

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ПРИРОДОКОРИСТУВАННЯ**

**ФАКУЛЬТЕТ МЕХАНІКИ, ЕНЕРГЕТИКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА

першого (бакалаврського) рівня вищої освіти

на тему: **“Розробка інформаційної системи із елементами
імітаційного моделювання технологічних операцій у виробництві”**

Виконав: студент гр. Іт-42сп _____
Спеціальності 126 – «Інформаційні системи та
технології» _____
(шифр і назва)

_____ **Солонинка Юрій Ярославович** _____
(Прізвище та ініціали)

Керівник: _____ **к.т.н., доц. Луб П.М.** _____
(Прізвище та ініціали)

Рецензенти: _____ **к.т.н., доц. Бабич М.І.** _____
(Прізвище та ініціали)

(Прізвище та ініціали)

ДУБЛЯНИ-2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ПРИРОДОКОРИСТУВАННЯ
ФАКУЛЬТЕТ МЕХАНІКИ, ЕНЕРГЕТИКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Перший (бакалаврський) рівень вищої освіти

126 – «Інформаційні системи та технології»

“ЗАТВЕРДЖУЮ”

Завідувач кафедри _____
д.т.н., проф. А.М. Тригуба
“ _____ ” _____ 2023 р.

ЗАВДАННЯ

на кваліфікаційну роботу студенту

Солонинці Юрію Ярославовичу

1. Тема роботи: «Розробка інформаційної системи із елементами імітаційного моделювання технологічних операцій у виробництві»

Керівник роботи Луб Павло Миронович, к.т.н., доцент
Затверджені наказом університету 27.11.2023 року №641/к-с.

2. Строк подання студентом роботи 14.06.2024 р.

3. Початкові дані до роботи: 1. Офіційні дані щодо виробничої діяльності підприємств. 2. Науково-технічна і довідкова література. 3. Методика відображення технологічних процесів імітаційною моделлю. 4. Методика функційного програмування.

4. Зміст розрахунково-пояснювальної записки:

1. Аналіз процесів планування інформаційних систем
2. Постановка задачі проектування інформаційної системи
3. Проектування інформаційної системи та формування початкових даних
4. Практичне використання програмного додатку імітаційного моделювання технологічних операцій
5. Висновки
6. Бібліографічний список.
7. Додатки

5. Перелік презентаційного матеріалу : 1 та 2 – Тема, мета, завдання роботи; 3 – Аналіз структури інформаційно-аналітичних систем; 4 – Послуги агросектору із застосуванням ІТ; 5 – Інформаційно-аналітичні системи які використовують на практиці; 6 – Графічне представлення способу відображення технологічних операцій моделлю; 7 – Блок-схема укрупненого алгоритму; 8 – Інтерфейс прикладної програми; 9 та 10 – Результати моделювання технологічних операцій.

6. Консультанти з розділів:

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|------------|--|----------------|------------------|
| | | завдання видав | завдання прийняв |
| 1, 2, 3, 4 | <i>Луб П.М., доцент кафедри інформаційних технологій</i> | | |
| 5 | <i>Городецький І.М., доцент кафедри фізики, інженерної механіки та безпеки виробництва</i> | | |

7. Дата видачі завдання 27.11.2023 р.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Етапи виконання кваліфікаційної роботи | Строк виконання етапів роботи | Примітка |
|-------|--|--------------------------------|----------|
| 1. | <i>Написання першого розділу та означення головних завдань роботи</i> | <i>27.11.2023 – 01.01.2024</i> | |
| 2. | <i>Виконання другого розділу та формування головних показників для розрахунків</i> | <i>01.01.2024 – 01.02.2024</i> | |
| 3. | <i>Виконання третього розділу, розрахунків та розробка листів</i> | <i>01.02.2024 – 01.03.2024</i> | |
| 4. | <i>Написання розділу: «Охорона праці та безпека в надзвичайних ситуаціях»</i> | <i>01.03.2024 – 01.04.2024</i> | |
| 5. | <i>Вартісне оцінення ефективності пропозицій роботи</i> | <i>01.04.2024 – 01.05.2024</i> | |
| 6. | <i>Завершення оформлення розрахунково-пояснювальної записки та аркушів графічної частини</i> | <i>01.05.2024 – 01.06.2024</i> | |
| 7. | <i>Завершення роботи в цілому</i> | <i>01.06.2024 – 14.06.2024</i> | |

Студент _____ Солонинка Ю.Я.
(підпис)

Керівник роботи _____ Луб П.М.
(підпис)

УДК: 004.94: 631.1

Кваліфікаційна робота: 56 с. текст. част., 16 рис., 3 табл., 10 слайдів, 17 джерел.

