

**Конкурсний відбір вступників на навчання  
за освітньо-професійною програмою підготовки фахівців  
освітньо-кваліфікаційного рівня «Бакалавр» (старші курси)**

Галузь знань: 0501 «Інформатика та обчислювальна техніка»  
Напрямок підготовки: 6.050103 «Програмна інженерія»  
Перелік програмних питань, які виносяться на співбесіду

Для вступників на 2 курс

1. Як визначається поняття «група»? За якими критеріями можна класифікувати групу?
2. Як можна охарактеризувати поняття «роль». Які чинники визначають поняття «роль»?
3. Як визначається структура групи? Яким чином розташування членів групи в мережі впливає на групові процеси?
4. Що таке єдність? Які фактори сприяють єдності групи?
5. Які існують моделі розвитку груп? Як можна охарактеризувати модель розвитку групи за Б.Тукманом?
6. Що таке команда? Які основні відмінності робочої групи і команди?
7. Як можна охарактеризувати функціональну модель прийняття рішень?
8. Як можна охарактеризувати форми групової дискусії?
9. Що таке комунікація? Які існують форми комунікації?
10. Як можна охарактеризувати елементи й етапи процесу комунікації?
11. Як можна охарактеризувати стратегії розподілу пам'яті: статичний розподіл, динамічний розподіл в блоці, динамічний розподіл в купі?
12. Що таке структурний оператор? Як можна охарактеризувати умовні оператори та оператори повторення?
13. Що таке перелічувальний тип даних? Як можна його описати? Як він використовується?
14. Що таке індексований тип даних? Як можна охарактеризувати одновимірні та багатовимірні індексовані типи даних?
15. Які існують методи сортування значень індексованих типів даних?
16. Що таке іменованний тип даних? Як можна його описати? Як він використовується?
17. Що таке лінійна структура даних. Які є способи представлення лінійних структур даних?
18. Що таке підпрограма. Як можна класифікувати підпрограми? Як описується підпрограма?
19. Які існують об'єкти підпрограми? Як їх описати? Як вони використовуються?
20. Які існують способи передачі параметрів у підпрограму та способи повертання результату з підпрограми?
21. Які є принципи об'єктно-орієнтованого програмування?

22. Що таке клас? Який устрій класу? Які є зони дії і правила доступу до компонентів класу? Що є об'єкт класу?
23. Що таке конструктор та деструктор? Яке їх призначення? Які є типи конструкторів?
24. Що таке композиція? Як вона реалізується?
25. Які є члени класу? Яке їх призначення та правила використання?
26. Що таке перевантаження унарних та бінарних операцій?
27. Що таке дружня підпрограма? Яке призначення та правила їх використання?
28. Що таке одиночне та множинне наслідування?
29. Що таке поліморфізм? Як він застосовується?
30. Що таке абстрактний клас? Які методи є чисто віртуальними?

Список літератури  
для самостійної підготовки вступника до співбесіди

**Основна література**

1. Сидоров М.О. Групова динаміка. –К.: НАУ, 2008. –52 с.
2. Шилд Г. С++. Базовый курс: пер. с англ. – М.: Издательский дом «Вильямс», 2010. – 624 с.
3. Дейтел Х.М., Дейтел П.Дж. Как программировать на С++: пер. с англ. – М.: ЗАО «Издательство БИНОМ», 2005. – 1248с.
4. Шилд Г. С#: учебный курс. СПб.: Питер; К.: Издательская группа ВНУ, 2002. – 512 с.
5. Робинсон С. и др. С# для профессионалов. Т1,2. – М.: Лори, 2003. – 995 с.

**Додаткова література**

1. Соммервилл И. Инженерия программного обеспечения. – М.: Издательский дом «Вильямс», 2002. – 540 с.
2. Рунов А.В. Особенности коммуникативной среды в моделях информационного общества. – М.: РГИУ, 2003. – 112 с.
3. Кнут Д. Искусство программирования. Т1-Т3: пер. с англ. – М.: Издательский дом «Вильямс», 2007.
4. Майо Дж. С#: Искусство программирования. Энциклопедия программиста: пер. с англ. – СПб.: ООО «ДиаСофтЮП», 2002. – 656 с.

Для вступників на 3 курс

31. Що таке структура даних? За якими критеріями можна класифікувати структури даних? Приклади.
32. Як визначається структура даних «множина», її реалізація на основі хеш-таблиці. За якими критеріями можна класифікувати хеш-таблиці?
33. Як визначається структура даних «дерево»? Які є види дерев? Якими способами здійснюється обхід дерева (наприклад бінарного)?

34. Як визначається поняття «алгоритм»? Які його властивості? Приклади.

35. Що розуміється під сортуванням? На які групи розділяються алгоритми сортування?

