

ZAGADNIENIA NA EGZAMIN DYPLOMOWY INŻYNIERSKI

Kierunek studiów: **INFORMATYKA**

Studia stacjonarne i niestacjonarne pierwszego stopnia

Lp.	Zagadnienie
Matematyka dyskretna	
1.	Pierwsza i druga zasada indukcji matematycznej.
2.	Liniowa, jednorodna zależność rekurencyjna rzędu drugiego ze stałymi współczynnikami i sposób jej rozwiązywania.
3.	Graf pełny, dopełnienie grafu, graf dwudzielny oraz pełny graf dwudzielny – definicje.
4.	Definicja transwersali oraz twierdzenie Halla w postaci transwersalowej.
Algorytmy i struktury danych	
5.	Drzewo binarne jako struktura danych - rodzaje i własności drzew oraz podstawowe operacje na drzewach.
6.	Problemy poszukiwania cyklu Eulera i cyklu Hamiltona - ich złożoność obliczeniowa.
7.	Problem plecakowy - złożoność obliczeniowa i metody jego rozwiązywania.
8.	Kiedy dochodzi do kolizji w funkcji haszującej? Podaj sposoby na rozwiązanie konfliktów spowodowanych przez kolizje.
Systemy operacyjne	
9.	Zarządzanie procesami i wątkami (w tym: algorytmy przydziału procesora).
10.	Zarządzania pamięcią operacyjną (organizacja pamięci, pamięć wirtualna).
11.	Obsługa urządzeń wejścia-wyjścia.
12.	Organizacja systemu plików dyskowych (metody alokacji plików na dysku, zarządzanie obszarami wolnymi, przykłady implementacji).
Podstawy elektroniki	
13.	Twierdzenie Thevenina i Nortona.
14.	Budowa i działanie tranzystorów NMOS i PMOS.
15.	Charakterystyki prądowo-napięciowe tranzystorów CMOS $I_{ds}=f(V_{ds})$, $I_{ds}=g(V_{gs})$.
16.	Budowa i działanie inwertera CMOS (bramka NOT).
17.	Budowa i działanie bramek NAND i NOR zbudowanych z wykorzystaniem tranzystorów CMOS.
18.	Bloki funkcjonalne układu FPGA i ich funkcje.
Programowanie deklaratywne	
19.	Paradygmat programowania deklaratywnego a inne paradygmaty programowania (proceduralny, obiektowy) - cechy charakterystyczne, różnice, interpretacja deklaratywna a proceduralna programu.
20.	Efektywność programów prologowych - źródła niskiej efektywności programów prologowych, metody ogólne poprawy efektywności i metody szczególne charakterystyczne dla języka Prolog.

Programowanie niskopoziomowe	
21.	Tworzenie obiektów przechowywanych na stosie i stercie w językach C i C++.
22.	Przekazywanie tablicy do funkcji w języku C.
Podstawy techniki cyfrowej	
23.	Sposoby określenia postaci funkcji przełączającej.
24.	Automaty Moore'a i Mealy'ego – różnice w działaniu i opisie sposobu działania.
25.	Opis układu cyfrowego na poziomie przestań międzyrejestrowych, przykłady operacji.
26.	Automat mikroprogramowany – warianty struktur i obsługiwanych rozkazów.
27.	Sieć działań algorytmu, a diagram ASM – porównanie.
Programowanie systemowe i współbieżne	
28.	Programowanie współbieżne i synchronizacja wątków - jakie własności powinien spełniać poprawny program, a jakich nie powinien i dlaczego?
29.	Omów konstrukcje programistyczne służące do synchronizacji wątków w programach współbieżnych (w tym do synchronizacji warunkowej) oraz wyjaśnij ogólnie jak te konstrukcje są zaimplementowane. W przypadku prostych zamków podaj wybrany algorytm wzajemnego wykluczania.
30.	Omów problem zakleszczenia w programowaniu współbieżnym oraz wyjaśnij przykładowe metody zapobiegania, detekcji, oraz unikania zakleszczenia.
31.	Zastosuj semaforey lub monitory do rozwiązania problemu czytelników i pisarzy oraz problemu producenta i konsumenta, czyli zaimplementuj odpowiednio: reader-writer lock i bufor o ograniczonym rozmiarze przy użyciu semaforów lub monitorów.
Badania operacyjne	
32.	Ogólne sformułowanie problemu programowania matematycznego, jego szczególne przypadki (programowanie liniowe, programowanie liniowe całkowitoliczbowe, zadania z hiperboliczną funkcją celu, zadania z minimaxową funkcją celu) oraz metody ich rozwiązywania.
33.	Metody konstruowania rozwiązania początkowego w metodzie sympleks.
34.	Algorytm kompresji, czyli jak użyć dodatkowych zasobów, aby uzyskać najkrótszy (graniczny) czas realizacji projektu przy najniższych kosztach.
Optymalizacja kombinatoryczna	
35.	Złożoność obliczeniowa problemów optymalizacyjnych, algorytmy aproksymacyjne i trudność problemów aproksymacji, matroidy.
36.	Praktyczne metody rozwiązywania problemów optymalizacji kombinatorycznej – metoda podziału i ograniczeń, metaheurystyki, algorytmy zachłanne.
37.	Przepływy w sieciach - sformułowanie problemu, metody rozwiązania, zastosowania.
38.	Kolorowanie grafów - sformułowania problemu, metody rozwiązania, zastosowania, ograniczenia na liczbę i indeks chromatyczny.
Programowanie obiektowe	
39.	Podstawowe założenia metodyk programowania obiektowego.
40.	Podstawowe mechanizmy programowania obiektowego.
Podstawy automatyki	
41.	Narzędzia opisu dynamiki i symulacji obiektów sterowania.
42.	Struktura układu regulacji automatycznej i charakterystyka jego elementów.

