągach liczbowych oraz operacje na samych ciągach, określa ich złożoność obliczeniową
3. konstruuje algorytmy z zastosowaniem struktur danych; zna przykłady zastosowań drzew binarnych; identyfikuje warianty algorytmów, dla których wskazuje przewagi jednych nad drugimi

w zakresie kompetencji społecznych:

1. korzysta ze źródeł dodatkowych do samodzielnego poszerzania wiedzy i rozwiązywania problemów w implementacji algorytmów

Treści programowe dla zajęć:

Złożoność czasowa i pamięciowa algorytmów, ich semantyczna poprawność; ocena złożoności obliczeniowej w granicy asymptotycznie dużych rozmiarów danych

Wstęp matematyczny obejmujący elementy algebry, probabilistyki i równań rekurencyjnych w zakresie niezbędnym dla prowadzenia analizy algorytmów

Algorytmy sortowania i analiza ich złożoności obliczeniowej; sposoby implementacji algorytmów, ilustracja metod ich konstrukcji
Operacje z użyciem ciągów liczbowych i ich różne implementacje: wyszukiwanie elementów, różne rodzaje selekcji, scalanie ciągów oraz ich implementacje
Proste, złożone i oparte na grafach struktury danych; zastosowania struktur danych do rozwiązywania problemów informatycznych; zbiory dynamiczne i operacje na zbiorach
Zasady opracowywania własnych implementacji prostych algorytmów obejmujące aspekty poprawności działania oraz analizy ich złożoności obliczeniowej

Nazwa zajęć: **Elementy fizyki 2**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. posiada poszerzoną wiedzę z zakresu elektryczności i magnetyzmu oraz praw elektromagnetyzmu.
2. potrafi rozwiązywać zadania z elektryczności, magnetyzmu oraz elektromagnetyzmu korzystając z technik matematyki wyższej na poziomie „Zbioru zadań – podstawy fizyki” J.Walkera, rozdz. 22-34.

w zakresie umiejętności:

1. potrafi pozyskiwać informacje z dostępnej literatury i baz danych oraz innych źródeł. Potrafi także wykorzystać uzyskane informacje do dokonania analizy wyników doświadczeń i sformułować proste wnioski.

w zakresie kompetencji społecznych:

1. potrafi rozwiązywać problemy fizyczne samodzielnie, jak również współpracować w ramach małego zespołu.

Treści programowe dla zajęć:

Ładunek elektryczny i pole elektryczne
Prawo Gaussa
Potencjał elektryczny
Pojemność elektryczna
Prąd elektryczny i opór elektryczny
Obwody elektryczne
Pole magnetyczne
Zjawisko indukcji i indukcyjność
Magnetyzm materii (dia-, para-, ferro-)
Drgania elektromagnetyczne

Nazwa zajęć: **Zarządzanie projektami**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna zasady prowadzenia projektów

w zakresie umiejętności:

1. potrafi przygotować dokumentację zarządczą do projektu
2. umie zaplanować projekt
3. potrafi zdefiniować ryzyka w projekcie

w zakresie kompetencji społecznych:

1. zna metody komunikacji na potrzeby zarządzania projektem
2. zna zasady organizacji współpracy w projekcie

Treści programowe dla zajęć:

O projektach i zarządzaniu nimi
Rola menadżera projektu
Definiowanie projektu
Budowanie i utrzymywanie zespołu
Planowanie i szacowanie
Plan projektu
Radzenie sobie z ryzykiem i niepewnością
Sprawowanie kontroli w czasie
Zarządzanie punktami styknymi z otoczeniem
Komunikacja i dokumentacja
Zakończanie projektu

Nazwa zajęć: **Systemy operacyjne 1**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. objaśnia, jak wykonywane są rozkazy w komputerze, rolę sygnałów i przerwań, jak system operacyjny komunikuje się z pamięciami i urządzeniami I/O
2. wymienia i omawia rolę i funkcjonalności systemów operacyjnych, ich modularności, interfejsy programisty i użytkownika oraz ich działanie w modelu klient-serwer
3. przedstawia systemy operacyjne MS Windows oraz typu UNIX, wskazuje ich założenia projektowe, zastosowania oraz kluczowe funkcjonalności
4. objaśnia kierunki rozwoju systemów operacyjnych i używa kryteriów ich oceny, omawia wpływ systemów operacyjnych na bezpieczeństwo pracy, korzystanie z sieci i wykorzystanie multimediiów

w zakresie umiejętności:

1. tworzy maszyny wirtualne lokalnie oraz zarządza nimi na platformie chmurowej, instaluje systemy MS Windows i Linux, potrafi je obsłużyć i zabezpieczyć oraz administrować nimi
2. programuje w języku powłoki sh i tworzy skrypty PowerShell oraz obsługuje transfery plików

w zakresie kompetencji społecznych:

1. korzysta ze źródeł systemowych, literaturowych i internetowych, także anglojęzycznych, do aktualizacji swojej wiedzy, by samodzielnie rozwiązywać problemy w obsłudze i administrowaniu systemami operacyjnymi

Treści programowe dla zajęć:

Systemy komputerowe (podstawowe komponenty, rejestry procesora i wykonywanie rozkazów, sygnały, przerwania, pamięci komputera, komunikacja z urządzeniami I/O)

Rola, zadania, modularność i funkcjonalności systemów operacyjnych (rola, zadania, moduły i ich funkcjonalności, interfejsy programisty i użytkownika, model klient-serwer)

MS Windows NT i późniejsze wersje (założenia projektowe, elementy składowe systemu, podsystemy środowiskowe, system plików, praca w sieci, interfejs programisty)

Uniksowe systemy operacyjne, w tym Linux (założenia projektowe, interfejsy programisty i użytkownika, zarządzanie procesami i pamięcią, systemy plików i I/O, komunikacja międzyprocesowa)

Ocena i rozwój systemów operacyjnych (kryteria oceny, przyszłościowe projekty, wpływ na bezpieczeństwo pracy, korzystanie z sieci i wykorzystanie multimediiów)

Maszyny wirtualne oraz zarządzanie nimi na platformie chmurowej, instalacja systemów MS Windows i Linux, ich obsługa i zabezpieczanie oraz administrowanie nimi, skrypty PowerShell oraz powłoki sh, obsługa transferu plików

Nazwa zajęć: **Architektura komputerów**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. wymienia i opisuje podstawowe elementy oraz mechanizmy działania systemu komputerowego.
2. porównuje najważniejsze typy architektury pamięci, listy rozkazów oraz komunikacji.
3. tłumaczy matematyczne podstawy kodowania liczb, logiki matematycznej oraz jej sprzętowej realizacji.
4. wyjaśnia zasady pisania, asemblowania i uruchamiania programów w języku assembler dla wybranych platform.

w zakresie umiejętności:

1. rozróżnia wartość liczbową zapisaną w różnych systemach kodowania.
2. stosuje poznaną wiedzę do przeanalizowania oraz napisania programu w assemblerze i uruchomienia go na wybranych platformach.
3. rozróżnia typ architektury zbioru rozkazów na podstawie trybów adresowania dostępnych w assemblerze dla danej platformy.

Treści programowe dla zajęć:

Architektura systemu komputerowego (podstawowe komponenty, magistrala systemowa, sygnały sterujące).

Architektura procesora (podstawowe bloki i rejestry, model programowy procesora).

Organizacja pamięci i tryby adresowania.

Cykl rozkazowy procesora, podprogramy i przerwania.

Układy wejścia-wyjścia.

Operacje arytmetyczno logiczne, dostęp do bitów.

Kodowanie liczb, matematyczne podstawy układów cyfrowych.

Architektura mikrokontrolera 8051 oraz platformy Raspberry Pi.

Assembler mikrokontrolera 8051 oraz platformy Raspberry Pi, implementacja pętli i rozgałęzień w kodzie assemblerowym.

Nazwa zajęć: Elementy gier komputerowych

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. wymienia i definiuje elementy i cele gier komputerowych z podziałem na kategorie oraz rozumie ich wpływ na odbiór gry przez gracza.
2. wymienia najważniejsze dokumenty projektowe oraz ich typową zawartość.

w zakresie umiejętności:

1. stosuje poznane narzędzia i techniki do samodzielnego stworzenia gry komputerowej.
2. analizuje tworzoną grę pod względem zastosowanych elementów gier komputerowych oraz szczelności poziomów z punktu widzenia reguł formalnych.

w zakresie kompetencji społecznych:

1. ocenia przydatność zasobów i narzędzi pod kątem wykorzystania we własnym projekcie przy uwzględnieniu ewentualnych ograniczeń licencyjnych.
2. ocenia możliwości monetyzacji tworzonej gry, włączając w to zarabianie na grach darmowych, wczesny dostęp oraz crowdfunding.

Treści programowe dla zajęć:

Elementy gier i ich przykładowe klasyfikacje.

Reguły formalne w levelach.

Rodzaje celów. Metody ich wprowadzania i podtrzymywania.

Dokumenty projektowe - przegląd.

Tworzenie gier z wykorzystaniem silnika Godot: scena, węzeł, korzeń. Sposoby dostosowywania funkcjonalności węzłów do własnych potrzeb.

Podstawy języka GDScript: składnia języka i struktury danych (w tym słowniki).

Wzorce projektowe: obserwator, wstrzykiwanie zależności, singleton (tak jakby).

Współprogramy i yield: zawieszanie i odwieszanie procesu, współpraca z sygnałami.

Grafika i animacje. Interpolacja w zastosowaniu do animacji.

Muzyka i dźwięki w grze. Problem odległości od źródła dźwięku.

Przegląd zasobów i narzędzi – dobrych i tanich, a nawet darmowych (również do użytku komercyjnego).

Przegląd możliwości monetyzacji gry niezależnej.

Nazwa zajęć: Statystyka laboratoryjna

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna podstawowe pojęcia z zakresu analizy danych, statystyki laboratoryjnej oraz teorii niepewności pomiarowych
2. zna zasady bezpiecznego posługiwania się prostymi narzędziami i przyrządami pomiarowymi

w zakresie umiejętności:

1. potrafi wykorzystać podstawowe narzędzia informatyczne do rozwiązywania prostych problemów z zakresu statystyki laboratoryjnej i analizy danych
2. potrafi zaplanować prosty eksperyment fizyczny, wykonać statystyczną analizę danych pomiarowych i przeprowadzić wnioskowanie statystyczne
3. potrafi przygotować raport z przeprowadzonego eksperymentu fizycznego zawierający analizę danych pomiarowych, prezentację graficzną danych oraz wnioskowanie statystyczne
4. potrafi pozyskiwać i korzystać z informacji źródłowych w celu rozwiązywania prostych problemów z zakresu informatyki, matematyki i fizyki, również w języku angielskim

w zakresie kompetencji społecznych:

1. jest gotów podjąć pracę w zespole uwzględniając kompetencje własne oraz pozostałych członków grupy z poszanowaniem zasad etyki zawodowej i praw autorskich
2. potrafi przygotować raport końcowy z wykonanej analizy statystycznej w sposób przejrzysty i zrozumiały posługując specjalistycznym słownictwem

Treści programowe dla zajęć:

- podstawowe pojęcia analizy danych
- podstawowe metody statystycznego opracowania danych
- podstawowe zagadnienia teorii niepewności pomiarowych wraz z kodyfikacją ISO
- podstawowe narzędzia informatyczne wykorzystywane do analizy danych i graficznej prezentacji danych
- zasady bezpiecznego posługiwania się narzędziami i przyrządami pomiarowymi
- planowanie przebiegu i wykonania prostego eksperymentu fizycznego
- przeprowadzenie prostego eksperymentu fizycznego z uwzględnieniem zasad bezpieczeństwa
- analiza ilościowa prostych zjawisk fizycznych z wykorzystaniem metod statystycznych

- biegle wykorzystywanie narzędzi informatycznych do analizy danych i prezentacji graficznej danych
- redagowanie raportu końcowego z analizy danych zawierającego wnioskowanie statystyczne

Nazwa zajęć: **Zastosowania teorii automatów**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Student/ka zna i rozumie klasyczne pojęcia oraz wyniki z zakresu teorii automatów.
2. Student/ka zna możliwości oraz języki do implementacji podstawowych automatów w przemysłowych systemach sterowania.

w zakresie umiejętności:

1. Student/ka potrafi realizować podstawowe i złożone automaty.
2. Student/ka potrafi wykorzystać dostępne urządzenia i systemy sterowania do projektowania automatów cyfrowych.

w zakresie kompetencji społecznych:

1. Student/ka wykazuje samokrytycyzm w pracy programistyczno-projektowej; rozumie i odczuwa potrzebę ciągłego dokształcania się - podnoszenia kompetencji zawodowych i osobistych, a zwłaszcza śledzenia i analizowania najnowszych osiągnięć związanych z reprezentowaną dyscypliną naukową.
2. Student/ka potrafi myśleć i działać w sposób niezależny, kreatywny i przedsiębiorczy, przejawia inicjatywę w kreowaniu nowych idei i poszukiwaniu innowacyjnych rozwiązań; wykazuje inicjatywę w określaniu nowych obszarów badań

Treści programowe dla zajęć:

Podstawowe pojęcia teorii automatów - wprowadzenie.

Wyrażenia regularne i wzorce.

Teoria automatów w układach mikroprocesorowych przeznaczonych do sterowania urządzeń i procesów przemysłowych.

Automaty sekwencyjne - asynchroniczne i synchroniczne.

Podstawowe i złożone układy arytmetyki - implementacja w przemysłowych systemach sterowania.

Rejestry, liczniki i timery - implementacja w przemysłowych systemach sterowania.

Podział sterowników PLC ze względu na: moc, szybkość oraz obsługę I/O.

Języki programowania - model programowania i model komunikacji. Podstawowe funkcje, bloki funkcyjne, działania logiczne i arytmetyczne.

Rozwiązania sprzętowe i programowe implementacji automatów: od prostych bramek logicznych po złożone struktury systemów sterowania i informatycznych (obiekt + PLC + SCADA/BMS).

Praktyczna prezentacja automatyzacji i oprogramowania przykładowych obiektów przemysłowych - podejście projektowe.

Konfiguracja sprzętowa i programowa. Obsługa I/O, komunikacja z urządzeniami peryferyjnymi oraz nadrzędnymi (HMI + SCADA/BMS).

Nazwa zajęć: **Prawo autorskie, prawo patentowe, przedsiębiorczość**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna rodzaje własności intelektualnej i potrafi je właściwie scharakteryzować
2. zna źródła prawa własności intelektualnej
3. zna i rozumie pojęcia: utwór, dozwolony użytek, plagiat, prawo cytatu, wynalazek, wzór użytkowy, wzór przemysłowy, znak towarowy, oznaczenia geograficzne
4. rozumie potrzebę ochrony własności intelektualnej i jej znaczenie dla przedsiębiorczości
5. rozumie istotę prawa autorskiego w Internecie
6. zna i rozumie podstawowe zasady tworzenia i rozwoju różnych form przedsiębiorczości

w zakresie umiejętności:

1. umie w praktyce wykorzystywać rozwiązania prawne przyjęte w prawie autorskim i prawie własności przemysłowej
2. potrafi stosować prawo autorskie w pracach naukowych i dyplomowych

w zakresie kompetencji społecznych:

1. rozumie i docenia znaczenie uczciwości intelektualnej w działaniach własnych i innych osób; ma świadomość problemów etycznych w kontekście rzetelności badawczej (plagiat czy też auto-plagiat)
2. jest gotów/gotowa do odpowiedzialnego i przedsiębiorczego działania w ramach różnych form działalności gospodarczej

Treści programowe dla zajęć:

Własność intelektualna, rodzaje własności intelektualnej, kapitał intelektualny, innowacje, know-how, gospodarka oparta na wiedzy; zarządzanie własnością intelektualną w przedsiębiorstwie.