36. У чому полягає сутність алгоритмів групи елементарного сортування?

37. У чому полягає сутність алгоритмів групи «швидкого» сортування? Які їх відмінності?

38. У чому полягає сутність групи алгоритмів порозрядного сортування?

39. Як визначається поняття "пошук"? За якими критеріями класифікуються алгоритми пошуку? Приклади.

40. Для чого необхідно аналізувати алгоритми? Які методи застосовуються для аналізу алгоритмів? Охарактеризуйте їх.

41. Що таке виключна ситуація? Які оброблюється виключна ситуація?

42. Які є принципи об'єктно-орієнтованого програмування?

43. Як організоване потокове введення/виведення даних в мові програмування C++?

44. Як організоване перевантаження операторів введення/виведення в мові програмування C++?

45. Як організоване файлове введення/виведення даних в мові програмування C++?

46. Що таке шаблон-функції та шаблон-класу. Як вони реалізуються в мові програмування C++?

47. Що таке STL C++. Як використовуються компоненти STL: контейнери, ітератори, алгоритми?

48. Як описати узагальнені типи в мові програмування C#. Що таке колекції Generic?

49. Що таке серіалізація в .Net? Як вона застосовується?

50. Що є делегатом та подією в .Net?

51. Що є вимогами до автоматизованої системи і програмного забезпечення? Які є основні види вимог до програмного забезпечення?

52. Хто є джерелом вимог? Хто є користувачем вимог?

53. Які є методи збору та виявлення вимог до програмного забезпечення?

54. Як можна охарактеризувати процес «Аналіз вимог»? Як застосовується уніфікована мова моделювання (UML) для аналізу вимог?

55. Які є стандарти документування вимог до програмного забезпечення?

56. Що таке специфікація вимог до програмного забезпечення?

57. Які існують значення специфікації вимог до програмного забезпечення для процесу розробки, його учасників та зацікавлених сторін?

58. Які існують принципи та прийоми управління вимогами?

59. Яка роль інженерії вимог у доменній інженерії?

60. Які є засоби управління і відстеження вимог та їх можливості?

61. Що є паттерном проектування? Як можна охарактеризувати поведінкові паттерни проектування?
62. Як можна охарактеризувати структурні паттерни проектування?
63. Як можна охарактеризувати породжуючі паттерни проектування?
64. Які є етапи створення, тестування та повторного використання компонентів і користувацьких елементів управління?
65. Які є види архітектурних паттернів?
66. Що таке архітектурний стиль? Як можна охарактеризувати архітектурні стилі?
67. Яким чином можна створювати та тестувати сервіси?
68. Як за допомогою рефлексії викликаються методи модулів, що підключаються?
69. Яким чином створюється та тестується windows-сервіс?
70. Як можна співставити функціональні вимоги до застосування з паттернами проектування?

### Список літератури

для самостійної підготовки вступника до співбесіди

#### **Основна література**

1. Роберт Седжвик. Фундаментальные алгоритмы на Java. Анализ/Структуры данных/Сортировка/Поиск. – К.: ООО "ТИД ДС", 2003. – 688с.
2. Т. Корман, Ч. Лейзерсон, Р. Ривест, К. Штайн. Алгоритмы: построение и анализ, 2-е изд. – М.: Издательский дом «Вильямс», 2007. – 1296с.

Закінчення додатку 1

3. Шилд Г. С++. Базовый курс: пер. с англ. – М.: Издательский дом «Вильямс», 2010. – 624 с.
4. Робинсон С. и др. С# для профессионалов. Т1,2. – М.: Лори, 2003. – 995 с.
5. Леффингуэлл Д., Уиндриг Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. – М.: Издательский дом «Вильямс», 2002. – 448 с.
6. Вигерс К. Разработка требований к программному обеспечению. – М.: Издательско-торговый дом «Русская редакция», 2004. – 576 с.
7. Басс Л., Клементс П., Кацман Р. Архитектура программного обеспечения на практике. 2-е издание.: пер. с англ. – СПб.: Питер, 2006. – 575с.
8. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования: пер. с англ. – СПб.: Питер, 2007. – 366 с.

#### **Додаткова література**

1. Рихтер Дж. Программирование на платформе Microsoft .Net Framework: пер. с англ. – М.: Издательско-торговый дом «Русская редакция», 2003. – 512 с.
2. Энди Кармайл, Дэн Хейвуд Быстрая и качественная разработка программного обеспечения. \_ М.: Издательский дом «Вильямс», 2009. – 263с.
3. Халл Э., Джексон К., дик Д. Разработка и управление требованиями. Практическое руководство пользователя: пер. с англ. – Лондон: Спрингер Сайнс, 2005. – 229 с.
4. Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. 2-е издание: пер. с англ. – М.: ДМК Пресс, 2007. – 496 с.