43.	Stabilność i jakość regulacji.
Architektura systemów komputerowych	
44.	Działanie komputera wg architektury von Neumanna.
45.	Mechanizm funkcjonowania systemu przerwań komputera.
46.	Idea potokowości w budowie procesorów.
Sieci komputerowe 1	
47.	Modele warstwowe architektury sieci komputerowej oraz podstawowe urządzenia sieciowe.
48.	Media transmisyjne i metody dostępu do medium.
49.	Protokół IP. Zasada działania, adresacja, podział na podsieci.
Systemy baz danych	
50.	Relacyjny model danych: struktury danych, operacje, więzy integralności.
51.	Projektowanie baz danych: model ER, transformacja modelu ER do postaci relacyjnej, proces normalizacji.
Grafika komputerowa i wizualizacja	
52.	Wizualizacja obiektów 3D.
53.	Etapy procesu wizualizacji danych.
Sieci komputerowe 2	
54.	Protokoły wyboru trasy.
55.	Sieci wirtualne VLAN.
56.	System nazw domen DNS.
57.	Filtracja pakietów i translacja adresów IP.
Zarządzanie bazami SQL i NoSQL	
58.	Transakcje w bazach danych i ich własności.
59.	Odtwarzanie spójnego stanu bazy danych po awarii.
60.	Synchronizacja współbieżnych transakcji.
61.	Dyskowe struktury danych, metody dostępu i optymalizacja zapytań w bazach danych.
62.	Bazy danych NoSQL.
Wspomaganie decyzji	
63.	Funkcyjne, relacyjne i regułowe modele preferencji – definicje i przykłady metod wspomaganie decyzji, które je wykorzystują.
64.	Konstrukcja funkcji użyteczności metodą regresji porządkowej w ramach funkcyjnego modelu preferencji.
65.	Konstrukcja relacji przewyższania w trybie testów zgodności i niezgodności w ramach relacyjnego modelu preferencji.
66.	Indukcja reguł decyzyjnych z przybliżeń klas decyzyjnych (podejście zbiorów przybliżonych).
Inżynieria oprogramowania	
67.	Zwinne podejścia do wytwarzania oprogramowania i Scrum.
68.	Metody specyfikacji wymagań.
69.	Modelowanie oprogramowania - w tym notacja UML oraz wzorce projektowe.

