

## **Тема 1.**

### **Сучасні структури багаторівневих АСУТП. Робототехнічні комплекси у складі АСУТП**

1. Які рівні входять до типових багаторівневих АСУТП та їх основні функції?
2. У чому полягають переваги багаторівневої структури управління технологічними процесами?
3. Які вимоги висуваються до інтеграції робототехнічних комплексів у АСУТП?
4. Приклади застосування робототехнічних систем у сучасних виробництвах.
5. Як забезпечується взаємодія між рівнями АСУТП (датчики – контролери – SCADA)?

## **Тема 2.**

### **Функції, завдання програмного забезпечення АСУТП**

1. Які основні функції виконує програмне забезпечення в системах АСУТП?
2. Чим відрізняється системне ПЗ від прикладного ПЗ в АСУТП?
3. Які завдання вирішуються засобами ПЗ під час моніторингу та керування процесами?
4. Як забезпечується надійність і безперервність роботи ПЗ АСУТП?
5. Які вимоги висуваються до інтерфейсу оператора в програмному забезпеченні?

## **Тема 3.**

### **Сучасні Softlogic- та SCADA-системи: поняття, функції, компоненти, середовища розробки**

1. Що таке Softlogic-системи та у чому їхні переваги над традиційними ПЛК?
2. Основні функції SCADA-систем у контексті управління технологічними процесами.
3. Які компоненти входять до типової SCADA-системи?
4. Порівняння середовищ розробки: Siemens WinCC, Wonderware, Trace Mode.
5. Як реалізується обмін даними між SCADA та польовими пристроями?

#### **Тема 4.**

#### **Середовища для створення прикладних програм мовами високого рівня**

1. Які середовища (IDE) використовуються для розробки промислового ПЗ (Visual Studio, Eclipse тощо)?
2. У чому переваги високорівневих мов під час створення додатків АСУТП?
3. Як організується структура проекту прикладної програми?
4. Що таке SDK та його роль у промисловій автоматизації?
5. Які вимоги до тестування прикладного ПЗ у виробничих умовах?

#### **Тема 5.**

#### **Основні поняття FBD-програмування**

1. Що таке FBD та які його ключові елементи (блоки, лінії зв'язку)?
2. Як у FBD реалізуються логічні операції (AND, OR, NOT)?
3. Приклади реалізації алгоритмів регулювання у FBD.
4. Переваги графічного програмування над текстовим у ПЛК.
5. Як у FBD реалізуються циклічні та часові функції?

#### **Тема 6.**

## **Блоки моделювання, регулювання та управління**

1. Які типи функціональних блоків застосовуються для регулювання (PID, Relay тощо)?
2. Як здійснюється моделювання технологічного процесу у SCADA/PLC-системах?
3. Чим відрізняються блоки управління від блоків регулювання?
4. Як враховується інерційність об'єкту в алгоритмах управління?
5. Приклади використання блоків зворотного зв'язку.

### **Тема 7.**

#### **Створення САР засобами SCADA-системи**

1. Які етапи побудови САР у SCADA (створення змінних, трендів, PID-блоків)?
2. Як реалізується зворотний зв'язок у SCADA-проєкті?
3. Що таке НМІ-панель і її роль у САР?
4. Як налаштовуються аварійні повідомлення та захист у системі керування?
5. Приклад САР для підтримання температури/тиску.

### **Тема 8.**

#### **Особливості створення програм мовою програмування LD**

1. Що таке LD (Ladder Diagram) і де вона використовується?
2. Як у LD реалізується логіка контакторів та реле?
3. Приклади організації міжблокувань та аварійних схем.
4. Особливості таймерів і лічильників у LD.
5. Порівняння LD з FBD та ST.

## **Тема 9.**

### **Основні поняття мови програмування ST**

1. Що таке Structured Text (ST) та його синтаксис?
2. Як у ST реалізуються умови та цикли?
3. Підтримка користувацьких функцій та FB.
4. Приклади математичних операцій у ST.
5. Особливості роботи зі змінними та типами даних.

## **Тема 10.**

### **Особливості створення програм мовою програмування ST**

1. Як структурувати програму ST у PLC-проєкті?
2. Реалізація ПІД-регулятора мовою ST.
3. Як у ST обробляються помилки та винятки?
4. Приклади роботи з масивами та структурами.
5. Оптимізація продуктивності ST-коду.

## **Тема 11.**

### **Особливості створення програм мовами високого рівня**

1. Які мови використовуються у SCADA (C#, Python, VBScript)?
2. Як реалізуються драйвери та API для промислових пристроїв?
3. Особливості роботи зі змінними реального часу.
4. Чим відрізняється НМІ-логіка від бекенд-логіки?
5. Використання бібліотек та модулів.

## **Тема 12.**

### **Розробка концепції та створення алгоритму САУ**

1. Етапи розробки концепції САУ (аналіз об'єкта, постановка завдань).

2. Як описати алгоритм управління мовою блок-схем?
3. Критерії вибору стратегії регулювання (ПІД, нечітка логіка).
4. Як враховуються аварійні режими у концепції?
5. Документування та верифікація алгоритму.

### **Тема 13.**

#### **Особливості створення програм мовою програмування SFC**

1. Що таке SFC (Sequential Function Chart) та його структура?
2. Як описуються переходи між станами?
3. Типові застосування SFC у послідовних процесах.
4. Взаємодія SFC з іншими мовами IEC 61131-3.
5. Приклади реалізації циклічного процесу у SFC.

### **Тема 14.**

#### **Прикладні програми для дистанційного та автоматичного режимів роботи САУ**

1. Вимоги до інтерфейсу дистанційного управління.
2. Як реалізується перемикання режимів (Manual/Auto)?
3. Забезпечення захисту та доступу користувачів.
4. Приклади реалізації SCADA/HMI для віддаленого моніторингу.
5. Особливості передачі даних (Modbus, OPC-UA).