Розробка інформаційної системи із елементами імітаційного моделювання технологічних операцій у виробництві. Солонинка Ю.Я. Кафедра ІТ. – Дубляни, Львівський НУП, 2024.

Проаналізовано інформаційно-аналітичні системи планування виробничих процесів. Сформовано задачу створення інформаційної системи із елементами імітаційного моделювання технологічних операцій у виробництві.

Охарактеризовано інформаційно-аналітичні системи планування та моніторингу виробничих проектів АПК. Наведено особливості предметної галузі та вимоги до системи планування робіт.

Описано спосіб відображення технологічних операцій виробництва у імітаційній моделі. Сформовано початкову базу даних.

Розроблено блок-схему алгоритму імітаційної моделі технологічних операцій збирання врожаю культури. Розроблено прикладну програму імітаційного моделювання технологічних операцій.

Представлено результати використання прикладної програми та можливість планування змісту проектів збирання врожаю.

Показано як використання розробленої прикладної програми дає змогу реалізовувати роботу інформаційної системи для супровід рішень у плануванні процесів збирання врожаю культури.

Розроблено заходи з охорони праці та безпека в надзвичайних ситуаціях.

Ключові слова: інформаційна система, імітаційне моделювання, технологічні операції, програмне забезпечення, оптимізація, ефективність.

Keywords: information system, simulation modeling, technological operations, software, optimization, efficiency.

ЗМІСТ

| | |
|---|----|
| ВСТУП | 7 |
| РОЗДІЛ 1 | |
| АНАЛІЗ ПРОЦЕСІВ ПЛАНУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ... | 8 |
| 1.1. Аналіз систем підтримки прийняття рішень..... | 8 |
| 1.2. Аналіз програмних засобів для розробки прикладних програм . | 11 |
| 1.3. Системи планування виробничих проектів в АПК..... | 12 |
| РОЗДІЛ 2 | |
| ПОСТАНОВКА ЗАДАЧІ ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ..... | 16 |
| 2.1. Технологія структурного аналізу та проектування ІС..... | 16 |
| 2.2. Вимоги до інформаційно-аналітичної системи планування робіт..... | 18 |
| РОЗДІЛ 3 | |
| ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ТА ФОРМУВАННЯ ПОЧАТКОВИХ ДАНИХ..... | 22 |
| 3.1. Опис інструментів та мови програмування..... | 22 |
| 3.2. Відображення технологічних операцій моделлю..... | 24 |
| 3.3. Розробка блок-схеми алгоритму прикладної програми..... | 27 |
| РОЗДІЛ 4. | |
| ПРАКТИЧНЕ ВИКОРИСТАННЯ ПРОГРАМНОГО ДОДАТКУ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ ТЕХНОЛОГІЧНИХ ОПЕРАЦІЙ..... | 33 |
| 4.1. Імітаційне моделювання технологічних операцій | 33 |
| 4.2. Результати підтримки прийняття рішень із оптимізації сезонної площі для бурякозбирального комбайном..... | 35 |
| РОЗДІЛ 5. | |
| ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ... | 38 |
| 5.1. Структурно-функціональний аналіз технологічного процесу... | 38 |
| 5.2. Розрахунок освітлення приміщення комп'ютерного кабінету... | 39 |
| 5.3. Безпека в надзвичайних ситуаціях..... | 42 |
| ВИСНОВКИ І РЕКОМЕНДАЦІЇ..... | 43 |
| БІБЛІОГРАФІЧНИЙ СПИСОК..... | 44 |
| ДОДАТКИ..... | 46 |

ВСТУП

Створення інформаційно-аналітичних систем, що відповідають цілям і завданням підприємств аграрного сектору, є достатньо складним процесом, що включає етапи формування концепції, проектування, розробки, впровадження і супроводу рішень. Для цього необхідно володіти загальною методикою створення інформаційно-аналітичних систем, що містить склад і послідовність робіт і завдань, склад ролевих функцій, документів, моделей, схем тощо [8, 14].