Prawo autorskie - pojęcie utworu i twórcy.

Prawo autorskie krajowe i unijne; ochrona praw autorskich osobistych i majątkowych.

Dozwolony użytek prywatny i publiczny, prawo cytatu, plagiat, prawo autorskie w pracach naukowych i dyplomowych, prawo autorskie w Internecie.

Własność przemysłowa i jej ochrona, prawo własności przemysłowej; wynalazki, wzory użytkowe, wzory przemysłowe, znaki towarowe i oznaczenia geograficzne.

Ochrona patentowa, procedura patentowa, patentowe bazy danych; prawo ochronne na wzory użytkowe; prawo z rejestracji wzorów przemysłowych; prawo znaków towarowych; ochrona oznaczeń geograficznych.

Procedury uzyskiwania ochrony na poszczególne dobra niematerialne; umowy w prawie autorskim i prawie własności przemysłowej.

Znaczenie ochrony intelektualnej dla przedsiębiorczości.

Wpływ prawa nowych technologii na obecne prawa własności intelektualnej.

Działalność gospodarcza, prawo działalności gospodarczej, formy działalności gospodarczej, pojęcie przedsiębiorcy, kategorie przedsiębiorców, umowy w działalności gospodarczej, rejestracja działalności gospodarczej, przedsiębiorczość akademicka.

Nazwa zajęć: Wstęp do informatyki i elektroniki cyfrowej

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. objaśnia, jak wykonywane są operacje arytmetyczne w komputerach, uwarunkowania i ograniczenia obliczeń binarnych wykonywanych na rejestrach określonej długości
2. objaśnia zasady projektowania i działania układów cyfrowych wychodząc od logiki
3. objaśnia podstawowe zasady i ograniczenia przetwarzania informacji w komputerze
4. wymienia i omawia zagadnienia, którymi zajmuje się dyscyplina naukowa Informatyka
5. objaśnia podstawy dyskretyzacji informacji nieznakowych, teorii informacji Shannona, prostego jej kodowania, przesyłania i wykrywania zakłóceń

w zakresie umiejętności:

1. wykonuje rachunki binarne na rejestrach określonej długości w zapisach uzupełnieniowych

Treści programowe dla zajęć:

Arytmetyka binarna maszyn cyfrowych (binarne zapisy uzupełnieniowe, stało- i zmiennoprzecinkowe; nadmiar, niedomiar, precyzja i uwarunkowania obliczeń)

Podstawy projektowania i działania układów cyfrowych (podstawy logiki, wyrażenia i równania logiczne, bramki logiczne, zasady projektowania i działania układów cyfrowych)

Automatyzacja przetwarzania informacji (komputery, przetwarzanie, sygnały i sterowanie, automaty i algorytmy, rozstrzygalność, obliczalność, maszyna Turinga, model komputera von Neumanna i jego rozwinięcia, współbieżność i przetwarzanie równoległe)

Dyscyplina naukowa Informatyka (przegląd zagadnień, które obejmuje informatyka, ACM, IEEE i światowe standardy programowe Informatyki, systemy informatyczne)

Przetwarzanie informacji w komputerze i jej przesyłanie (wstęp do teorii informacji Shannona, kodowanie, przepustowość kanału, kontrola poprawności)

Nazwa zajęć: Architektura internetu

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. analizuje i objaśnia działanie sieci komputerowych w oparciu o warstwowe modele architektury Internetu i znajomość protokołów komunikacyjnych.
2. zna najważniejsze instytucje odpowiedzialne za standardy sieciowe i charakter udostępnianych przez nie dokumentów.

w zakresie umiejętności:

1. projektuje i buduje sieci lokalne z dostępem do Internetu w standardzie Ethernet – przewodowe i bezprzewodowe.
2. konfiguruje stacje robocze i urządzenia sieciowe oraz analizuje ruch sieciowy.
3. instaluje i konfiguruje podstawowe serwisy internetowe, takie jak: routing, DNS, DHCP, FTP, HTTP i objaśnia zasady ich działania.
4. korzysta z różnych źródeł: systemowych, literaturowych i internetowych, w tym także anglojęzycznych, do samodzielnego rozwiązywania problemów konfiguracji administracji lokalnych sieci komputerowych.

Treści programowe dla zajęć:

Modele warstwowe sieci komputerowych (OSI, TCP/IP) Geneza Internetu i standardy jego architektury.
Instytucje odpowiedzialne za standardy internetowe i ich zasoby.
Architektura i elementy sieci lokalnej opartej na standardzie Ethernet.
Metody transmisji danych – analogowej i cyfrowej. Media transmisyjne. Sieci przewodowe i bezprzewodowe. Rodzaje okablowania.
Fizyczna i logiczna struktura sieci. Topologie, metody dostępu do nośnika. Urządzenia sieciowe i łączenie sieci.
Podstawowe protokoły komunikacyjne warstwy internetowej i ich współdziałanie.
Zasady adresacji IPv4 i IPv6. Mapowanie adresów fizycznych i logicznych. Trasowanie.
Sterowanie transmisją – protokoły: TCP i UDP. Transmisja grupowa (strumieniowa).
Podstawowe usługi sieciowe – DNS, DHCP, TELNET (SSH), FTP, SMTP, MIME, HTTP.
Zarządzanie siecią i bezpieczeństwo sieciowe.

Nazwa zajęć: **Inżynieria oprogramowania**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Zna i rozumie rolę inżynierii oprogramowania oraz podstawowe pojęcia z nią związane.
2. Zna i rozumie wybrane metody i narzędzia stosowane w inżynierii oprogramowania.
3. Zna i rozumie wybrane metody i narzędzia do modelowania systemów informatycznych.
4. Zna i rozumie co składa się na jakość oprogramowania.
5. Zna i rozumie wybrane dobre praktyki wykorzystywane do rozwiązywania powtarzających się problemów przy tworzeniu kodu systemu informatycznego.
6. Zna i rozumie działanie wybranych metodyk wytwarzania oprogramowania.
7. Zna i rozumie aspekty etyczne związane z inżynierią oprogramowania.

w zakresie umiejętności:

1. Potrafi wykorzystać wybrane metody i narzędzia w procesie wytwarzania oprogramowania.
2. Potrafi zastosować odpowiednie metody do modelowania systemu informatycznego.
3. Potrafi zastosować wybrane elementy zwiększające jakość tworzonego oprogramowania.
4. Potrafi zastosować wybrane dobre praktyki do rozwiązywania powtarzających się problemów.
5. Potrafi efektywnie działać w wybranych metodykach wytwarzania oprogramowania.

w zakresie kompetencji społecznych:

1. Jest gotów/gotowa do oceny i działania zgodnie z normami etycznymi przy wytwarzaniu oprogramowania.
2. Jest gotów/gotowa do stosowania nowych narzędzi i metod w zakresie inżynierii oprogramowania.
3. Jest gotów/gotowa do pełnienia określonej roli w procesach związanych z wytwarzaniem oprogramowania.

Treści programowe dla zajęć:

Rola inżynierii oprogramowania i najczęściej zadawane pytania.
Wybrane metody i narzędzia stosowane w inżynierii oprogramowania.
Metody i narzędzia modelowania systemów informatycznych.
Jakość w tworzeniu oprogramowania.
Wzorce projektowe i dobre praktyki stosowane do rozwiązywania powtarzających się problemów.
Metodyki wytwarzania oprogramowania.

Nazwa zajęć: **Matematyka dyskretna**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna podstawowe pojęcia z zakresu kombinatoryki i teorii grafów.
2. zna najważniejsze prawa rachunku zdań oraz rachunku zbiorów.

w zakresie umiejętności:

1. potrafi wykorzystać odpowiednie metody zliczania do rozwiązywania konkretnych problemów.
2. zbudować i przeanalizować graf odpowiadający określonej problemowi informatycznemu o charakterze sprzętowym (np. organizacja sieci), jak i algorytmicznym.

Treści programowe dla zajęć:

1. Rachunek zdań (elementarna logika matematyczna)
2. Ogólne techniki dowodzenia twierdzeń, w tym indukcja matematyczna
1. Elementy naiwnej teorii zbiorów (rachunek zbiorów)
2. Podstawowe relacje dwuargumentowe
1. Podstawowe zasady i prawa przeliczania

2. Schematy wyboru
3. Tożsamości kombinatoryczne
4. Najważniejsze algorytmy kombinatoryczne
1. Podstawowe typy grafów
2. Drogi i cykle
3. Podstawowe algorytmy przeszukiwania
1. Drzewa i lasy
2. Konstrukcja drzewa spinającego
3. Znajdowanie dróg w grafie i inne zastosowania

Nazwa zajęć: **Optyka z fotoniką**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. potrafi przedstawić i omówić równania Maxwella w postaci całkowitej lub różniczkowej oraz wyprowadzić równanie falowe dla ośrodka jednorodnego
2. umie wyprowadzić podstawowe zasady optyki geometrycznej i wykorzystać je do opisu podstawowych zjawisk w optyce: odbicia, załamania fal na granicy między dwoma ośrodkami
3. potrafi objaśnić formowanie struktury pasmowej w kryształach fonicznych i możliwości modelowania propagacją fal elektromagnetycznych przy jej wykorzystaniu
4. potrafi wyprowadzić relację dyspersji dla fal elektromagnetycznych, polarytonów plazmonowych i omówić różnice i podobieństwa z falami elektromagnetycznymi z zakresu podczerwieni i światła widzialnego
5. wskazać i omówić działanie przykładowych urządzeń stosowanych w fotonice i optyce.

w zakresie umiejętności:

1. umie wywieść podstawowe właściwości fal elektromagnetycznych z równań Maxwella
2. umie posługiwać się operatorami wektorowymi do opisu zjawisk falowych.
3. Wykazuje umiejętność czytania artykułów naukowych ze zrozumieniem i umiejętność przedstawiania wyników, streszczenia i znaczenia badań na piśmie.

Treści programowe dla zajęć:

Równania Maxwella i równanie falowe (wprowadzenie do rachunku wektorowego i tensorowego, równania w próżni, ośrodku dielektrycznym i niejednorodnym, warunki brzegowe).

Fale elektromagnetyczne (widmo fal elektromagnetycznych, fale płaskie, fale monochromatyczne, wektor Poytinga, prędkość fazowa i grupowa fali, paczki falowe, optyka geometryczna, prawo Snella, prawo Fresnela).

Kryształy foniczne (fale elektromagnetyczne w ośrodku periodycznym, foniczna struktura pasmowa, metody obliczeniowe struktury pasmowej, zastosowania kryształów fonicznych w optoelektronice, meta-materiały).

Plazmony i polarytony – sprzężenie fali elektromagnetycznej z drganiami gazu elektronowego i sieci jonowej (gaz elektronów swobodnych, model Drudego, perspektywy zastosowania plazmoniki w optoelektronice i fotonice).

Podstawowe urządzenia stosowane w optyce i fotonice (budowa i zasada działania: anteny, lasery, faldowody, izolatory, modulatory i detektory).

Nazwa zajęć: **Fizyka nośników informacji**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. posiada podstawową wiedzę z zakresu fizyki ciała stałego, przemian fazowych.
2. ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie fizyki nośników informacji oraz fotoniki niezbędną do zrozumienia fizycznych podstaw działania systemów zapisu i przetwarzania informacji.
3. rozumie znaczenie niektórych zjawisk fizycznych dla przetwarzania informacji.

w zakresie umiejętności:

1. potrafi sformułować podstawowe prawa i zasady fizyczne oraz zależności stosowane w technologiach komputerowych używając formalizmu matematycznego.
2. potrafi numerycznie modelować proste zjawiska fizyczne i przygotować graficzną prezentację wyników.

w zakresie kompetencji społecznych:

1. potrafi dyskutować i formułować opinie na temat nowoczesnych technologii na styku fizyki, informatyki i telekomunikacji.
2. potrafi pracować samodzielnie wykazując gotowość do stałego pogłębiania wiedzy.

3. rozumie i widzi potrzebę redukcji zapotrzebowania na energię przy przetwarzaniu informacji.

Treści programowe dla zajęć:

Podstawy fizyki ciała stałego: statyczny obraz sieci krystalicznej, dynamika sieci (drgania sieci, ciepło właściwe sieci)

Układy dwustanowe i przemiany fazowe: ferromagnetyki, ferroelektryki i ferroelastyki

Typy zapisu informacji: dyski magnetyczne, dyski optyczne, zapis półprzewodnikowy

Technologiczne aspekty zapisu informacji: nanoelektronika, nanolitografia, spintronika

Nazwa zajęć: **Podstawy programowania**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna strukturę programu w języku C# oraz odpowiednik w języku C.
2. zna elementarne algorytmy.
3. zna podstawowe typy danych i metody konwersji oraz rzutowania typów.
4. zna podstawowe operatory m.in. logiczne, arytmetyczne, porównania, równości, przypisania.
5. zna podstawowe instrukcje programistyczne: warunkowe, wyboru, iteracji, sterujące itp.
6. zna zasady deklarowania, tworzenia i wywoływania metod (funkcji) oraz zna kilka sposobów przekazywania parametrów.
7. zna strukturalne typy danych: tablice i łańcuchy oraz operacje na nich. Zna metody inicjalizacji generatora liczb pseudolosowych.
8. zna podstawowe algorytmy sortowania.
9. zna zasady rekurencji, wie na czym polega zastosowanie rekurencji przy tworzeniu metody.
10. zna obsługę błędów i wyjątków w języku C#.

w zakresie umiejętności:

1. potrafi napisać, skompilować i uruchomić prostą aplikację konsolową (program strukturalny). Potrafi użyć debugera do analizy kodu.
2. potrafi zakodować prosty algorytm.
3. potrafi prawidłowo zadeklarować typ danych. Potrafi dokonać konwersji i rzutowania typów.
4. potrafi stosować podstawowe operatory logiczne, arytmetyczne, porównania, równości, przypisania.
5. potrafi stosować instrukcje programistyczne: warunkowe, wyboru, iteracji i sterujące.
6. potrafi deklarować, tworzyć i wywoływać metody (funkcje) oraz potrafi przekazać parametry na kilka sposobów.
7. potrafi posługiwać się tablicami i łańcuchami, wykonywać na nich operacje. Potrafi zaimplementować generator liczb pseudolosowych.
8. potrafi napisać metodę sortowania.
9. potrafi zastosować metody rekurencyjne i iteracyjne do rozwiązania tego samego zagadnienia.
10. umie obsłużyć wyjątek i błąd systemowy. Potrafi napisać własną obsługę błędów.

Treści programowe dla zajęć:

Platforma .NET i środowisko programistyczne Visual Studio. Tworzenie aplikacji konsolowych. Metoda programowania. Pojęcie algorytmu oraz sposoby jego zapisu.

