

Załącznik do Uchwały Nr 20/2020 Rady Programowo-Dydaktycznej
Wydziału Nauk Humanistycznych i Informatyki
Mazowieckiej Uczelni Publicznej w Płocku z dnia 25 listopada 2020 r.

Mazowiecka Uczelnia Publiczna w Płocku
Wydział Nauk Humanistycznych i Informatyki

LISTA ZAGADNIEŃ OBOWIĄZUJĄCA NA EGZAMINIE INŻYNIERSKIM
DLA KIERUNKU INFORMATYKA

Na egzaminie student powinien znać i omówić materiał zawarty w pracy oraz znać podstawowe zagadnienia z dziedziny, z której pisana jest praca. Ponadto student powinien wykazać się znajomością wymienionych poniżej podstawowych zagadnień z matematyki i informatyki (jest to warunek konieczny do zdania egzaminu inżynierskiego).

1. Podstawowe operacje na zbiorach i prawa rachunku zbiorów.
2. Rachunek zdań. Pojęcie tautologii w rachunku zdań.
3. Macierze oraz podstawowe operacje na macierzach.
4. Ciągi liczbowe (granica, zbieżność, monotoniczność).
5. Pojęcie granicy i ciągłości funkcji w punkcie.
6. Pochodna funkcji w punkcie i jej interpretacja geometryczna.
7. Ekstremum funkcji i ich zależność z pochodną.
8. Całka nieoznaczona i oznaczona funkcji.
9. Klasyczna definicja prawdopodobieństwa.
10. Struktura logiczna i funkcjonalna klasycznego komputera.
11. Reprezentacja liczb w systemie pozycyjnym (dziesiętnym, dwójkowym i szesnastkowym).
12. Arytmetyka stałopozycyjna i zmiennopozycyjna. Reprezentacja liczb w komputerze.
13. Zmienne i ich typy.
14. Zmienne (typy) wartościowe i referencyjne.
15. Metody przekazywania parametrów do funkcji.
16. Iteracja i rekurencja. Przykłady.
17. Paradygmaty programowania strukturalnego (sekwencyjność, wybór, iteracja).
18. Paradygmat abstrakcji w programowaniu obiektowym.
19. Paradygmat hermetyzacji w programowaniu obiektowym.
20. Paradygmat dziedziczenia w programowaniu obiektowym.
21. Paradygmat polimorfizmu w programowaniu obiektowym.
22. Rodzaje metod: konstruktory, destruktory, metody operatorowe, metody statyczne.
23. Tablice, listy, stosy i kolejki. Budowa i zastosowania.
24. Drzewa – budowa i zastosowania.
25. Grafy i metody ich przeszukiwania.
26. Złożoność obliczeniowa algorytmu. Przykłady.
27. Elementarne algorytmy sortowania i wyszukiwania.

28. Podstawowe pojęcia analizy numerycznej: błąd bezwzględny i względny, epsilon maszynowy, uwarunkowanie, punkt startowy i kryterium końca.
29. Definicja bazy danych i relacyjnej bazy danych.
30. Definicja relacji i jej własności.
31. Definicja klucza. Rodzaje kluczy w relacyjnych bazach danych.
32. Podział i składnia instrukcji języka SQL.
33. Kaskadowy i przyrostowy model tworzenia oprogramowania.
34. Etapy życia oprogramowania – od „idei” do końca użytkowania.
35. Systemy kontroli wersji – przykłady i zastosowanie.
36. Warstwowy model sieci ISO/OSI.
37. Warstwowy model TCP/IP.
38. Adresowanie IP (budowa adresu, maska, adres sieci, adres rozgłoszeniowy, ilość adresów w sieci).
39. Budowa dokumentu i znaczniki HTML. Metody przekazywania danych pomiędzy stronami.
40. Popularne metody sztucznej inteligencji: (sieci neuronowe, algorytmy genetyczne, zbiory i logika rozmyta).

Dodatkowe pytania dla specjalności

Grafika komputerowa i projektowanie gier

1. Elementarne operacje wykonywane na punkcie, wektorze i macierzy wykorzystywane w grafice komputerowej.
2. Pojęcie bryły i obiektu w przestrzeni 3d.
3. Wektory normalne obiektu i ich zastosowanie.
4. Współrzędne tekstury i ich wykorzystanie.
5. Mapowanie środowiskowe – czym jest i do czego służy?
6. Struktury danych w grafice komputerowej.
7. Sposoby modelowania obiektów graficznych.
8. Modele barw w grafice komputerowej.
9. Obróbka grafiki rastrowej i wektorowej dla potrzeb serwisów internetowych.
10. Projektowanie i implementacja algorytmów gier komputerowych. Typy gier.

Dodatkowe pytania dla specjalności

Programowanie i bazy danych

1. Budowa dokumentu XML.
2. Wyrażenia ścieżkowe XPath. Podstawowe konstrukcje języka XQuery.
3. Paradygmat polimorfizmu w programowaniu obiektowym (metody wirtualne, klasy abstrakcyjne, interfejsy).
4. Zaawansowane algorytmy sortowania (QuickSort, MergeSort) i ich złożoność.
5. Wzorce projektowe. Przykłady.
6. Budowa aplikacji warstwowych. Przykłady.
7. Architektura bazodanowej aplikacji webowej. Warstwy i komunikacja.
8. Mechanizm mapowania relacyjno-obiektowego.
9. Systemy kontroli wersji. Przykłady i zastosowanie.
10. Narzędzia programistyczne (debuger, profiler).

Dodatkowe pytania dla specjalności

Sieci komputerowe i sieciowe systemy operacyjne

1. Struktura dokumentacji projektowej sieci komputerowej.
2. Zasady sporządzania harmonogramu prac wykonawczych sieci komputerowej.
3. Projektowanie i montaż okablowania sieci komputerowej. Metody i zasady pomiarów okablowania strukturalnego. Rodzaje testów i pomiarów pasywnych/aktywnych.
4. Przykłady usług sieciowych. Omów szczegółowo jedną z nich.
5. Proces i narzędzia wykorzystywane do konfiguracji interfejsu.
6. Proces i narzędzia wykorzystywane do konfiguracji trasowania.
7. Czynniki wpływające na bezpieczeństwo sieci.
8. Definicja systemu czasu rzeczywistego. Co różni te systemy od innych systemów.
9. Przykłady algorytmów szeregowania znajdujących zastosowanie w systemach czasu rzeczywistego.
10. Wirtualne sieci lokalne.