Слід також відмітити системи підтримки прийняття рішень (СППР), що формують інтерактивні структури до складу яких входить – обладнання, програмне забезпечення, дані, база моделей і робота менеджера. СППР – комплекс програмних засобів, що включає комплекс різних алгоритмів підтримки рішень, базу моделей, базу даних, допоміжні та керівну програми і т.д. СППР використовується для підтримки різних видів діяльності у процесі прийняття рішень – надання допоміжної інформації, особливо для виконання неструктурованих або слабоструктурованих завдань, для яких важко заздалегідь визначити дані та процедури відповідних рішень

Мета роботи – підвищити ефективність проектів збирання врожаю культур на підставі застосування інформаційної системи підтримки прийняття рішень із планування робіт у цих проектах.

Завдання кваліфікаційної роботи:

- проаналізувати процеси планування інформаційних систем;
- сформулювати вимоги до проектування інформаційної системи;
- запроектувати інформаційну систему та сформулювати початкові дані;
- виконати моделювання технологічних операцій.

РОЗДІЛ 1

АНАЛІЗ ПРОЦЕСІВ ПЛАНУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ

1.1. Аналіз систем підтримки прийняття рішень

СППР являє собою взаємодіючу з іншими системами комп'ютеризовану систему для надання допомоги менеджерам у процесі прийняття рішень. СППР допомагає менеджерам знаходити, обчислювати і аналізувати дані, що відносяться до рішення, яке приймається. Системи підтримки прийняття рішень (Decision Support Systems — DSS), належать до інформаційних систем нового покоління, головне призначення яких полягає в забезпеченні комп'ютерною підтримкою прийняття рішень зі слабоструктурованих та неструктурованих проблем організаційного управління на різних етапах підготовки рішень і моніторингу. Незважаючи на те, що на даний час у світі розроблено сотні типів СППР, такі системи в Україні практично не використовуються.

В класичному розумінні СППР є інформаційною системою, котра має такі компоненти, які зображені на рис. 1.1: інтерфейс користувача, систему керування базами даних (СКБД), систему керування базами моделей (СКБМ), систему керування повідомленнями (СКП), причому підсистема СКП з'явилася лише останніми роками. Ця структура може бути основою для ідентифікації наявних СППР.

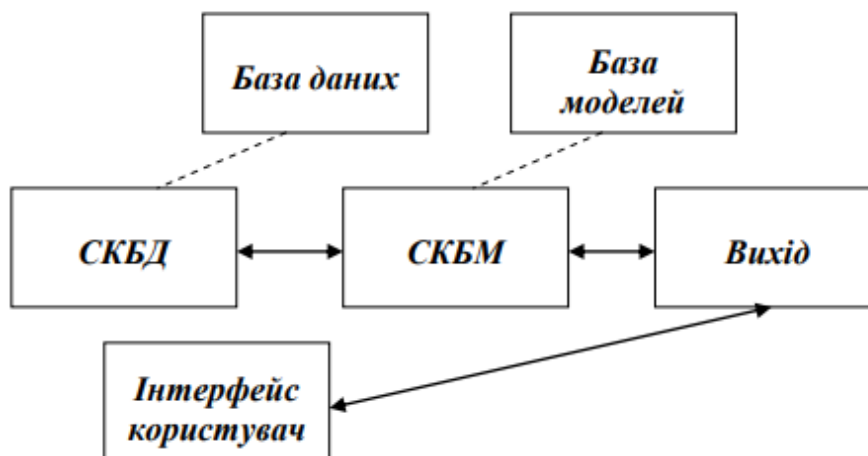


Рис. 1.1. Класична структура СППР: СКБД – система керування базою даних; СКБМ – система керування базою моделей

Ці три підсистеми утворюють основу класичної структури СППР, завдяки якій останні відрізняються від інших типів інформаційних систем. Останнім часом з розвитком глобальної мережі Інтернет, корпоративних (Інтранет) та міжорганізаційних (Ентернет) мереж до СППР додають нову підсистему – систему керування повідомленнями (комунікаціями або зв'язком) – СКП. Окремі компоненти цих підсистем зображені на рис. 1.2. Компоненти СППР відображають скоріше функціональний, а не формальний поділ системи на окремі підсистеми з погляду її проектування, тобто на перший план виступає питання стосовно того, що буде робити дана СППР, зокрема, використовуючи поняття її архітектури, передусім База даних, База моделей, СКБД, СКБМ, Вихід, Інтерфейс, користувач, а створюють користувацький інтерфейс, систему керування даними і систему керування моделями.