Struktura programu i metody w odniesieniu do aplikacji konsolowych (kodowanie, kompilacja, uruchamianie, testowanie i debugowanie).

Typy danych, użycie zmiennych oraz wykorzystanie operatorów. Konwersja i rzutowanie typów danych. Operacje inkrementacji i dekrementacji.

Podstawowe operatory logiczne, arytmetyczne, porównania, równości itp.

Stosowanie instrukcji programistycznych. Instrukcje warunkowe, wyboru, skoku i obsługa wyjątków.

Instrukcje iteracyjne i sterujące.

Tworzenie metod (funkcji) oraz sposoby przekazywania argumentów.

Strukturalne typy danych: tablice (jednowymiarowe i wielowymiarowe) i łańcuchy. Podstawowe operacje na ww. typach. Generator liczb pseudolosowych.

Metody sortowania.

Pojęcie rekurencji. Metody i struktury rekurencyjne.

Obsługa błędów i wyjątków (własna i systemowa).

Nazwa zajęć: **Projektowanie baz danych**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna etapy tworzenia baz danych: Analiza. Projekt. Kodowanie. Testowanie i instalacja. Problemy pojawiające się przy tworzeniu aplikacji w modelu klient-serwer. Zagadnienia związane z dwoistą naturą

aplikacji typu klient-serwer. Kwestie decyzji projektowych związane z teorią i praktyką tworzenia baz danych. Normalizacja. Zagadnienia związane z definiowaniem przeznaczenia aplikacji.

2. zna zagadnienia związane z: Wyborem typu aplikacji. Prezentacją procesów operacyjnych w formie graficznej. Projektowaniem hierarchii formularzy i raportów. Wyszukanie przydatnych, niezależnie opracowanych elementów programu (bibliotek). Opracowaniem kalendarza prac. Tworzeniem aplikacji.

3. zna zagadnienia związane z: Klasycznymi problemami izolowania transakcji. Zarządzaniem transakcjami. Poziomem izolowania transakcji w DBMS. Sterowaniem transakcjami z wykorzystaniem języka SQL. Izolowaniem transakcji na serwerze. Wyborem odpowiedniego poziomu izolowania transakcji. Sterowaniem transakcjami - komendy SET TRANSACTION, COMMIT i ROLLBACK. Systemy sterowania współbieżnością: pesymistyczne i optymistyczne. Zarządzaniem protokołem transakcji. Minimalizowaniem informacji pamiętanych w protokole. Podziałem dużych transakcji na mniejsze.

w zakresie umiejętności:

1. potrafi zasady modelowania reguł przetwarzania. Zagadnienia związane z umieszczeniem reguł logiki aplikacji: Implementację reguł logiki aplikacji na serwerze i po stronie klienta. Silne i słabe strony implementacji serwerowej. Silne i słabe strony implementacji klienckiej. Silne i słabe strony oprogramowania pośredniczącego (multi tier architecture).
2. potrafi projektować formularze - formularze wspomagające podejmowanie decyzji, - formularze interaktywne, - formularze do wprowadzania danych; projektować raporty; testować aplikacje pod kątem zgodności z zakładanym przeznaczeniem i zakresem funkcji; instalować aplikacje.
3. potrafi tworzyć aplikacje bazodanowe.

Treści programowe dla zajęć:

Omówienie zagadnień przedstawionych w efektach kształcenia.

Etapy tworzenia baz danych: Analizy. Projektowania. Kodowania. Testowania i instalacji.

Problemy pojawiające się przy tworzeniu aplikacji w modelu klient-serwer. Zagadnienia związane z dwoistą naturą aplikacji typu klient-serwer.

Kwestie decyzji projektowych związane z teorią i praktyką tworzenia baz danych. Normalizacja. Zagadnienia związane z definiowaniem przeznaczenia aplikacji.

Zasady modelowania reguł przetwarzania. Zagadnienia związane z umieszczeniem reguł logiki aplikacji: Implementację reguł logiki aplikacji na serwerze i po stronie klienta. Silne i słabe strony implementacji serwerowej. Silne i słabe strony implementacji klienckiej. Silne i słabe strony oprogramowania pośredniczącego (*multi tier architecture*).

Znajomość zagadnień związanych z: Klasycznymi problemami izolowania transakcji. Zarządzaniem transakcjami. Poziomem izolowania transakcji w DBMS. Sterowaniem transakcjami z wykorzystaniem języka SQL. Izolowaniem transakcji na serwerze. Wyborem odpowiedniego poziomu izolowania transakcji.

Nazwa zajęć: Matematyka 1

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna podstawowe pojęcia związane z algebrą liczb zespolonych.
2. zna podstawowe pojęcia związane z algebrą wektorów.
3. zna pojęcia granicy ciągu i szeregu liczb rzeczywistych oraz granicy i ciągłości funkcji.
4. zna definicję pochodnej i jej interpretację oraz reguły różniczkowania.
5. zna definicję pochodnej cząstkowej i jej interpretację.

w zakresie umiejętności:

1. umie przeprowadzać obliczenia na liczbach zespolonych.
2. umie wykonywać operacje na wektorach.
3. umie obliczać granice ciągów i szeregów liczbowych.
4. umie obliczać granice funkcji i badać ich ciągłość.
5. umie obliczać pochodną pierwszego i wyższych rzędów funkcji jednej zmiennej oraz stosować rachunek różniczkowy do badania przebiegu zmienności funkcji.
6. umie obliczać pochodne cząstkowe pierwszego i wyższych rzędów.

Treści programowe dla zajęć:

Działanie i grupa. Ciało liczbowe. Ciało liczb zespolonych: jednostka urojona, postać wielomianowa, algebraiczna, geometryczna, trygonometryczna, wykładnicza. Dodawanie liczb zespolonych, mnożenie, pierwiastki, logarytm, moduł.

Wektory: wielkości wektorowe i skalarnie, obraz geometryczny, własności działań, obraz geometryczny – klasy i wektor swobodny. Przestrzeń i podprzestrzeń wektorowa. Kombinacja liniowa i liniowa

niezależność. Baza i wymiar przestrzeni wektorowej. Iloczyn skalarny, przestrzeń unitarna. Baza ortogonalna i ortonormalna.

Wektory w kartezjańskiej przestrzeni trójwymiarowej: dodawanie wektorów, mnożenie przez liczbę, iloczyn skalarny, iloczyn wektorowy, iloczyn mieszany. Liniowa niezależność wektorów.

Ciąg liczbowy: określenia, działania, ciągi w zbiorach uporządkowanych. Granica ciągu: własności ciągów zbieżnych, ciągi rozbieżne, symbole nieoznaczone.

Szeregi liczbowe: warunek konieczny zbieżności, zbieżność bezwzględna i warunkowa. Kryteria zbieżności: Leibniza, Cauchy'ego, d'Alamberta, porównawcze. Szeregi potęgowe: twierdzenie Abela, promień zbieżności.

Granica funkcji: granica funkcji w punkcie, granica funkcji w nieskończoności. Asymptoty poziome, ukośne i pionowe.

Własności odwzorowania / funkcji: dziedzina, zbiór wartości. Własności funkcji: suriekcja, injekcja, bijekcja, funkcja odwrotna. Ograniczoność i ekstrema globalne, ekstrema lokalne, monotoniczność.

Definicja ciągłości funkcji w punkcie. Funkcje ciągłe. Własności funkcji ciągłych (tw. Weierstrassa, tw. Darboux).

Pochodna funkcji w punkcie (i iloraz różnicowy) i jej interpretacja geometryczna, funkcja pochodna (przykłady). Pochodne funkcji elementarnych. Działania na funkcjach: pochodna sumy funkcji, pochodna iloczynu funkcji, pochodna ilorazu funkcji. Pochodna funkcji złożonej. Różniczka funkcji jednej zmiennej, operator różniczkowania. Pochodne i różniczki wyższych rzędów. Pochodna funkcji odwrotnej.

Zastosowania pochodnych: reguła de L'Hospitala, monotoniczność funkcji a znak pochodnej, ekstrema lokalne (warunek konieczny i wystarczający). Wypukłość i punkt przegięcia. Równanie stycznej do wykresu funkcji. Badanie przebiegu zmienności funkcji.

Twierdzenia o wartościach średnich: tw. Rolle'a, tw. Lagrange'a, tw. Cauchy'ego. Wzór na szereg Taylora i Maclaurina.

Pochodne funkcji skalarnych i wektorowych, pochodna iloczynu skalarnego i wektorowego.

Pochodna cząstkowa: definicja, przykłady, własności. Pochodne cząstkowe wyższych rzędów. Różniczka zupełna i pochodna zupełna. Ekstrema funkcji wielu zmiennych.

Gradient pola skalarnego, operator Hamiltona (nabla), dywergencja i rotacja pola wektorowego, operator Laplace'a.

Nazwa zajęć: **Języki programowania**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. wie jakie są elementarne instrukcje wykonywane przez procesory.
2. wie co to są języki wysokiego poziomu proceduralne i obiektowe.
3. wie co to są tablice, wskaźniki, łańcuchy znakowe (stringi) oraz programowanie dynamiczne.
4. wie co to są klasy, ich budowa oraz warianty w programowaniu obiektowym.

w zakresie umiejętności:

1. potrafi pisać i kompilować proste programy z wykorzystaniem składni języka C oraz C++.
2. potrafi pisać programy (w językach C i C++) używając różnych typów danych, tablic, struktur i wskaźników.
3. potrafi pisać programy z dynamiczną alokacją pamięci.
4. potrafi definiować klasy, odpowiednio używać zdefiniowane przez siebie typy danych, odpowiednio kodować zdefiniowane problemy z wykorzystaniem programowania obiektowego.
5. potrafi używać w programowaniu dziedziczenie, dziedziczenia wielokrotnego oraz funkcji wirtualnych.

w zakresie kompetencji społecznych:

1. rozumie potrzebę uczenia się i doskonalenia swoich umiejętności przez całe życie, jednocześnie potrafi inspirować i organizować proces uczenia się innych osób
2. ma wiedzę z zakresu własności intelektualnej (własność kodów źródłowych).
3. umiejętność identyfikacji różnych metod programowania, a jednocześnie posiada świadomość skutków wadliwie działających programów i systemów informatycznych.

Treści programowe dla zajęć:

Informacje ogólne o językach programowania. Kompilacja. Identyfikatory proste (wbudowane), typy danych oraz ich reprezentacja binarna. Zakres widoczności zmiennych. Podstawowe operatory.

Instrukcje sterujące, instrukcje warunkowe oraz pętle. Formatowanie wydruków. Użycie funkcji.

Tablice, macierze, wskaźniki, łańcuchy znakowe (stringi).

Arytmetyka wskaźników i dynamiczna alokacja pamięci.

Struktury i Unie w ujęciu programowania proceduralnego a klasy w ujęciu programowania obiektowego

Wstęp do programowania obiektowego: klasy, składowe klasy, konstruktory i destruktory, funkcje stałe, deklaracje i definicje klasy, zagnieżdżanie klas.

Analiza i programowanie proceduralne oraz zorientowane obiektowo. Różnice w kodach źródłowych. Pojęcie własności intelektualnej w programowaniu (własność kodów źródłowych a plików wykonywalnych).

Referencje, przeciążanie funkcji i konstruktorów. Funkcje zaprzyjaźnione.

Dziedziczenie i dziedziczenie wielokrotne. Dostęp do składowych klas bazowych i potomnych. Wirtualne klasy bazowe.

Operacje I/O w języku C++. Manipulatory, Operatory << oraz >>.

Nazwa zajęć: **Teoria informacji i kodowanie**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna i rozumie fundamentalne zasady i koncepcje teorii informacji i kodowania; zna standardowe narzędzia teorii informacji i kodowania dla gromadzenia, analizy i prezentacji danych; rozumie własności, charakterystyki i funkcje informacji oraz entropii;
2. zna standardowe metody kodowania danych informatycznych

w zakresie umiejętności:

1. potrafi obliczyć ilość informacji, entropię źródła informacji oraz redundancję metody kodowania; stosuje standardowe metody kodowania danych informatycznych
2. potrafi oszacować efektywność metod kodowania przy realizacji systemu informatycznego

w zakresie kompetencji społecznych:

1. potrafi pozyskiwać informacje o nowoczesnych sposobach kodowania informacji z literatury i innych źródeł, integrować je, dokonywać ich interpretacji, wyciągać logiczne wnioski i formułować opinie
2. potrafi uzupełniać i doskonalić nabytą wiedzę i umiejętności; jest przygotowany do samodzielnego zdobywania i doskonalenia wiedzy oraz umiejętności profesjonalnych i badawczych; rozumie podstawy teorii informacji i kodowania i potrafi funkcjonować w społeczeństwie informacyjnym

Treści programowe dla zajęć:

Notacja

Podstawy kodowania, klasyfikacja kodów

Teoria informacji, entropia, ilość informacji

Optymalne kodowanie, kodowanie Huffmana i Shannona-Fano

Kodowanie słownikowe

Kodowanie arytmetyczne

Nazwa zajęć: **Projektowanie układów logicznych**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Zna podstawy logiki dwuwartościowej oraz algebry Boole'a
2. Zna bramki logiczne i potrafi je łączyć w większe układy

w zakresie umiejętności:

1. Potrafi tworzyć projekty układów logicznych w języku VHDL oraz wykonać symulację ich działania
2. Potrafi zaprojektować kombinacyjne układy logiczne (np. multipleksery) oraz weryfikować poprawność ich działania
3. Potrafi zaprojektować arytmetyczne układy logiczne (np. sumatory) oraz weryfikować poprawność ich działania
4. Potrafi zaprojektować sekwencyjne układy logiczne (np. przerzutniki) oraz weryfikować poprawność ich działania

Treści programowe dla zajęć:

Algebra Boole'a

Środowisko Xilinx ISE

Bramki logiczne

Kombinacyjne układy logiczne (multipleksery)

Arytmetyczne układy logiczne (sumatory, subtraktory)

Sekwencyjne układy logiczne (przerzutniki)

Nazwa zajęć: **Projekt zespołowy**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna zasady prowadzenia projektów

w zakresie umiejętności:

1. potrafi przygotować projekt
2. potrafi rozdzielić prace w zespole projektowym
3. potrafi dobrać technologie do komponentów wdrażanego systemu
4. potrafi opracować dokumentację do wdrażanego rozwiązania

w zakresie kompetencji społecznych:

1. potrafi zarządzać pracami projektowymi

Treści programowe dla zajęć:

Metoda zwinna scrum
Przygotowanie projektu informatyczne
Organizacja pracy w zespole projektowym
Kontrola postępu prac
Opracowanie dokumentacji rozwiązania i jego wdrożenia
Potrafi wybrać technologie i je zastosować do realizacji rozwiązań na potrzeby projektu

Nazwa zajęć: Matematyka 2

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. posiada wiedzę na temat metod matematycznych z zakresu analizy matematycznej
2. zna metody algebry liniowej i podstawowe pojęcia z tego zakresu

w zakresie umiejętności:

1. potrafi obliczyć całki nieoznaczone różnymi metodami (całkowanie przez części, całkowanie przez podstawienie, całki funkcji wymiernych)
2. potrafi obliczyć całki oznaczone (Riemanna), w szczególności do rozwiązania pewnych problemów fizycznych
3. potrafi rozwiązywać równania różniczkowe zwyczajne i cząstkowe z naciskiem na pewne problemy fizyczne (np.: oscylator harmoniczny, obwody LC, równanie dyfuzji)
4. potrafi rozwiązywać równania i układy równań zespolonych
5. potrafi wykonywać operacje na macierzach, w tym rozwiązywanie zagadnienia własnego
6. potrafi rozwiązywać układy równań liniowych wykorzystując wzory Cramera

Treści programowe dla zajęć:

Całka nieoznaczona (funkcja pierwotna), własności, podstawowe wzory, całkowanie przez części, wzór na całkowanie przez podstawienie
Całka (oznaczona) Riemanna, własności, interpretacja geometryczna, podstawowe twierdzenie rachunku całkowego oraz związek między całką oznaczoną i nieoznaczoną, wzory na całkowanie przez części oraz przez podstawienie dla całki oznaczonej, całki niewłaściwe: po nieograniczonych przedziałach i z nieograniczonych funkcji
Równania różniczkowe zwyczajne i cząstkowe
Liczby zespolone II
Macierze, działania na macierzach, wyznacznik macierzy: własności oraz sposoby liczenia, twierdzenie Laplace'a; minor, dopełnienie algebraiczne, rząd macierzy, transpozycja macierzy, macierz odwrotna, wzór na macierz odwrotną, zagadnienie własne
Układy równań liniowych. Wzory Cramera

Nazwa zajęć: Fizyczne Laboratorium Mikrokomputerowe

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna elementy środowiska programistycznego LabView.
2. zna zaawansowane narzędzia LabView służące pomiarowi sygnałów elektrycznych.
3. zna zaawansowane narzędzia LabView służące generacji sygnałów elektrycznych sterujących lub niosących informację.
4. zna podstawowe narzędzia LabView służące analizie sygnału elektrycznego rejestrowanego za pomocą układu pomiarowego.

w zakresie umiejętności:

1. potrafi korzystać ze środowiska programistycznego LabView w zakresie podstaw programowania, w tym w szczególności w zakresie operacji wejścia-wyjścia, obliczeń matematycznych przeprowadzanych na zbiorach danych, prezentacji tych danych i wyników tych obliczeń.
2. potrafi korzystając z LabView stworzyć oprogramowanie komunikujące się z dowolnym urządzeniem pomiarowym dysponując opisem instrukcji użycia tego urządzenia oraz odpowiednimi sterownikami pozwalającymi na komunikację z tym urządzeniem.

3. potrafi korzystając z LabView stworzyć oprogramowanie komunikujące się z dowolnym urządzeniem generującym wybrany sygnał dysponując opisem instrukcji użycia tego urządzenia oraz odpowiednimi sterownikami pozwalającymi na komunikację z tym urządzeniem.

4. potrafi przygotować raport z budowy układu kontrolno-pomiarowego i uzyskanych przy użyciu tego układu wyników.

w zakresie kompetencji społecznych:

1. potrafi pracować w zespole, w szczególności dzielić się wiedzą lub korzystać z niej z innymi członkami zespołu.

Treści programowe dla zajęć:

Wprowadzenie do podstaw programowania w LabView: środowisko programistyczne, elementarne elementy graficznego języka programowania G - konstrukcje programowania strukturalnego, typy i struktury danych, funkcje, operatory matematyczne i logiczne, elementy interfejsu.

Łańcuchy i operacje wejścia-wyjścia.

Zaawansowane elementy LabView: struktury zdarzeń, sekwencyjne, formuła node, pomiar czasu.

Prezentacja danych w LabView: rodzaje wykresów, formatowanie danych.

Utworzenie generatora sygnału sinusoidalnego i rejestratora tego sygnału przy użyciu LabView i dwóch kart dźwiękowych, oraz ich wykorzystanie w pomiarze rezonansu w układzie typu RLC.

Utworzenie cyfrowego oscyloskopu przy użyciu LabView i karty kontrolno-pomiarowej firmy National Instruments.

Utworzenie aplikacji kontrolno-pomiarowej realizującej algorytm proporcjonalno-całkowo-różniczkowy w stabilizacji temperatury układu grzejącego.

Utworzenie określonej przez studenta aplikacji korzystającej z kontrolera Kinect firmy Microsoft.

Utworzenie aplikacji rejestrującej obraz kamery przemysłowej, rozpoznającej kształty wybranych przedmiotów widocznych w obrazie kamery i zliczającej te przedmioty.

Nazwa zajęć: Elementy fizyki 1

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. potrafi wytłumaczyć matematyczne opisy zjawisk z zakresu wybranych zagadnień mechaniki i termodynamiki.

w zakresie umiejętności:

1. potrafi rozwiązywać proste problemy i zagadnienia dotyczące wybranych zagadnień mechaniki i termodynamiki korzystając ze standardowych metod matematycznych.

2. potrafi planować i przeprowadzać proste eksperymenty, wspomagane również komputerowo, oraz poddawać krytycznej analizie otrzymane rezultaty.

Treści programowe dla zajęć:

Kinematyka: ruch prostoliniowy, wektory, ruch w dwóch i trzech wymiarach, elementy kinematyki relatywistycznej.

Dynamika: siła i zasady dynamiki Newtona.

Energia i zasada zachowania energii.

Układy cząstek i zderzenia.

Dynamika bryły sztywnej.

Grawitacja.

Statyka i dynamika płynów.

Ruch drgający.

Nazwa zajęć: Zajęcia wyrównawcze z fizyki

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna i rozumie podstawową wiedzę na temat ogólnych praw fizyki, wielkości fizycznych oraz oddziaływań fundamentalnych.

2. zna i rozumie uporządkowaną wiedzę z zakresu: mechaniki punktu materialnego, optyki, elektromagnetyzmu.

w zakresie umiejętności:

1. potrafi wyjaśnić podstawowe prawa fizyki i potrafi wytłumaczyć na ich podstawie przyczyny i przebieg zjawisk fizycznych.

2. potrafi analizować i rozwiązywać proste problemy fizyczne opierając się na poznanych prawach i metodach fizyki, stosując odpowiedni aparat matematyczny.

Treści programowe dla zajęć:

Wielkości wektorowe w fizyce: prędkość, pęd, siła, moment pędu i moment siły. Działania na wektorach (dodawanie, odejmowanie oraz mnożenie skalarne i wektorowe). Jednostki fizyczne podstawowe i pochodne w układzie SI. Alfabet grecki. Przedrostki zmniejszające i zwiększające jednostki.

Ruch postępowy punktu materialnego i wielkości go opisujące (ruch jednostajny prostoliniowy i ruch jednostajnie zmienny prostoliniowy).

Ruch punktu materialnego po okręgu i wielkości go opisujące (okres, prędkość kątowna i przyspieszenie dośrodkowe).

Ruch w polu grawitacyjnym (rzut pionowy, rzut poziomy, rzut ukośny i spadek swobodny).

Zasady dynamiki układów punktów materialnych (równia pochyła i opory ruchu).

Praca, moc, energia. Zasady zachowania w fizyce (zasada zachowania energii i pędu).

Drgania swobodne, drgania tłumione (wahadło matematyczne i masa na sprężynie).

Elementy termodynamiki (budowa materii, zmiany stanu skupienia oraz równanie Clapeyrona i przemiany gazowe).

Zasady optyki geometrycznej (prawo odbicia i załamania, konstrukcja obrazu – zwierciadło, soczewka oraz obraz rzeczywisty i pozorny).

Ruch falowy.

Elektryczność i magnetyzm (oddziaływania elektrostatyczne, ruch ładunku w polu elektrycznym i magnetycznym, prąd elektryczny)

Nazwa zajęć: **Systemy operacyjne 2**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. objaśnia metody działania i struktury systemów operacyjnych oraz modele jądra, jak również ich wpływ na rozwój technik informatycznych
2. objaśnia konieczność synchronizacji i planowania działań procesów oraz wsparcia sprzętowego dla efektywnego zaimplementowania wielozadaniowości oraz jej znaczenie
3. objaśnia planowanie działań przez systemy operacyjne dla procesów, kryteria, algorytmy i polityki ich szeregowania
4. objaśnia sposoby i algorytmy zarządzania zasobami pamięci komputera przez systemy operacyjne

w zakresie umiejętności:

1. potrafi posługiwać się narzędziami do programowania systemowego w zakresie obsługi plików i sygnałów, zarządzania procesami i komunikacją pomiędzy nimi

Treści programowe dla zajęć:

Podstawy działania systemów operacyjnych (metody działania, struktury, modele jądra, interfejs użytkownika, maszyny wirtualne, projektowanie, implementacja i generowanie systemu operacyjnego, procesy, wątki, rozwój technik informatycznych, stan systemu, ochrona i tryb pracy)

Wielozadaniowość systemów operacyjnych (błędy złej synchronizacji procesów, problem producent-hurtownia-konsument, blok kontrolny procesu, rola przerwań, model sekcji krytycznej, semaforey, monitory, głodzenie procesów, zakleszczenie, problemy czytelników i pisarzy oraz obiadujących filozofów)

Planowanie działań przez systemy operacyjne (procesy, polityka ich szeregowania, wymiana i wyłączenie, kryteria i algorytmy planowania procesów, ich dobór dla różnych zastosowań, systemy operacyjne czasu rzeczywistego)

Zarządzanie zasobami pamięci (metody zarządzania pamięcią, adresy logiczne i fizyczne, kod przemieszczalny, przydział ciągły i fragmentacja pamięci, segmentacja, stronicowanie, pamięć wirtualna; systemy plików, zarządzanie przestrzenią dyskową i planowanie dostępu)

Implementacje i porównania typowych systemów operacyjnych (rozwiązania kluczowych problemów w systemach operacyjnych MS Windows NT, UNIX i Linux; dobór systemu operacyjnego)

Programowanie systemowe (obsługa plików i sygnałów, zarządzanie procesami i komunikacja pomiędzy nimi)

Nazwa zajęć: **Fizyka komputerowa**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. Zna wybrane zagadnienia fizyki, dla których zrozumienia pożądane jest stosowanie metod obliczeniowych
2. Zna podstawowe techniki obliczeniowe wykorzystywane w fizyce obliczeniowej

w zakresie umiejętności:

1. Potrafi skorzystać z technik komputerowych pozwalających modelować badane zjawisko lub proces fizyczne

2. Potrafi stworzyć, przetestować i wykorzystać aplikację służącą do modelowania wybranego zagadnienia fizyki komputerowej.

Treści programowe dla zajęć:

Podstawowe techniki całkowania równań różniczkowych. Elementy dynamiki nieliniowej i chaotycznej. Błądzenie przypadkowe, gaz sieciowy, kinetyka sieciowych modeli reakcji chemicznych

Sieciowe modele wzrostu. Struktury fraktalne.

Modele perkolacyjne i algorytmy znajdowania klastrow

Dynamika populacji i modelowanie ekosystemów. Dylematy społeczne i modelowanie agentowe. Sieci złożone.

Przykłady zagadnień i technik optymalizacyjnych: algorytmy genetyczne i symulowane wyżarzanie.

Generatory liczb losowych i ich testowanie.

Techniki wizualizacyjne, obsługa zdarzeń i wielowątkowość w aplikacjach symulacyjnych.

Elementy algorytmiki: rekursja w perkolacyjnych modelach sieciowych

Projekt zaliczeniowy obejmujący napisanie aplikacji służącej do symulacji, wizualizacji i analizy wybranego problemu z fizyki komputerowej

Nazwa zajęć: **Technologie aplikacji internetu przyszłości**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. Zna i rozumie główne kierunki rozwoju Internetu.
2. Zna i rozumie mocne i słabe strony stosowania architektur systemów IT opartych na usługach.
3. Zna i rozumie podstawy języka XML i jego zastosowania.
4. Zna i rozumie metody i języki wspierające przetwarzanie i wizualizację danych zapisanych w formacie XML.
5. Zna i rozumie jak tworzyć aplikacje internetowe z wykorzystaniem REST API.
6. Zna i rozumie podstawy sieci semantycznych i obszary ich zastosowania.
7. Zna i rozumie proces tworzenia i modelowania procesów biznesowych.

w zakresie umiejętności:

1. Potrafi tworzyć i weryfikować poprawność utworzonych plików XML.
2. Potrafi tworzyć i weryfikować poprawność działania różnych metod wspierających przetwarzanie i wizualizację danych zapisanych w formacie XML.
3. Potrafi tworzyć różnego rodzaju elementy sieci semantycznych.
4. Potrafi analizować i tworzyć opisy procesów biznesowych.

w zakresie kompetencji społecznych:

1. Jest gotów/gotowa do rozwiązywania praktycznych zagadnień z modelowania procesów.
2. Jest gotów/gotowa do pełnienia roli informatyka ze znajomością kierunków rozwoju Internetu oraz świadomością zmian jakie mogą zajść w przyszłości.
3. Jest gotów/gotowa do uczestnictwa w projektach informatycznych z wykorzystaniem języka XML czy REST API.

Treści programowe dla zajęć:

Główne kierunkami rozwoju Internetu.

Architektury systemów IT opartych na usługach.

Język XML i jego zastosowania.

Metody i standardy wykorzystywane do przetwarzania danych zapisanych w formacie XML.

Wykorzystanie REST API do budowania aplikacji internetowych.

Podstawy sieci semantycznych i ich zastosowanie.

Tworzenie i modelowanie procesów biznesowych.

