

**Зміст програмного матеріалу з дисциплін, які виносяться на
кваліфікаційний екзамен**
освітньо-професійна програма «Інженерія програмного забезпечення»

1. Теоретична частина

- 1.1. Охарактеризувати основні процеси життєвого циклу програмного забезпечення.
- 1.2. Які Ви знаєте моделі життєвого циклу ПС, надати їх коротку характеристику.
- 1.3. Збіркове програмування. Сутність та методи.
- 1.4. Розкрити сутність сервіс-орієнтованої архітектури для реалізації колективної розробки.
- 1.5. Розкрити суть технології Model Driven Architecture.
- 1.6. Розкрити суть технології Model Driven Development.
- 1.7. Розкрити онтологію поняття «інформаційна безпека», охарактеризувати її складові, навести приклад.
- 1.8. Розкрити поняття «технології соціальної інженерії», охарактеризувати її складові, навести приклад.
- 1.9. Навести функції і задачі соціального програмного забезпечення.
- 1.10. Розкрити сутність поняття «інформаційно-психологічний вплив», навести приклади.
- 1.11. Розкрити поняття «соціальне програмне забезпечення», охарактеризувати її складові, навести приклад.
- 1.12. Розкрити онтологію поняття «кібернетична безпека».
- 1.13. Сутність когнітивної науки. Математичні основи когнітивних наук.
- 1.14. Сутність системного аналізу. Методи індукції та дедукції.
- 1.15. Сутність багатокритеріальних задач прийняття рішень.
- 1.16. Розкрити сутність понять: синергетика, самоорганізація, соціотехнічна система.
- 1.17. Основні постулати ситуаційного управління.
- 1.18. Основні положення нейромережних технологій.
- 1.19. Розкрити поняття наука, науковий результат.
- 1.20. Які виділяють етапи наукових досліджень, їх зміст та сутність.
- 1.21. Форми представлення наукових результатів – тези, стаття, монографія.
- 1.22. Які існують методи наукових досліджень, розкрити їх сутність.
- 1.23. Дати класифікацію та характеристику математичних методів наукових досліджень.
- 1.24. Розкрити сутність методів моделювання.
- 1.25. Класифікація методів аналізу предметної області.
- 1.26. Семантичні моделі представлення та дослідження предметної області.
- 1.27. Термінологічний аналіз предметної області
- 1.28. Семантичний Web

- 1.29. Мови представлення онтологічних моделей - RDF/RDFS, OWL.
- 1.30. Сховища триплетів онтологічних моделей. Основні засоби мови SPARQL.
- 1.31. Процес впровадження agile-методів: умови і організація.
- 1.32. Принципи організації і робочий процес agile-методу Scrum.
- 1.33. Принципи організації і робочий процес agile-методу XProgramming.
- 1.34. Принципи організації і робочий процес agile-методу Kanban.
- 1.35. Design Thinking і побудова успішної Agile-команди, що самоорганізується.
- 1.36. Ідентифікація і категоризація ризиків в Agile-проектах.
- 1.37. Основні напрями екологізації знань та досвіду практик щодо розробки і супроводу програмного забезпечення.
- 1.38. Концепція сталого розвитку.
- 1.39. Концепція ноосфери.
- 1.40. П'ять головних принципів концепції сталого розвитку.
- 1.41. Складові концепції сталого розвитку.
- 1.42. Екологічні принципи.
- 1.43. Екологічне виробництво і використання програмного забезпечення.
- 1.44. Зелені інформаційні системи.
- 1.45. Утилізація обладнання.
- 1.46. Зелені центри обробки даних.
- 1.47. Зелені інформаційні технології.
- 1.48. Безвідходні та ресурсозберігаючі технології.
- 1.49. Безвідходне та ресурсозберігаюче виробництво програмного забезпечення.
- 1.50. Процеси утилізації програмного забезпечення.
- 1.51. Екосистема програмного забезпечення.
- 1.52. Моделювання екосистем програмного забезпечення.
- 1.53. Що таке дослідження програмного забезпечення? Основні типи і відмінності.
- 1.54. Що таке статичний аналіз? Коли він важливий, і чому?
- 1.55. Рівні абстракції дослідження програмного забезпечення.
- 1.56. Аналіз моделей.
- 1.57. Верифікація в процесах дослідження програмного забезпечення.
- 1.58. Валідація в процесах дослідження програмного забезпечення.
- 1.59. Що таке динамічний аналіз? Коли він важливий, і чому?
- 1.60. Види динамічного аналізу програмного забезпечення.