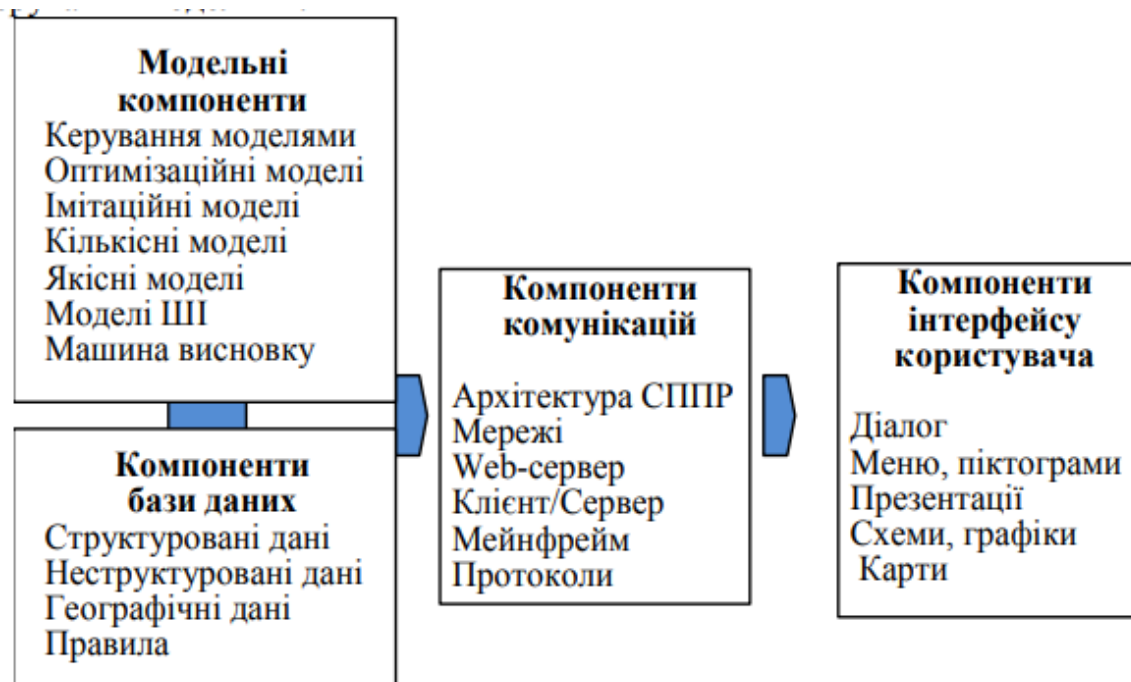


Рис. 1.2. Компоненти підсистем СППР

Аналіз різних поглядів на розроблення і застосування комп'ютерних систем підтримки прийняття рішень, на способи одержання, подання і структуризації інформації, на специфічні відмінності СППР від інших типів інформаційних систем дає змогу виділити для класифікації СППР ряд класифікаційних ознак-підходів для поділу всієї сукупності систем на класифікаційні групи (табл. 1.1).

Таблиця 1.1. Класифікація СППР

| Категорії класифікації | Ознака (основа) класифікації | Класифікаційні групи (типи систем) |
|--|---|--|
| Концептуальна модель | Інформаційний підхід | Концептуальна модель Спрага Модель еволюціонуючої СППР |
| | Підхід, оснований на знаннях | Орієнтовані на знання СППР Орієнтовані на правила СППР |
| | Інструментальний підхід | Спеціалізовані (прикладні) СППР СППР-генератори СППР -інструментарії |
| Користувачі | Ієрархічний рівень управління | Вища ланка управління (виконавчі інформаційні системи) Середня ланка управління Нижча ланка управління |
| | Спосіб взаємодії користувача з системою | Термінальний режим Режим клерка Режим посередника Автоматизований режим |
| | Ступінь залежності осіб у процесі прийняття рішення | Персональна підтримка (персональні СППР) Групова підтримка (групові СППР) Організаційна підтримка (багатокористувацькі, інтер-організаційні, інтра-організаційні СППР) |
| Завдання, що потребує прийняття рішень | Новизна завдання | Унікальні проблеми (СППР на даний випадок (ad hoc)) Повторювані проблеми (інституціональні СППР) |
| | Характер опису проблеми | Цілісний вибір Багатокритеріальний вибір (наприклад, СППР Decision Grid) |
| | Тип моделі | Об'єктивна модель Суб'єктивна модель |
| | Діапазон підтримуваних функцій | Функціонально-специфічні СППР СППР загального призначення |
| Забезпечуючі засоби | Рівень підтримки прийняття рішень | СППР, орієнтовані на дані СППР, орієнтовані на моделі СППР, орієнтовані на документи СППР, орієнтовані на комунікації Web-орієнтовані СППР |
| | Рівень користувацького інтерфейсу мов | Процедурні мови Командні мови Непроцедурні мови Природні мови |
| Галузі застосування | Професійна сфера | Мікроекономіка Макроекономіка Конторська діяльність (офісні СППР) Оцінювання розповсюдження технологій Юриспруденція Медицина і т. ін. |
| | Часовий горизонт | Стратегічне управління (довгострокові рішення) Тактичне управління (середньострокові рішення) Операційне управління (короткострокові рішення) |