Nazwa zajęć: **Metody numeryczne i symulacje**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna i rozumie metody numeryczne
2. zna i rozumie zasady symulacji

w zakresie umiejętności:

1. potrafi zaprogramować obliczenia wykorzystujące metodę numeryczną
2. potrafi stworzyć symulację

Treści programowe dla zajęć:

Całkowanie i różniczkowanie numeryczne
Interpolacja i ekstrapolacja
Symulacje z wykorzystaniem generatorów zmiennych losowych
Przetwarzanie danych
Wizualizacja wyników

Nazwa zajęć: **Bezpieczeństwo usług typu hosting WWW**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Posiada wiedzę o architekturze i zasadach działania serwera WWW
2. Posiada wiedzę o protokołach komunikacji używanych przez serwery WWW (http, https)
3. Posiada wiedzę o architekturze bezpiecznego dostępu do stron WWW (certyfikaty, CA)
4. Posiada wiedzę o aplikacjach server-side, związanych z nimi ryzykami i sposobach zabezpieczenia
5. Posiada wiedzę o typach ataków na strony WWW i sposobach zabezpieczeń przed nimi

w zakresie umiejętności:

1. Posiada wiedzę i umiejętności konfiguracji oprogramowania serwera WWW (Apache, Nginx)
2. Posiada wiedzę i umiejętności konfiguracji kontroli dostępu do usług WWW
3. Posiada wiedzę i umiejętności konfiguracji przekształceń adresu URL na poziomie serwera WWW (mod_rewrite)
4. Posiada wiedzę i umiejętności konfiguracji oprogramowania typu firewall
5. Posiada wiedzę i umiejętności bezpiecznej konfiguracji maszyn wirtualnych

Treści programowe dla zajęć:

Podstawowe informacje o architekturze usługi WWW
Protokół HTTP (request i response, podstawowe pola nagłówka, sygnalizacja błędów)
Protokół HTTPS i związana z nim infrastruktura (certyfikaty, podpisy, CER, CA)
Podstawowe informacje o serwerze WWW Apache
Podstawowa konfiguracja Apache (dyrektywy core, dyrektywy blokowe, system modułów Apache'a)
Manipulowanie adresem URL na poziomie serwera WWW – mod_rewrite
Aplikacje server side na przykładzie PHP
Typowe ataki na serwer WWW (XSS, injection, DoS) i sposoby przeciwdziałania
Zabezpieczenie serwera WWW przy pomocy firewalla
Podstawy wirtualizacji
Hosting na maszynach wirtualnych na przykładzie Virtualbox

Nazwa zajęć: **Teoria sterowania**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. wie jakie są zasady sterowania i regulacji.
2. wie jakie są podstawowe obiekty automatyki oraz regulatory i tym samym jakie można tworzyć układy automatycznej regulacji (UAR).
3. wie jak dokonać podstawowej identyfikacji obiektu oraz w jaki sposób zamodelować układy automatycznej regulacji i dokonać ich oceny jakości regulacji.
4. wie jaki jest podział i podstawowe składowe hierarchicznego systemu sterowanie (SCADA/BMS).

w zakresie umiejętności:

1. potrafi posługiwać się pakietem/językiem Matlab oraz programem do modelowania i symulacji Simulink.
2. potrafi zaprojektować/zamodelować określone układy regulacji/sterowania oraz dokonać ich oceny jakości regulacji używając co najmniej kilku kryteriów.
3. potrafi wyznaczać modele złożonych systemów i procesów, a także wykorzystywać je do celów analizy i projektowania układów automatyki.
4. potrafi formułować i weryfikować symulacyjnie hipotezy związane z zadaniami inżynierskimi i trudnymi problemami badawczymi z zakresu automatyki
5. potrafi dokonać krytycznej analizy sposobu funkcjonowania systemów sterowania.

w zakresie kompetencji społecznych:

1. ma świadomość konieczności połączenia wiedzy matematycznej z wiedzą techniczną do profesjonalnego rozwiązywania problemów technicznych.
2. ma umiejętność pracy w zespole.
3. ma umiejętność definiowania ogólnych problemów technicznych i ich rozwiązywania z wykorzystaniem nowoczesnych narzędzi informatycznych.

Treści programowe dla zajęć:

Wstęp do automatyki i sterowania. Wprowadzenie pojęć: obiekt i regulator.
Matematyczny opis układu regulacji (przekształcenie Laplace'a, transmitancja operatorowa).
Podstawowe elementy automatyki: inercyjny 1-go rzędu, całkujący idealny, całkujący z inercją, różniczkujący idealny, różniczkujący z inercją, oscylacyjny, inercyjny 2-go rzędu, inercyjny n-go rzędu, opóźniający.
Klasyfikacja układów automatyki: regulacja, regulatory. Podział układów automatycznej regulacji.
Metody identyfikacji obiektów. Badanie stabilności układów.
Charakterystyki układów regulacji i ich korekta. Badanie odpowiedzi na standardowe pobudzenia.
Metody oceny jakości regulacji - bezpośrednie i pośrednie.
Metody doboru parametrów regulatorów - badanie układów automatycznej regulacji (UAR).
Regulacja dyskretna i ciągła
Hierarchiczne systemy sterowania - automatyka przemysłowa (SCADA) oraz automatyka budynkowa (BMS)
Pakiet/Język Matlab - od korespondencji w wierszu poleceń do pisanie funkcji i skryptów z wykorzystaniem modułów graficznych.
Program Simulink - czyli od modelowania prostych UAR po złożone symulacje i analizy.

Nazwa zajęć: **Praktyki zawodowe**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna i rozumie podstawowe zagadnienia dotyczące oprócz klasycznych zagadnień informatyki i fizyki, zagadnienia takie jak cyfrowe przetwarzanie danych, elektronika analogowa i cyfrowa, telekomunikacja i automatyka, inżynieria systemów informatycznych, oprogramowania i sieci, przez usługi internetowe po technologie wirtualnej rzeczywistości i potrafi je zastosować w pracy zawodowej.

w zakresie umiejętności:

1. zna i rozumie sposób funkcjonowania Zakładu pracy, jego strukturę organizacyjną, zasady pracy, zadania realizowane przez Zakład pracy oraz obowiązujące w Zakładzie pracy przepisy wewnętrzne.

w zakresie kompetencji społecznych:

1. nabywa kompetencji społecznych związanych z pracą zawodową w zakresie organizacji własnego czasu pracy, jak i pracy w zespole.

2. jest przygotowany/a do podejmowania przyszłych decyzji zawodowych zgodnych z własnymi zainteresowaniami.

Treści programowe dla zajęć:

Zastosowanie w praktyce wiedzy zdobytej na studiach, pozwalającej na wykonywanie powierzonych zadań. Identyfikowanie problemów i zastosowanie właściwych technologii, procedur, narzędzi oraz dostępnych źródeł informacji w danym Zakładzie pracy w celu ich rozwiązania.

Zapoznanie się ze sposobem funkcjonowania Zakładu pracy, jego strukturą organizacyjną, zasadami pracy, zadaniami realizowanymi przez Zakład pracy, obowiązującymi w Zakładzie pracy przepisami wewnętrznymi.

Realizowanie powierzonych zadań przy jednoczesnym podnoszeniu własnych kwalifikacji. Asystowanie i pomaganie w zespołowych pracach Zakładu pracy, wykazywanie odpowiedzialności za ich prawidłowe i bezpieczne wykonanie, rzetelne realizowanie powierzonych zadań, dbanie o powierzony sprzęt i bezpieczeństwo pracy.

Nabywanie wiedzy o funkcjonowaniu Zakładu pracy celem podejmowania w przyszłości decyzji zawodowych zgodnych z własnymi zainteresowaniami.

Nazwa zajęć: **Zajęcia wyrównawcze z matematyki**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna i rozumie zagadnienia z matematyki na poziomie rozszerzonym

w zakresie umiejętności:

1. potrafi biegle posługiwać się wyrażeniami algebraicznymi

2. potrafi rozwiązywać zadania związane z funkcją; biegle włada zagadnieniami związanymi z funkcją liniową, kwadratową, wykładniczą i logarytmiczną; potrafi użyć tej wiedzy do rozwiązywania problemów z fizyki obejmujących pierwsze trzy lata studiów

3. potrafi wykorzystywać trygonometrię, planimetrię oraz stereometrię na poziomie rozszerzonym matematyki

4. potrafi posługiwać się rachunkiem wektorowym w stopniu pozwalającym przyswajać zagadnienia z fizyki obejmujące I. stopień kształcenia
5. potrafi korzystać z pochodnej funkcji; potrafi obliczyć pochodną funkcji oraz zastosować rachunek różniczkowy w fizyce
6. potrafi sprawnie wykonywać operacje na wielomianach

Treści programowe dla zajęć:

Rachunek wektorowy, elementy geometrii analitycznej
Wyrażenia algebraiczne
Ogólne pojęcia związane z funkcją
Funkcja liniowa
Funkcja kwadratowa
Funkcja wykładnicza i logarytmiczna
Wielomiany
Trygonometria
Planimetria
Stereometria
Wstęp do rachunku różniczkowego

Nazwa zajęć: **Wprowadzenie do Sztucznej Inteligencji i Rzeczywistości Cyfrowych.**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna i rozumie podstawową terminologię, koncepcje i obszary zastosowań sztucznej inteligencji (SI) i rzeczywistości cyfrowych (RC).
2. zna i rozumie typy narzędzi SI stosowanych do uczenia się, produktywności i badań.
3. zna i rozumie podstawowe koncepcje sztucznych sieci neuronowych (SSN), funkcji aktywacji i podstawowych mechanizmów uczenia.
4. zna i rozumie jak wykorzystać standardowe biblioteki do definiowania i trenowania prostych modeli SSN.
5. zna i rozumie rolę komponentów sprzętowych w kontekście wydajności procesów SI.
6. zna i rozumie metody konfiguracji środowisk oraz komunikacji z modelami i usługami SI.
7. zna i rozumie możliwości i ograniczenia wspomaganie kodowania przez SI.
8. zna i rozumie metody i architektury tworzenia rozwiązań SI oraz potrafi zidentyfikować cechy głównych typów modeli SI.
9. zna i rozumie zasady generowania zasobów wspomaganego przez SI oraz praktyki zarządzania wersjami.

w zakresie umiejętności:

1. potrafi wykorzystywać inteligentne narzędzia do usprawniania własnego uczenia się i przetwarzania informacji.
2. potrafi zaimplementować, wytrenować i ocenić prosty model SSN, używając standardowe biblioteki.
3. potrafi ocenić podstawową adekwatność sprzętu do typowych lokalnych zadań SI.
4. potrafi skonfigurować środowiska programistyczne odpowiednie do pracy z SI i RC.
5. potrafi komunikować się z usługami/modelami SI i wybierać odpowiednie modele do prostych zadań.
6. potrafi stosować asystentów kodowania SI do generowania, debugowania lub wyjaśniania kodu.
7. potrafi generować podstawowe media wizualne/dźwiękowe za pomocą SI i integrować je w prostym prototypie RC.
8. potrafi stosować podstawowe operacje zarządzania wersjami.
9. potrafi pracować indywidualnie i zespołowo

w zakresie kompetencji społecznych:

1. jest gotów/gotowa do samokształcenia przy użyciu różnorodnych zasobów cyfrowych i wspomaganych przez SI.
2. jest gotów/gotowa do efektywnej komunikacji i pracy w środowiskach cyfrowych.
3. jest gotów/gotowa do dokumentowania procesów i przejrzystego prezentowania wyników pracy.
4. jest gotów/gotowa do uczenia się przez całe życie w dynamicznie rozwijających się dziedzinach SI i RC.
5. jest gotów/gotowa do efektywnej pracy przy użyciu narzędzi cyfrowych.
6. jest gotów/gotowa do eksperymentowania i oceny nowych narzędzi SI i technik programistycznych.
7. jest gotów/gotowa do adaptacji rozwiązań technicznych w odpowiedzi na rozwój SI.
8. jest świadomy/świadoma kosztów wydajności i implikacji związanych z różnymi architekturami SI.

Treści programowe dla zajęć:

Wprowadzenie do sztucznej inteligencji (SI) i rzeczywistości cyfrowych (RC)

Aktualne rozwiązania SI – Narzędzia, wzorce i przetwarzanie
Przegląd modeli SI – Analiza architektur i możliwości
Wykorzystanie systemów inteligentnych do zwiększania produktywności
Wymagania sprzętowe dla lokalnej budowy i wdrażania rozwiązań SI
Podstawy programowania SI/RC
Praktyczne trenowanie modeli SI – Środowiska pracy i biblioteki
Komunikacja z modelami i usługami SI - API i platformy
Zasady rozwoju oprogramowania w zespole i zarządzania wersjami
Przygotowanie środowisk programistycznych dla realizacji SI i RC
Wspomagane przez SI generowanie treści wizualnych i zasady integracji
Wspomagane przez SI generowanie treści dźwiękowych i zasady integracji
Kodowanie i prototypowanie wspomagane przez SI
Sieci neuronowe i procesy uczenia
Projekt - Tworzenie prototypu aplikacji RC zintegrowanej z SI

Nazwa zajęć: **Serwery baz danych**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. poznaje zasady związane z pracą z serwerami baz danych na przykładzie wybranych serwerów. Instalacja serwera. Tworzenie baz. Administracja użytkowników, backupów etc. Tworzenie stron internetowych w połączeniu z bazami danych.

w zakresie umiejętności:

1. instaluje przykładowe serwery (MS SQL Server, Oracle, Firebird, MySQL ew. inne)
2. pracuje z konsolą administracyjną. Prowadzi administrację użytkowników. Nadaje uprawnienia etc. Administruje bazami: backupy, protokoły, logi, analiza ruchu, optymalizacja obciążeń.
3. instaluje serwery www (Apache, NGINX)
4. tworzy strony www (ew. aplikacje webowe) współpracujące z serwerami SQL'owymi. tzw. backend.
Np. MySQL

Treści programowe dla zajęć:

Omówienie zagadnień związanych z pracą z serwerami baz danych na przykładzie wybranych serwerów. Kwestie instalacji, administrowania i konfiguracji.

Łącuchy połączeniowe. Współpraca z aplikacjami korzystającymi z serwerów. Omówienie narzędzi do tworzenia stron i popularnych bibliotek,

Nazwa zajęć: **Wstęp do metrologii, metod probabilistycznych i statystyki**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna pojęcie modelu matematycznego prostego zjawiska oraz rozumie wpływ zdeterminowanego i probabilistycznego modelu błędu na jakość tego modelu.
2. zna i rozumie podstawowe pojęcia teorii prawdopodobieństwa: zdarzenia i zmienne losowe, prawdopodobieństwo, rozkład zmiennej losowej dyskretnej i ciągłej
3. zna podstawowe metody statystyczne stosowane w metrologii.

w zakresie umiejętności:

1. potrafi rozwiązywać problemy stosując poznane metody rachunku prawdopodobieństwa.
2. potrafi posługiwać się wybranymi metodami statystycznymi: estymacją i testami statystycznymi w problemach związanych z metrologią.
3. potrafi krytycznie stosować odpowiednie metody statystyczne oraz analizować otrzymane wyniki.

Treści programowe dla zajęć:

Obserwacja i pomiar, typy skal i układy jednostek pomiarowych.

Model matematyczny, niedokładność i nieadekwatność modelu, deterministyczny i probabilistyczny model błędu i niepewności pomiarowej.

Zdarzenia losowe, definicje prawdopodobieństwa, prawdopodobieństwo warunkowe, całkowite, twierdzenie Bayes'a.

Jednowymiarowa zmienna losowa dyskretna i ciągła, dystrybuenta, funkcje zmiennej losowej.

Charakterystyki liczbowe zmiennych losowych dyskretnych i ciągłych.

Wybrane rozkłady prawdopodobieństwa: dwumianowy, Poissona, Maxwella, normalny.

Twierdzenia graniczne i prawa wielkich liczb.

Estymacja przedziałowa wartości oczekiwanej i wariancji, minimalna liczebność próby.

Parametryczne testy istotności dla wartości przeciętnej i wariancji (próba pojedyncza, dwie próby)

Regresja liniowa.