1. Теоретична частина

- 1.1. Хмарні обчислення, їх переваги.
- 1.2. Особливості технології IaaS.
- 1.3. Хмарні технології і розподілені обчислення.
- 1.4. Охарактеризувати основні процеси життєвого циклу програмного забезпечення.
- 1.5. Які Ви знаєте моделі життєвого циклу ПС, надати їх коротку характеристику.
- 1.6. Збіркове програмування. Сутність та методи.
- 1.7. Розкрити сутність сервіс-орієнтованої архітектури для реалізації колективної розробки.
- 1.8. Розкрити суть технології Model Driven Architecture.
- 1.9. Розкрити суть технології Model Driven Development.
- 1.10. Розкрити онтологію поняття «інформаційна безпека», охарактеризувати її складові, навести приклад.
- 1.11. Розкрити поняття «технології соціальної інженерії», охарактеризувати її складові, навести приклад.
- 1.12. Навести функції і задачі соціального програмного забезпечення.
- 1.13. Розкрити сутність поняття «інформаційно-психологічний вплив», навести приклади.
- 1.14. Розкрити поняття «соціальне програмне забезпечення», охарактеризувати її складові, навести приклад.
- 1.15. Розкрити онтологію поняття «кібернетична безпека».
- 1.16. Сутність когнітивної науки. Математичні основи когнітивних наук.
- 1.17. Сутність системного аналізу. Методи індукції та дедукції.
- 1.18. Сутність багатокритеріальних задач прийняття рішень.
- 1.19. Розкрити сутність понять: синергетика, самоорганізація, соціотехнічна система.
- 1.20. Основні постулати ситуаційного управління.
- 1.21. Основні положення нейромережних технологій.
- 1.22. Розкрити поняття наука, науковий результат.
- 1.23. Методи і рівні дослідження культури інженерії програмного забезпечення.
- 1.24. Модель культури інженерії програмного забезпечення.
- 1.25. Правила побудови діаграми гнучкості для візуалізації обмежень в проекті.
- 1.26. Дати класифікацію та характеристику математичних методів наукових досліджень.
- 1.27. Модель прийняття рішення. Якість прийняття рішень.
- 1.28. Класифікація методів аналізу предметної області.
- 1.29. Семантичні моделі представлення та дослідження предметної області.
- 1.30. Термінологічний аналіз предметної області.

- 1.31. Семантичний Web.
- 1.32. Мови представлення онтологічних моделей – RDF/RDFS, OWL.
- 1.33. Сховища триплетів онтологічних моделей. Основні засоби мови SPARQL.
- 1.34. Процес впровадження agile-методів: умови і організація.
- 1.35. Принципи організації і робочий процес agile-методу Scrum.
- 1.36. Принципи організації і робочий процес agile-методу XProgramming.
- 1.37. Принципи організації і робочий процес agile-методу Kanban.
- 1.38. Design Thinking і побудова успішної Agile-команди, що самоорганізується.
- 1.39. Ідентифікація і категоризація ризиків в Agile-проектах.
- 1.40. Сучасні системи підтримки прийняття рішень (Decision Support System, DSS).
- 1.41. Концепція сталого розвитку.
- 1.42. Розподіл ролей в команді. Стадії формування згуртованої групи. Модель групи.
- 1.43. П'ять головних принципів концепції сталого розвитку.
- 1.44. Складові концепції сталого розвитку.
- 1.45. Екологічні принципи.
- 1.46. Екологічне виробництво і використання програмного забезпечення.
- 1.47. Зелені інформаційні системи.
- 1.48. Утилізація обладнання.
- 1.49. Зелені центри обробки даних.
- 1.50. Зелені інформаційні технології.
- 1.51. Безвідходні та ресурсозберігаючі технології.
- 1.52. Безвідходне та ресурсозберігаюче виробництво програмного забезпечення.
- 1.53. Процеси утилізації програмного забезпечення.
- 1.54. Екосистема програмного забезпечення.
- 1.55. Моделювання екосистем програмного забезпечення.
- 1.56. Що таке дослідження програмного забезпечення? Основні типи і відмінності.
- 1.57. Що таке статичний аналіз? Коли він важливий, і чому?
- 1.58. Рівні абстракції дослідження програмного забезпечення.
- 1.59. Аналіз моделей.
- 1.60. Верифікація в процесах дослідження програмного забезпечення.