Також в інформаційному просторі можна знайти СППР, що допомагають обирати маршрути, управляти портфелями акцій, вибирати напрямки інвестування, планувати подорожі. Перелік найвідоміших «комерційних» СППР містить сотні назв. Загальний обсяг продажу на ринку СППР перевищує 10 мільярдів доларів [1, 17].

1.2. Аналіз програмних засобів для розробки прикладних програм

На сьогоднішній день існує багато прикладних середовищ програмування. Усі вони мають різні інтерфейси, орієнтовані на різні мови програмування, мають різні набори можливостей. Всі вони мають як і плюси, так і мінуси.

Одним з найпопулярніших середовищ програмування сьогодні є оболонка Visual Studio, розроблена корпорацією Microsoft. Вона має широкий спектр можливостей, дозволяє створювати додатки для Windows, Android та Linux мовами програмування C, C++, C#, F#, JavaScript, Visual Basic та Python.

IntelliJ IDEA – середовище розробки, яке розробляється компанією JetBrains та орієнтоване на роботу з мовами програмування Java та Kotlin. Kotlin - це мова програмування, яка появилася відносно недавно та працює на віртуальній машині Java. Середовище IntelliJ IDEA дозволяє створювати додатки для багатьох операційних систем: Windows, Linux, MacOS та Android. Остання є найбільш популярною, оскільки IDEA має вбудовані інструменти для розробки додатків Android.

PyCharm - ще одне середовище від компанії JetBrains для розробки мовою програмування Python. Дана оболонка підтримує розробку звичайних додатків, web-додатків та проведення дослідень.

Eclipse – середовище розробки, розроблене Eclipse Foundation, яке в першу чергу орієнтоване на Java-розробників, але також вміє працювати із Javascript, Rust, Php, C, та C++. Перевагою цієї оболонки є легкість та невимогливість до системи.