Nazwa zajęć: Bezpieczeństwo informacji

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Zna i rozumie pojęcia bezpieczeństwa i jego roli w organizacji.
2. Zna i rozumie modele ochrony i ataków na bezpieczeństwo informacyjne.
3. Zna i rozumie cechy informacji jakie podlegają ochronie w zakresie bezpieczeństwa informacji.
4. Zna i rozumie wybrane metody analizy zagrożeń.
5. Zna i rozumie wybrane metody prognozowania zagrożeń i ryzyka.
6. Zna i rozumie zakres i zastosowanie wybranych norm z serii ISO 27001 oraz polityki bezpieczeństwa informacji.
7. Zna i rozumie wybrane elementy i rozwiązania zapewniające bezpieczeństwo informacji.

w zakresie umiejętności:

1. Potrafi zidentyfikować i przeanalizować określone zagrożenia dla bezpieczeństwa informacji.
2. Potrafi zastosować w praktyce wybrane metody prognozowania zagrożeń i ryzyka.
3. Potrafi przygotować i omówić wybrane zabezpieczenia z normy ISO 27001.

w zakresie kompetencji społecznych:

1. Jest gotów/gotowa do pogłębiania wiedzy i przedstawiania w sposób zrozumiały wiedzy z zakresu bezpieczeństwa informacji.
2. Jest gotów/gotowa do pełnienia roli informatyka ze świadomością ciągłych zmian w obszarze bezpieczeństwa informacji.
3. Jest gotów/gotowa do rozumienia roli zgodności z regulacjami oraz postępowania etycznego.

Treści programowe dla zajęć:

Wprowadzenie do bezpieczeństwa w organizacji i bezpieczeństwa informacji.

Omówienie metod identyfikacji i analizy zagrożeń i ryzyka.

Przedstawienie przykładowych modeli ochrony, modeli ataków oraz kategorii rozwiązań zwiększających bezpieczeństwo informacji.

Przedstawienie zagadnień związanych z wybranymi normami z serii ISO27000.

Stosowanie wybranych elementów i rozwiązań zwiększających bezpieczeństwo informacji.

Nazwa zajęć: Elementy nietechniczne w pracy inżyniera

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Rozumie rolę negocjacji w życiu codziennym, zarówno w sytuacjach zawodowych jak i niezawodowych, niskiego oraz wysokiego szczebla. Ma uporządkowaną wiedzę na temat etapów negocjacji, gamy możliwych sposobów działania oraz ich interpretacji.
2. Posiada wiedzę na temat cech, jakie aspekty działania odróżniają ludzi działających skutecznie od pozostałych wg metodyki Covey'a.
3. Ma wiedzę na temat mechanizmów realizacji procedury szukania pracy, w tym rozmowy kwalifikacyjnej. Rozumie poszczególne etapy tej procedury oraz ich znaczenie
4. Ma wiedzę w zakresie podstawowych zasad savoir-vivre, zarówno w sytuacjach zawodowych jak i prywatnych. Rozumie rolę zasad savoir vivre w życiu codziennym.

w zakresie umiejętności:

1. Potrafi podać i prawidłowo zinterpretować przykłady negocjacji w życiu codziennym, prywatnym oraz zawodowym oraz ocenić ich zgodność z zaleceniami dotyczącymi procesu negocjacji
2. Potrafi podać i prawidłowo zinterpretować różne aspekty skutecznego działania na bazie własnych oraz cudzych przykładów a także zastosować je do własnych działań
3. Potrafi podać i prawidłowo zinterpretować różne aspekty procesu szukania pracy na bazie własnych oraz cudzych przykładów a także zastosować je do własnych działań
4. Potrafi podać i prawidłowo zinterpretować różne aspekty reguł savoir-vivre na bazie własnych oraz cudzych przykładów a także zastosować je do własnych działań

w zakresie kompetencji społecznych:

1. Posiada gotowość do uczestniczenia w pracy zawodowej w kontekście wielorakich jej aspektów dotyczących działania organizacji, interpretowania zachowań ludzkich, pracy w zespole i realizacji celów osobistych i zespołowych.

Treści programowe dla zajęć:

Mity o negocjacjach, negocjacje w trybie: wygrana-wygrana, etapy negocjacji, przygotowanie, stawianie celów, utrzymywanie emocjonalnego dystansu, aktywne słuchanie, finalizowanie negocjacji, najczęstsze błędy

Rola proaktywności, stawianie celów strategicznych a realizacja taktyki, sprawy ważne a pilne, delegowanie zadań, tworzenie sytuacji: wygrana-wygrana, skuteczna komunikacja, syndrom ostrzeżenia pily

Szukanie pracy jako sprzedaż, rola sprzedaży w gospodarkach konkurencyjnych, szukanie pracy jako proces dołączania do grupy, etapy szukania pracy, materiały marketingowe w procesie szukania pracy, rola i główne elementy rozmowy kwalifikacyjnej, typowe błędy

Zasady ogólne, przedstawianie się, zasady starszeństwa, mówienie sobie po imieniu, zasady ubioru biznesowego, elementy zachowania się przy posiłkach

Nazwa zajęć: Teleinformatyka

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. Zna i rozumie takie pojęcia jak system informacyjny, system informatyczny, system teleinformatyczny.
2. Zna i rozumie podstawowe elementy architektury systemów informatycznych.
3. Zna i rozumie różnice pomiędzy bazami relacyjnymi i nierelacyjnymi.
4. Zna i rozumie podstawy tworzenia rozwiązań informatycznych z wykorzystaniem platform wirtualizacyjnych.
5. Zna i rozumie zastosowanie narzędzi tworzących infrastrukturę sprzętowo-programową wspierającą tworzenie rozwiązań informatycznych.
6. Zna i rozumie mechanizmy zapewnienia jakości i interoperacyjności w działaniu sieci.
7. Zna i rozumie zagadnienia związane z niezawodnością i podstawami projektowania sieci teleinformatycznych.

w zakresie umiejętności:

1. Potrafi utworzyć i wykonywać podstawowe operacje na nierelacyjnej bazie danych.
2. Potrafi przygotować rozwiązanie informatyczne z wykorzystaniem platform wirtualizacyjnych.
3. Potrafi zastosować wybrane narzędzia wspierające tworzenie rozwiązań informatycznych w pracy zespołowej.
4. Potrafi zastosować podstawowe działania związane z projektowaniem sieci teleinformatycznych.

w zakresie kompetencji społecznych:

1. Jest gotów/gotowa do zrozumienia roli wspierającej prace informatyczne w organizacji.
2. Jest gotów/gotowa do wsparcia procesów w organizacjach wytwarzających oprogramowanie.

Treści programowe dla zajęć:

Informacje podstawowe z zakresu systemów teleinformatycznych.

Relacyjne i nierelacyjne bazy danych.

Wirtualizacja i środowiska wirtualizacyjne.

Zapewnienie jakości i interoperacyjności w sieciach teleinformatycznych.

Narzędzia i platformy wspierające zespołowe wytwarzanie oprogramowania.

Aspekty niezawodności i projektowania sieci teleinformatycznych.

Nazwa zajęć: Podstawy przedsiębiorczości i etyka

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. Zna i rozumie znaczenie pożądanых cech osób przedsiębiorczych oraz główne aspekty prowadzenia działalności gospodarczej.
2. Zna i rozumie aspekty ryzyka oraz innowacji w działalności gospodarczej.
3. Zna i rozumie aspekty zakładania spółki oraz różnych form prowadzenia działalności gospodarczej.
4. Zna i rozumie różne formy zatrudnienia.
5. Zna i rozumie różne struktury organizacyjne w firmie.
6. Zna i rozumie aspekty inwestowania i tworzenia nowych produktów i usług.
7. Zna i rozumie podstawy działania firmy na rynku konkurencyjnym.
8. Zna i rozumie podstawy organizacji i procesu sprzedaży w firmie.

w zakresie umiejętności:

1. Potrafi przygotować i przedstawić model biznesowy dla nowego produktu lub usługi.
2. Potrafi przygotować raport oraz wziąć aktywny udział w dyskusji na tematy związane z elementami przedsiębiorczości.

w zakresie kompetencji społecznych:

1. Jest gotów/gotowa do oceny uczciwości działań oraz postępowania etycznego w prowadzeniu działalności gospodarczej.

2. Jest gotów/gotowa do prowadzenia dyskusji w zakresie elementów przedsiębiorczości oraz korzystania z opinii ekspertów.

Treści programowe dla zajęć:

Główne cechy osób przedsiębiorczych i ich analiza.
Innowacje i podpatrywanie pomysłów w działalności gospodarczej.
Aspekty ryzyka i głównych czynników związanych z działalnością gospodarczą.
Formy prowadzenia działalności gospodarczej i zakładanie spółki.
Formy zatrudnienia oraz podstawowe informacje związane z podatkami.
Struktury organizacyjne w firmie.
Elementy inwestycji i tworzenia nowych produktów lub usług.
Działanie na rynku konkurencyjnym.
Organizacja i działanie sprzedaży w firmie.

Nazwa zajęć: **Urządzenia mobilne w praktyce pomiarowej: Pracownia systemów pomiarowych, Programowanie urządzeń mobilnych**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Wie jak zaprojektować interfejs pomiędzy urządzeniem mobilnym a układem pomiarowym.
2. Wie jak przeprowadzać podstawowe symulacje obwodów elektrycznych.
3. Wie jak posługiwać się oprogramowaniem do projektowania obwodów drukowanych.
4. Wie jak przygotować dokumentację wykonawczą do zaprojektowanych obwodów drukowanych.
5. Wie jak zmontować i uruchomić zaprojektowany układ pomiarowy.
6. Wie jak posługiwać się oprogramowaniem do tworzenia aplikacji mobilnych.
7. Wie jak tworzyć interfejs użytkownika.
8. Wie jak tworzyć oprogramowanie do komunikacji z układami peryferyjnymi (sensorami).
9. Wie jak tworzyć oprogramowanie do przetwarzania i analizowania danych pomiarowych.
10. Wie jak połączyć zaprojektowany i zbudowany interfejs pomiarowy z napisanym oprogramowaniem.

w zakresie umiejętności:

1. Umie zaprojektować interfejs pomiędzy urządzeniem mobilnym a układem pomiarowym.
2. Umie przeprowadzać podstawowe symulacje obwodów elektrycznych.
3. Umie posługiwać się oprogramowaniem do projektowania obwodów drukowanych.
4. Umie przygotować dokumentację wykonawczą do zaprojektowanych obwodów drukowanych.
5. Umie zmontować i uruchomić zaprojektowany układ pomiarowy.
6. Umie posługiwać się oprogramowaniem do tworzenia aplikacji mobilnych.
7. Umie tworzyć interfejs użytkownika.
8. Umie tworzyć oprogramowanie do komunikacji z układami peryferyjnymi (sensorami).
9. Umie tworzyć oprogramowanie do przetwarzania i analizowania danych pomiarowych.
10. Umie połączyć zaprojektowany i zbudowany interfejs pomiarowy z napisanym oprogramowaniem.

Treści programowe dla zajęć:

Interfejsy pomiarowe, czujniki, przetwarzanie A/D D/A, próbkowanie, dopasowanie, impedancja, linie transmisyjne.

Symulacje obwodów elektrycznych z wykorzystaniem darmowego oprogramowania. Zasady projektowania obwodów drukowanych EDA/CAD/CAM. Projekt interfejsu pomiarowego. Przygotowanie dokumentacji wykonawczej do obwodów PCB.

Zasady montażu układów elektronicznych THT, SMD. Sposoby uruchamiania i testowania zmontowanych urządzeń elektronicznych. Kalibracja.

Programowanie JAVA, Android, zasady projektowania oprogramowania, symulacje parametrów fizycznych, przetwarzanie danych, algorytmy, biblioteki matematyczne, odsumianie, kalibracja, wizualizacja wyników, przygotowanie dokumentacji.

Nazwa zajęć: **Seminarium dyplomowe**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna i rozumie najnowsze osiągnięcia naukowe, metody badawcze i technologie dotyczące tematu badawczego w ramach realizowanej pracy inżynierskiej
2. zna metody analizy danych właściwych dla rozwiązywanego zagadnienia.

w zakresie umiejętności:

1. potrafi przygotować i wygłosić referat na zadany temat związany z tematyką pracy inżynierskiej.
2. potrafi korzystać ze źródeł literaturowych, także w języku angielskim, czyta ze zrozumieniem tekst fachowy oraz naukowy dotyczący tematyki pracy.

3. potrafi przygotować prezentację przedstawiającą koncepcję projektu/teorię/koncepcję badań, aktualny stan wiedzy, metody badawcze, dostępne technologie i metody oraz potrafi przeprowadzić interpretację i dyskusję otrzymanych wyników.

w zakresie kompetencji społecznych:

1. jest gotów/gotowa do krytycznej oceny posiadanej wiedzy i dyskusji na temat prowadzonych badań.
2. jest gotów/gotowa do prowadzenia dyskusji na temat etyki zawodowej badacza

Treści programowe dla zajęć:

sposoby prezentacji (przygotowanie i wygłoszenie prezentacji) oraz prowadzenie profesjonalnej/naukowej dyskusji.

metodyka przygotowania fachowych/naukowych opracowań, w tym pracy inżynierskiej.

problematyka etyki i plagiatu w badaniach i opracowaniach naukowych, np. w pracach dyplomowych, artykułach naukowych, cytowanie źródeł.

Nazwa zajęć: **Technologie handlu elektronicznego e-commerce**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna zagadnienia dotyczące marketingu cyfrowego.
2. zna zagadnienia dotyczące strategii, taktyk i znaczenia budowy i promocji osobistej marki.
3. zna zagadnienia dotyczące doboru strategii, rodzaju marketingu i typu kampanii marketingowej.
4. zna zagadnienia dotyczące skuteczności różnych strategii związanych z doбором platform organicznego marketingu internetowego.
5. zna zagadnienia dotyczące skuteczności różnych strategii związanych z doбором i integracją płatnych platform marketingu internetowego.
6. zna zagadnienia dotyczące organicznego i płatnego marketingu w mediach społecznościowych w porównaniu z marketingiem w wyszukiwarkach internetowych.
7. zna zagadnienia dotyczące programowania i rozwiązań sztucznej inteligencji wspierających działania marketingowe.
8. zna zagadnienia dotyczące badań, audytu, monitorowania i analizy konkurencyjności firmy i działań rynkowych konkurencji.
9. zna zagadnienia dotyczące personalizacji i badań segmentacji rynku.
10. zna zagadnienia dotyczące planowania i implementacji zintegrowanej sieci pomiaru efektywności i monitorowania kampanii marketingowych.
11. zna zagadnienia dotyczące podejmowania inteligentnych decyzji oraz integracji narzędzi raportowania, wizualizacji danych historycznych i czasu rzeczywistego wspomagających proces wnioskowania i podejmowania inteligentnych decyzji.
12. zna zagadnienia dotyczące optymalizacji on-page, on-site i off-site oraz doboru typu i formatu treści dla różnych typów kampanii marketingowych.
13. zna zagadnienia dotyczące doboru i skuteczności metod kierowania przekazu marketingowego dla różnych platform, kanałów i typów kampanii marketingowych.
14. zna zagadnienia związane ze strategiami, taktykami oraz skutecznością prowadzenia kampanii remarketingowych.
15. zna zagadnienia dotyczące audytów, monitorowania, optymalizacji i rozszerzania obszarów skuteczności kampanii marketingowych.
16. zna zagadnienia dotyczące planowania i prowadzenia kampanii budowy zaufania i autorytetu marki oraz zarządzania reputacją.

w zakresie umiejętności:

1. potrafi promować osobistą markę.
2. potrafi dobrać rodzaj i typ kampanii marketingowej do wybranych strategii.
3. potrafi dobrać odpowiednie strategie i platformy organicznego marketingu internetowego.
4. potrafi dobrać odpowiednie strategie i platformy płatnego marketingu internetowego oraz integrować je w spójny system analityczny i zarządzania kampaniami online.
5. potrafi zaplanować i prowadzić kampanie organiczne i płatne w mediach społecznościowych oraz w wyszukiwarkach internetowych.
6. potrafi stworzyć własne rozwiązania programistyczne i wykorzystać rozwiązania sztucznej inteligencji w celu wsparcia działań marketingowych.
7. potrafi przeprowadzić badania, audyt, monitorować i przeanalizować konkurencyjność firmy i działania rynkowe konkurencji.
8. potrafi przeprowadzić badania segmentacji rynku w celu personalizacji komunikacji marketingowej.
9. potrafi zaplanować i zaimplementować zintegrowaną sieć pomiaru efektywności i monitorowania kampanii marketingowych.