70.	Testowanie oprogramowania.
Komunikacja człowiek-komputer	
71.	Porównanie modeli barw RGB i HSV a teoria percepcji barw.
72.	Metoda ewaluacji heurystycznej do oceny użyteczności interfejsu użytkownika.
73.	Przetwarzanie sygnału analogowego na postać cyfrową.
Sztuczna inteligencja	
74.	Heurystyczne i nieheurystyczne strategie przeszukiwania przestrzeni stanów.
75.	Formy reprezentacji wiedzy dziedzinowej - systemy regułowe, logika rozmyta, wnioskowanie bayesowskie.
76.	Sztuczne sieci neuronowe - typu architektur, budowa perceptronu, algorytm wstecznej propagacji, sieć Hopfielda ze sprzężeniem zwrotnym.
Systemy wbudowane	
77.	Transmisja danych w warstwie fizycznej w paśmie podstawowym i radiowym. Metody modulacji.
78.	Interfejsy komunikacyjne wykorzystywane w systemach wbudowanych.
79.	Wykorzystanie systemów wbudowanych w sterowaniu procesami.
Podstawy aplikacji internetowych	
80.	Reprezentacja dokumentu HTML za pomocą modelu DOM.
81.	CSS: pojęcia, zastosowania, składnia.
82.	Protokół HTTP: zastosowania, komunikaty, komendy.
83.	Zmienne Cookies: własności, przekazywanie, sposoby użycia.
84.	Przykłady zagrożeń bezpieczeństwa dla aplikacji WWW.
85.	Charakterystyka języka programowania JavaScript.
Przetwarzanie rozproszone	
86.	Charakterystyka systemów przetwarzania rozproszonego: (kanały komunikacyjne, modele żądań, relacja poprzedzania, diagramy przestrzenno-czasowe, niedeterminizm przetwarzania, zegary wektorowe i skalarne oraz algorytmy je realizujące).
87.	Problem konstrukcji spójnego obrazu stanu globalnego i przykładowe algorytmy jego rozwiązania.
88.	Problem detekcji zakończenia i przykładowe algorytmy jego rozwiązania.
Aplikacje mobilne	
89.	Cechy różnych typów aplikacji mobilnych (natywne, hybrydowe, wieloplatformowe itd.).
90.	Cykl życia aktywności i fragmentów w aplikacjach mobilnych dla systemu Android.
91.	Mechanizmy stosowane w natywnym programowaniu aplikacji mobilnych dla systemu Android (adaptery, handlers, intencje, interfejsy, stos cofnięć, transakcje, układy, zasoby, biblioteka wsparcia wzornictwa itd.).
92.	Rola biblioteki wsparcia wzornictwa w projektowaniu i programowaniu aplikacji mobilnych dla systemu Android.
Przetwarzanie równoległe	
93.	Lokalność dostępu do danych – rodzaje, charakterystyka i wpływ na efektywność przetwarzania równoległego dla procesorów wielordzeniowych.

94.	Zasady przetwarzania równoległego ogólnego przeznaczenia przy użyciu procesorów kart graficznych.
95.	Prawo Amdahla, prawo Gustafsona – implikacje dla realizacji efektywnych obliczeń w systemach równoległych.
Wprowadzenie do sieci neuronowych i uczenia maszynowego	
96.	Przedstaw i zilustruj graficznie działanie algorytmu k-średnich (k-means). Do jakiego typu problemów zazwyczaj go stosujemy? Jakie są jego zalety i wady?
97.	Wyjaśnij pojęcie klątwy wymiarowości. Jakie ma ono implikacje dla praktyki uczenia maszynowego? Jakie znasz środki zaradcze które adresują to wyzwanie?
98.	Opisz formalnie i objaśnij działanie nieliniowej jednostki (neurona) stosowanego w sztucznych sieciach neuronowych. Wymień i opisz przynajmniej 3 typy nieliniowości implementowanych przez tego typu jednostki. Jaki jest ich wpływ na przebieg uczenia?
Bezpieczeństwo systemów informatycznych	
99.	Modele uwierzytelniania oraz mechanizmy je implementujące.
100.	Modele autoryzacji i mechanizmy implementujące kontrolę dostępu.
101.	Ataki odmowy dostępu do usług (Denial of Service) – przykład ataku i sposoby jego unikania.
102.	Wykorzystanie wirtualizacji do podnoszenia bezpieczeństwa systemu.
Przetwarzanie masywnych danych - BigData	
103.	Indeks Blooma, sposób działania, przykłady wykorzystania w systemach Big Data.
104.	Obsługa operacji modyfikacji danych na przykładzie wybranych 2-3 platform Big Data korzystających z HDFS.
105.	Porównanie kluczowych silników przetwarzania danych Big Data: MapReduce, Tez i Spark, funkcjonujących jako aplikacje YARN.