10. potrafi integrować i wykorzystać narzędzia raportowania, wizualizacji danych historycznych i czasu rzeczywistego w procesie wnioskowania i podejmowania decyzji.
11. potrafi zaplanować i wykonać optymalizację on-page, on-site i off-site dobierając odpowiednie typy i formaty treści.
12. potrafi dobrać i skonfigurować skuteczne metody kierowania dla odpowiedniego przekazu marketingowego dla różnych platform, kanałów i typów kampanii marketingowych.
13. potrafi zaplanować i przeprowadzić kampanię remarketingową.
14. potrafi przeprowadzić audyt, monitorować, optymalizować i rozszerzyć obszary skuteczności kampanii marketingowych.
15. potrafi zaplanować i przeprowadzić kampanię budowy zaufania i autorytetu marki oraz zarządzać jej reputacją.

w zakresie kompetencji społecznych:

1. jest gotowy/a do globalnych wyzwań, rynku pracy i budowania profesjonalnej sieci, łącząc się z ekspertami i innymi studentami z branży marketingu cyfrowego.
2. jest gotowy/a do budowania marki osobistej i tworzenia własnego portfolio.
3. jest gotowy/a do uczenia się przez całe życie i efektywnego rozwoju osobistego w stale rozwijającej się branży marketingu cyfrowego.
4. jest gotowy/a do efektywnej pracy indywidualnej i współpracy w zespole.
5. jest gotowy/a pełnić rolę lidera w projektach zespołowych.
6. jest gotowy/a do efektywnego dostosowywania się do zmieniających się wymagań, nowych narzędzi i pojawiających się trendów w zakresie marketingu cyfrowego.

Treści programowe dla zajęć:

Wprowadzenie do marketingu cyfrowego - znaczenie budowy i promocji osobistej marki - strategię i taktyki.

Rodzaje marketingu i typy kampanii - porównanie i dobór strategii w inteligentnej erze wielokanałowej. Przegląd typów platform organicznego marketingu internetowego i strategii ich skutecznego wykorzystania.

Przegląd płatnych platform marketingowych - integracja w konstrukcji zintegrowanego systemu marketingowego i strategię ich skutecznego wykorzystania.

Organiczny i płatny marketing społecznościowy, a marketing w wyszukiwarkach internetowych.

Programowanie i rozwiązania sztucznej inteligencji wspierające działania marketingowe.

Badania, audyt, monitorowanie i analiza konkurencyjności firmy i działań rynkowych konkurencji.

Personalizacja i badanie segmentacji rynku - analiza charakterystyk i zachowań docelowych grup odbiorców.

Planowanie i implementacja zintegrowanej sieci pomiaru efektywności i monitorowanie kampanii marketingowych - narzędzia analityczne do agregacji danych i zarządzania tagami (identyfikacja KPIs mikro i makro konwersji etc.).

Podejmowanie inteligentnych decyzji - integracja narzędzi raportowania, wizualizacji danych historycznych i czasu rzeczywistego.

Optymalizacja on-page, on-site i off-site - dobór typu i formatu treści dla różnych typów kampanii marketingowych.

Dobór i skuteczność metod kierowania przekazu marketingowego dla różnych platform, kanałów i typów kampanii marketingowych.

Strategie, taktyki oraz skuteczność kampanii remarketingowych - cyfrowy dialog i negocjacje z klientami.

Audyt, monitorowanie, optymalizacja i rozszerzanie obszarów skuteczności kampanii marketingowych.

Zarządzanie reputacją marki - kampanie budowy zaufania i autorytetu. [W1,3,7,16 U2,6,15 K1,,3,4,5,6]

Nazwa zajęć: **Bazy danych / SQL**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna: Podstawowe pojęcia teorii baz danych. Związki. Model związków encji (entity-relationship model, E-R model). Modele: relacyjny, hierarchiczny sieciowy. Podstawy teoretyczne - algebra relacji. Terminologię.
2. zna: Fizyczną reprezentację danych. Dyski. Powiązanie z systemem operacyjnym. Koncepcję architektury baz lokalnych. Koncepcję architektury klient-serwer i baz rozproszonych.
3. zna: SQL (Structured Query Language). Zna: Logiczny model danych. Definicje obiektów bazodanowych: Baza danych (database). Tabela (table). Wiersz (row). Kolumna (column). Klucz główny (primary key). Klucz obcy (foreign key) Klucz potencjalny (candidate key). Złączenie

(join). Więzy (constraint). Indeks (index). Perspektywa (view). Procedura pamiętana (stored procedure). Procedura zdarzeń (trigger).

4. zna zagadnienia związane z: Klasycznymi problemami izolowania transakcji. Zarządzaniem transakcjami. Poziomem izolowania transakcji w DBMS. Sterowaniem transakcjami z wykorzystaniem języka SQL. Izolowaniem transakcji na serwerze. Wyborem odpowiedniego poziomu izolowania transakcji. Zna zasady normalizacji baz danych.

w zakresie umiejętności:

1. Tworzenie nowych baz danych (schem). Tworzenie nowych tabel. Typy danych. Views (perspektyw). Wpisywanie danych. Instrukcja SELECT. Wybieranie warunkowe. Operatory relacji. Warunki złożone. Operatory: AND, OR. Klauzule IN i BETWEEN. Instrukcja LIKE. Złącza (joins). Klucze (keys). Instrukcja DISTINCT i eliminacja powtórzeń. Aliasy i podzapytania. Funkcje sumaryczne (agregacje). Modyfikacja tabel. Indeksy. Instrukcje GROUP BY, ORDER BY, HAVING, EXISTS, ALL, UNION. Łączy zewnętrzne. Transakcje. Tworzenie procedur pamiętanych (stored procedures) i procedur zdarzeń (triggers). Usuwanie danych. Modyfikacja i usuwanie obiektów bazy.

2. Opanowanie pracy z konsolą serwera bazy danych.

Treści programowe dla zajęć:

Podstawy teoretyczne. Praktyczne implementacje systemów bazodanowych. Bazy lokalne. Bazy w architekturze klient serwer.

Podstawy języka SQL (Structured Query language). Definicje obiektów bazodanowych. Terminologia wykorzystywana w praktyce tworzenia baz danych. Sterowanie transakcjami z wykorzystaniem języka SQL.

Nazwa zajęć: **Projekt informatyczny**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. zna zasady prowadzenia projektów

w zakresie umiejętności:

1. potrafi przygotować projekt

2. potrafi dobrać rozwiązania do postawionego problemu technicznego

3. potrafi dobrać technologie do komponentów wdrażanego systemu

4. potrafi planować i testować rozwiązania

5. potrafi opracować dokumentację do wdrażanego rozwiązania

w zakresie kompetencji społecznych:

1. jest gotów optymalizować rozwiązania

Treści programowe dla zajęć:

Przygotowanie projektu informatycznego

Opracowanie dokumentacji rozwiązania i jego wdrożenia

Potrafi wybrać technologie i je zastosować do realizacji rozwiązań na potrzeby projektu

Opracowanie rozwiązań technicznych

Nazwa zajęć: **Zaawansowane języki programowania**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie wiedzy:

1. wie: Jakie są elementarne instrukcje wykonywane przez procesory. Co to są asemblery, języki: kompilowane, interpretowane i działające w oparciu o maszyny wirtualne. Zna podstawowe elementy występujące w językach programowania.

2. zna: podstawowe słowa kluczowe języków C i C++. Zna składnię, wbudowane typy danych, instrukcje sterujące. Zna pojęcia klas. Zna koncepcję dziedziczenia, funkcji wirtualnych, klas abstrakcyjnych, polimorfizmu.

3. zapoznaje się z bibliotekami runtime C/C++, pojęciami szablonów, obiektów funkcyjnych. Zapoznaje się z biblioteką STL - strukturami danych, iteratorami, algorytmami.

4. zapoznaje się z zasadami tworzenia programów interakcyjnych. Zapoznaje się z koncepcją GUI (Graphical User Interface) na przykładzie biblioteki wxWidgets

5. zapoznaje się z zasadami tworzenia programów wielowątkowych. Mutex'y Semaphores. Zapoznaje się z koncepcją smart pointers - klasy unique_ptr, shared_ptr, weak_ptr,

w zakresie umiejętności:

1. zapoznaje się ze środowiskiem IDE. Potrafi tworzyć "projekty", pisać i kompilować proste programy.

2. pisać programy używając różne typy danych, tablice, struktury, wskaźniki, dynamiczną alokację pamięci, operacje wejścia wyjścia.
3. potrafi definiować klasy. Używać zdefiniowane przez siebie typy danych. Używać bardziej złożone typy danych. Wykorzystywać dziedziczenie i funkcje wirtualne. Wykorzystywać wyjątki.
4. potrafi używać szablony klas i funkcji. Tworzyć programy z wykorzystaniem STL.
5. potrafi tworzyć programy z interface'em graficznym.

Treści programowe dla zajęć:

Informacje ogólne o językach programowania. Kompilacja. Identyfikatory proste (wbudowane) typy danych oraz ich reprezentacja binarną. Zakres widoczności zmiennych. Podstawowe operatory. Instrukcje sterujące, pętle. Instrukcje warunkowe. Funkcje scanf i printf. Formatowanie wydruków (łańcuchy formatujące) w C. Operatory ++ i --. Wywoływanie funkcji z argumentami. Wskaźniki. Tablice. Struktury. Struktury anonimowe. Typy danych użytkownika w C. Dostęp do składowych struktur. Wywoływanie funkcji z wykorzystaniem struktur. Pola bitowe. Unie. Deklarowanie wskaźników. Arytmetyka na wskaźnikach. Konwersja wskaźników. Wskaźniki do funkcji. Stałe a wskaźniki.

Inicjalizacja zmiennych. Operacje wejścia wyjścia w C. Przykłady. Ćwiczenia: mnożenie macierzy, Dynamiczna tabela łańcuchów. Alokacja tablic w „stylu Pascal'a”. Symulacja rzutów monetą. Biblioteczne funkcje do manipulacji łańcuchów. Listy. Stos. Kolejka.

Operator wywołania funkcji. Modyfikatory deklaracji.

Klasy. Obiekty. Deklaracje klas za pomocą: class, struct, union.

Dziedziczenie. Dostęp do składowych klas.

Dostęp do składowych klas bazowych i potomnych.

Wirtualne klasy bazowe.

Funkcje zaprzyjaźnione.

Operacje wejścia wyjścia w C++. Manipulatory. Klasy ofstream, ifstream. Operatory << >>.

Właściwości konstruktorów i destruktorów. Konstruktory: domyślny, kopiujący.

Przeddefiniowanie konstruktorów. Kolejność wywołania konstruktorów.

Inicjalizacja klas. Konstruktory jednoargumentowe.

Destruktry. Wywołanie destruktorów. Destruktry a funkcja atexit i dyrektywa kompilatora #pragma exit. Destruktry a funkcja exit.

Destruktry a funkcja abort. Destruktry wirtualne

Słowo kluczowe this.

Składowe statyczne.

Funkcje inline.

Przeddefiniowywanie operatorów – binarnych, unarnych. Operatorów: =, (), [], ->. Definicje operatorów inkrementacji i dekrementacji.

Zmiana definicji operatorów &&, || i, .

Funkcje wirtualne.

Klasy abstrakcyjne. Klasy polimorficzne.

Wyjątki.

Pominięte wywołania destruktorów. Wyjątki w konstruktorach.

Niejawna (implicit) konwersja typów danych.

Operatory new i delete.

Makra. Zagnieżdżone nawiasy. Przecinki.

Wstawianie tokena za pomocą ##.

Zamiana identyfikatora zmiennej na łańcuch za pomocą #.

Kompilacja warunkowa.

Szablony. Szablony funkcji. Przekrywanie szablonów funkcji. Jawne i niejawne szablony funkcji.

Szablony klas. Argumenty szablonów.

Biblioteka STL (SCL). Motywacja utworzenia biblioteki. STL a standard ANSI/ISO. Klasy nice.

Obiekty funkcyjne.

Przegląd biblioteki STL. Kontenery. Array, Vector. List. Double ended queue. Set. Map. Hash tables.

Iteratory. Iteratory wejścia/wyjścia. Iteratory jednokierunkowe. Iteratory dwukierunkowe. Iteratory swobodnego dostępu.

Algorytmy i obiekty funkcyjne. Jak utworzyć algorytm generyczny?

Inne tematy związane z STL. Adaptory. Stos. Kolejki. Adaptory iteratorów. Iteratory wsteczne. Iteratory pamięci. Binder'y.

Alokatory. Adaptory funkcji.

Kontenery skojarzeniowe

Mutex. Semafor. Inteligentne wskaźniki.

Wątki. Wątki utworzone za pomocą API Windows. Priorytety wątków.
Sekcja krytyczna.
Synchronizacja wątków
Przeszukiwanie dysku za pomocą funkcji shell'a (API).
Wininet API. Biblioteka cURL.
Biblioteka wxWidgets

Nazwa zajęć: **Zarządzanie usługami typu hosting WWW**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Posiada wiedzę o architekturze i zasadach działania serwera WWW
2. Posiada wiedzę o protokołach komunikacji używanych przez serwery WWW (http, https)
3. Posiada wiedzę o architekturze bezpiecznego dostępu do stron WWW (certyfikaty, CA)
4. Posiada wiedzę o aplikacjach server-side, związanym z nimi ryzykiem i sposobach i zabezpieczenia
5. Posiada wiedzę o typach ataków na strony WWW i sposobach zabezpieczeń przed nimi

w zakresie umiejętności:

1. Posiada wiedzę i umiejętności konfiguracji oprogramowania serwera WWW (Apache, Nginx)
2. Posiada wiedzę i umiejętności konfiguracji kontroli dostępu do usług WWW
3. Posiada wiedzę i umiejętności konfiguracji przekształceń adresu URL na poziomie serwera WWW (mod_rewrite)
4. Posiada wiedzę i umiejętności konfiguracji oprogramowania typu firewall
5. Posiada wiedzę i umiejętności bezpiecznej konfiguracji maszyn wirtualnych

Treści programowe dla zajęć:

Podstawowe informacje o architekturze usługi WWW
Protokół HTTP (request i response, podstawowe pola nagłówka, sygnalizacja błędów)
Protokół HTTPS i związana z nim infrastruktura (certyfikaty, podpisy, CER, CA)
Podstawowe informacje o serwerze WWW Apache
Podstawowa konfiguracja Apache (dyrektywy core, dyrektywy blokowe, system modułów Apache'a)
Manipulowanie adresem URL na poziomie serwera WWW – mod_rewrite
Aplikacje server side na przykładzie PHP
Typowe ataki na serwer WWW (XSS, injection, DoS) i sposoby przeciwdziałania
Zabezpieczenie serwera WWW przy pomocy firewalla
Podstawy wirtualizacji
Hosting na maszynach wirtualnych na przykładzie Virtualbox

Nazwa zajęć: **Pracownia dyplomowa**

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. zna i rozumie podstawowe pojęcia i zagadnienia związane z wybranymi głównymi obszarami fizyki i informatyki.
2. zna właściwe dla wybranych obszarów zastosowań technologie, narzędzia i urządzenia informatyczne, oraz fizyczne podstawy ich działania.

w zakresie umiejętności:

1. potrafi stosować zdobytą wiedzę do prowadzenia projektu obejmującego zagadnienia z obszaru kształcenia na kierunku TK.
2. potrafi korzystać z literatury fachowej, również w języku angielskim.
3. potrafi samodzielnie pracować nad określonym projektem z obszaru kształcenia na kierunku TK.
4. potrafi pisać dłuższe opracowanie opisujące projekt inżynierski z uwzględnieniem danych literaturowych.

w zakresie kompetencji społecznych:

1. jest gotów/gotowa do prezentowania zdobytej wiedzy i wypracowanych technologii w ramach projektu oraz prowadzenia dyskusji w tym obszarze.

Treści programowe dla zajęć:

organizacja pracy nad projektem inżynierskim.
wybór i zastosowanie właściwych dla wybranych obszarów zastosowań technologii, narzędzi, metod i sprzętu w celu realizacji projektu pracy inżynierskiej; umiejętność systematycznej pracy, gromadzenia wyników/danych oraz formułowania wniosków.
umiejętność pisania pracy inżynierskiej na bazie wykonanego projektu z uwzględnieniem literatury.
umiejętność przygotowania prezentacji wykonanego projektu inżynierskiego.

Nazwa zajęć: Sztuczna Inteligencja - Symulacje w Środowisku Rzeczywistości Cyfrowych.
Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie wiedzy:

1. Zna i rozumie metodyki projektowania symulacji wykorzystując rozwiązania technologii sztucznej inteligencji (SI).
2. Zna i rozumie techniki zarządzania systemami wieloagentowymi.
3. Zna i rozumie zasady modelowania dynamiki systemów fizycznych w symulacjach.
4. Zna i rozumie sposoby modelowania różnych sensorów i ich danych w symulacji.
5. Zna i rozumie metody generowania i strukturyzacji danych syntetycznych.
6. Zna i rozumie metody integracji symulacji z zewnętrznymi środowiskami SI.
7. Zna i rozumie podstawowe algorytmy i paradygmaty uczenia agentów w symulacjach w środowiskach rzeczywistości cyfrowych (RC).
8. Zna i rozumie, jak formułować problemy symulacyjne dla badań ewolucyjnych/behawioralnych.
9. Zna i rozumie implementację podstawowych algorytmów ewolucyjnych/adaptacyjnych.
10. Zna i rozumie techniki wizualizacji danych symulacyjnych w środowiskach RC.
11. Zna i rozumie koncepcje i wyzwania wstępnego trenowania w symulacjach RC.
12. Zna i rozumie koncepcje wierności symulacji, walidacji i luki sim-to-real.
13. Zna i rozumie strategie i metody skalowania symulacji SI.

w zakresie umiejętności:

1. Potrafi projektować eksperymenty symulacyjne dla agentów SI.
2. Potrafi implementować systemy wielu agentów w RC.
3. Potrafi implementować modele systemów dynamicznych.
4. Potrafi implementować i konfigurować różne symulowane sensory.
5. Potrafi konfigurować symulacje do generowania etykietowanych danych syntetycznych.
6. Potrafi ustanawiać komunikację między symulacją a zewnętrznym kodem SI.
7. Potrafi implementować podstawowe pętle uczenia agentów w symulacjach RC.
8. Potrafi projektować środowiska symulacyjne do badania adaptacji/ewolucji.
9. Potrafi implementować podstawowe algorytmy ewolucyjne/adaptacyjne w symulacji.
10. Potrafi implementować wizualizacje danych w symulacji RC.
11. Potrafi zaprojektować procesy przetwarzania dla wstępnego trenowania w symulacji RC.
12. Potrafi stosować podstawowe techniki walidacji symulacji.
13. Potrafi stosować metody skalowania zadań symulacyjnych.

w zakresie kompetencji społecznych:

1. Jest gotów/gotowa do formułowania problemów badawczych/inżynierskich dla symulacji SI.
2. Jest gotów/gotowa do efektywnej współpracy przy eksperymentach symulacyjnych.
3. Jest gotów/gotowa do krytycznej interpretacji wyników symulacji i oceny agentów.
4. Jest gotów/gotowa do nauki i stosowania technik symulacji i SI w rozszerzonym zakresie
5. Jest gotów/gotowa do dokumentowania eksperymentów symulacyjnych.

Treści programowe dla zajęć:

Analiza architektur symulacji wieloagentowych.
Projektowanie symulacji do trenowania systemów inteligentnych.
Modelowanie systemów dynamicznych w środowiskach cyfrowych.
Symulowana percepcja - Modalności sensoryczne i generowanie danych.
Strategie generowania danych syntetycznych do trenowania SI.
Tworzenie interfejsów między symulacjami a środowiskami inteligencji obliczeniowej.
Paradygmaty uczenia agentów w symulacjach RC.
Projektowanie symulacji do badania zachowań emergentnych i optymalizacji.
Implementacja algorytmów adaptacyjnych i ewolucyjnych w symulacjach.
Techniki wizualizacji i analizy danych SI/RC.
Wstępne trenowanie modeli SI w symulacjach RC i transfer do świata rzeczywistego.
Wierność symulacji, walidacja i wypełnianie luki między symulacją a światem rzeczywistym (*sim-to-real*).
Skalowalne i rozproszone metody symulacji.
Badania i eksperymenty w środowisku SI/RC.
Projekt – Projektowanie, implementacja i analiza symulacji SI w środowisku RC.

Nazwa zajęć: Język angielski A2

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie umiejętności:

1. potrafi czytać ze zrozumieniem krótsze teksty w języku angielskim o charakterze ogólnym.

2. potrafi zrozumieć prosty oryginalny materiał audio lub wideo z życia codziennego, kulturalnego i społecznego, na poziomie ogólnym jak i wychwycić niezbędne szczegóły.

w zakresie kompetencji społecznych:

1. porozumiewać się w rutynowych, prostych sytuacjach komunikacyjnych, wymagających jedynie bezpośredniej wymiany zdań na tematy znane i typowe. Potrafi w prosty sposób opisywać swoje pochodzenie i otoczenie, w którym żyje, a także poruszać sprawy związane z najważniejszymi potrzebami życia codziennego.

Treści programowe dla zajęć:

Czasy gramatyczne potrzebne do wyrażania różnorodnych czynności osadzonych w czasie Present Simple and Present Continuous, Past Simple and Past Continuous, Present Perfect and Present Perfect Continuous, Past Perfect oraz czasach przyszłych na poziomie A2

Inne struktury gramatyczne potrzebne do wyrażania różnorodnych treści i opinii (np. czasowniki modalne, przymiotniki, strona bierna, zdania warunkowe, mowa zależna) dla poziomu A2

Słownictwo dotyczące życia codziennego oraz związane z bezpośrednim środowiskiem studenta (jedzenie, osobowość, podróże, zainteresowania, edukacja, zakupy, pieniądze, technologia, rodzina, studia, praca, technologia, podstawowe słownictwo związane z kierunkiem studiów)

Strategie efektywnego czytania w celu zrozumienia ogólnego sensu wypowiedzi; domyślanie się znaczenia nieznanych słów

Strategie efektywnego słuchania w celu zrozumienia ogólnego sensu wypowiedzi; domyślanie się znaczenia nieznanych słów

Wyrażanie różnorodnych funkcji językowych np. prośby, opisy, wyrażanie opinii, wyrażanie zgody, brak zgody, pytania o pozwolenie, skargi, itp.

Nazwa zajęć: Język angielski B1

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie umiejętności:

1. potrafi tworzyć ustne wypowiedzi na przygotowane tematy, prezentować i argumentować swoje stanowisko oraz innych osób w zakresie problematyki związanej ze swoim otoczeniem jak i w zakresie tematyki ogólno-akademickiej

2. potrafi czytać ze zrozumieniem teksty w języku angielskim o charakterze ogólnym jak i akademickim oraz analizować ich treść i wybierać niezbędne informacje.

3. potrafi zrozumieć dostosowany do poziomu oryginalny materiał audio lub wideo na poziomie ogólnym oraz wychwytywać niezbędne szczegóły.

w zakresie kompetencji społecznych:

1. trenuje umiejętności komunikacyjne i pracy w zespole. wypracowuje wspólny efekt.

Treści programowe dla zajęć:

Czasy gramatyczne potrzebne do wyrażania różnorodnych czynności osadzonych w czasach: Present Simple and Present Continuous, Narrative Tenses, Present Perfect and Present Perfect Continuous, Future Perfect and Future Continuous.

Inne struktury gramatyczne potrzebne do wyrażania różnorodnych treści i opinii: mowa zależna oraz pytania w mowie zależnej, formy przymiotnikowe i przysłówkowe.

Słownictwo dotyczące życia codziennego oraz jak i ogólno-akademickie w zakresie następujących tematów: praca, rozmowa kwalifikacyjna o pracę, służba zdrowia, podróżowanie, moda oraz dress code, środowisko naturalne, zmiany klimatyczne.

Strategie efektywnego czytania w celu zrozumienia ogólnego sensu wypowiedzi; domyślanie się znaczenia nieznanych słów w zakresie bloków tematycznych określonych w treści 3.

Strategie efektywnego słuchania w celu zrozumienia ogólnego sensu wypowiedzi; domyślanie się znaczenia nieznanych słów w zakresie bloków tematycznych określonych w treści 3.

Udzielanie odpowiedzi, udział w dyskusji oraz wyrażanie różnorodnych funkcji językowych w zakresie: przeprowadzania oraz udziału w rozmowie kwalifikacyjnej o pracę, przedstawiania problemów, moderowania dyskusji oraz wyrażania opinii na tematy zawarte w treści 3.

Nazwa zajęć: Język angielski B21

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka

w zakresie umiejętności:

1. potrafi tworzyć płynne wypowiedzi ustne na przygotowane tematy, prezentować i argumentować swoje stanowisko oraz innych osób na tematy związane ze swoim otoczeniem jak i na tematy ogólno-akademickie.

2. potrafi czytać ze zrozumieniem teksty w języku angielskim o charakterze ogólnym jak i akademickim, związane z kierunkiem studiów, oraz analizować ich treść i wybierać niezbędne informacje.

3. potrafi zrozumieć oryginalny materiał audio lub wideo na większość tematów dotyczących życia codziennego, kulturalnego i społecznego, na poziomie ogólnym jak i wychwycić niezbędne szczegóły.
4. potrafi przygotować i wygłosić prezentację na wybrany temat.
5. potrafi opracować teksty oraz wypowiedzi dotyczące życia społecznego, uniwersyteckiego i ^[13]_[SEP] zawodowego.
6. redagować wybrane teksty w stylu formalnym.
7. uzupełniać i doskonalić nabytą wiedzę i umiejętności.

Treści programowe dla zajęć:

Swobodne posługiwanie się czasami gramatycznymi w języku angielskim.

Inne struktury gramatyczne potrzebne do wyrażania różnorodnych treści i opinii: okresy warunkowe typ 1,2,3 oraz mieszane; struktury gramatyczne 'wish,'get used to/used to, past modals, formy bezokolicznikowe i imiesłowowe.

Słownictwo dotyczące problematyki współczesnego świata w zakresie następujących tematów: ekstremalne sytuacje, refleksja na temat planów życiowych, terapeutyczna funkcja muzyki, higiena snu, komunikacja niewerbalna oraz wybrane słownictwo akademickie i specjalistyczne związane z kierunkiem studiów.

Strategie efektywnego czytania w celu zrozumienia ogólnego sensu wypowiedzi w tekstach popularno-naukowych oraz specjalistycznych; domyślanie się znaczenia nieznanych słów w zakresie bloków tematycznych określonych w treści 3.

Strategie efektywnego słuchania w celu zrozumienia ogólnego sensu wypowiedzi; domyślanie się znaczenia nieznanych słów w zakresie bloków tematycznych określonych w treści 3.

Udzielanie odpowiedzi, udział w dyskusji oraz wyrażanie różnorodnych funkcji językowych w zakresie tematyki określonej w treści 3.

Redagowanie wybranych typów tekstów formalnych

Nazwa zajęć: Język angielski B22

Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się student/ka w zakresie umiejętności:

1. potrafi tworzyć płynne wypowiedzi ustne na przygotowane tematy, prezentować i argumentować swoje stanowisko oraz innych osób na tematy związane ze swoim otoczeniem jak ja na tematy ogólno-akademickie.
2. potrafi czytać ze zrozumieniem teksty w języku angielskim o charakterze ogólnym jak i akademickim, związane z kierunkiem studiów, oraz analizować ich treść i wybierać niezbędne informacje.
3. potrafi zrozumieć oryginalny materiał audio lub wideo na większość tematów dotyczących życia codziennego, kulturalnego i społecznego, na poziomie ogólnym jak i wychwycić niezbędne szczegóły.
4. potrafi przygotować i wygłosić prezentację na wybrany temat.
5. potrafi opracować teksty oraz wypowiedzi dotyczące życia społecznego, uniwersyteckiego i ^[13]_[SEP] zawodowego.
6. potrafi redagować wybrane teksty w stylu formalnym.
7. potrafi uzupełniać i doskonalić nabytą wiedzę i umiejętności.

Treści programowe dla zajęć:

Swobodne posługiwanie się czasami gramatycznymi w języku angielskim.

Inne struktury gramatyczne potrzebne do wyrażania różnorodnych treści i opinii: strona bierna, następstwo czasów, zdania celu, porównania, rzeczowniki policzalne i niepoliczalne, przedimki.

Słownictwo dotyczące problematyki współczesnego świata w zakresie następujących tematów: system sprawiedliwości, przestępstwa internetowe, świat mediów i e-mediów, problematyka biznesu i ekonomii, reklamy, nowoczesne miasta, wystąpienia publiczne, problemy współczesnej nauki, tematyka science-fiction oraz wybrane słownictwo akademickie i specjalistyczne związane z kierunkiem studiów

Strategie efektywnego czytania w celu zrozumienia ogólnego sensu wypowiedzi w tekstach popularno-naukowych oraz specjalistycznych; domyślanie się znaczenia nieznanych słów w zakresie bloków tematycznych określonych w treści 3.

Strategie efektywnego słuchania w celu zrozumienia ogólnego sensu wypowiedzi; domyślanie się znaczenia nieznanych słów w zakresie bloków tematycznych określonych w treści 3.

Udzielanie odpowiedzi, udział w dyskusji oraz wyrażanie różnorodnych funkcji językowych w zakresie tematyki określonej w treści 3.

Redagowanie wybranych typów tekstów formalnych