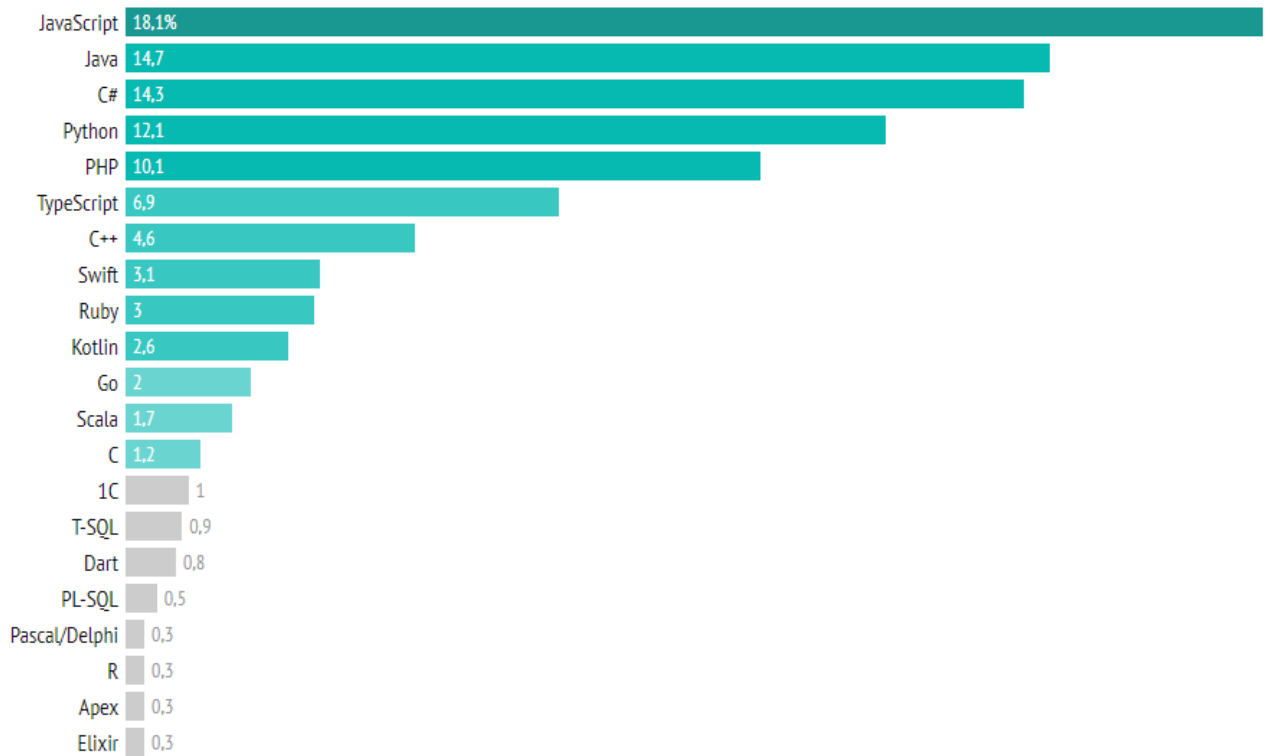


Рис. 1.3. Популярність мов програмування (дані Українського профільного ресурсу DOU.UA, 2023 р.)

Code::Blocks – безкоштовне середовище програмування з відкритим вихідним кодом для розробки мовами програмування C, C++ та Fortran. Перевагою цієї оболонки також є невимогливість до системи.

Отже, на сьогоднішній день існує дуже багато різноманітних засобів розробки програмного забезпечення, всі вони мають різні можливості та дозволяють працювати з різними мовами програмування.

1.3. Системи планування виробничих проектів в АПК

Сьогодні значна частина агрохолдингів прийшла до такого етапу свого розвитку, що конкуренція на ринку АПК змушує їх використовувати не тільки продуктивні технології виробництва продукції із застосуванням дедалі нових сортів, добрив та хімічного захисту, але й використання сучасних ІТ для

ефективного використання наявних ресурсів та управління виробництвом загалом.

Переважно, такі системи є платними і доволі важко досяжні для менших господарств. Вони також є дещо узагальненими та не розглядають особливості виробничих процесів на рівні окремих технологічних операцій.

Цю прогалину можна заповнити розробкою спеціальних прикладних програм для підтримки прийняття рішень – цьому завданню і присвячено кваліфікаційну роботу.

Отже, доволі широкого застосування зазнала система Cropio – це система дистанційного контролю сільськогосподарських угідь, що дозволяє здійснювати оперативний моніторинг стану посівних площ, авто документування, прогнозування і планування сільськогосподарських операцій. За родом діяльності компанія відноситься до операторів послуг супутникового моніторингу посівів. Система дозволяє контролювати рівень вегетації посівів, вміст поживних речовин і вологість ґрунту, отримувати актуальну інформацію про погоду і динаміці цін, отримувати повідомлення про суттєві зміни на полях. Система Cropio підходить для моніторингу всіх без винятку зернових і олійних культур. При використанні системи для таких культур, як цукрові буряки, необхідно враховувати той факт, що принцип дії заснований на аналізі насиченості рослини хлорофілом, що не завжди передає об'єктивну інформацію про стан плода, що знаходиться в ґрунті.



Рис. 1.4. Система управління агропідприємством – Cropio

Дещо схожим є інформаційний сервіс FarmingOS – це платформа для візуалізації аграрного бізнесу. За допомогою FarmingOS можна приймати рішення ґрунтуючись на реальних цифрах, доступних в один клік. За її використання операційний прогрес, витрати, доходи і маржинальність доступні в розрізі будь-якого проміжку часу, полів, культур і основних видів діяльності тощо.



Рис. 1.5. Платформа для візуалізації аграрного бізнесу – FarmingOS

Застосування FarmingOS дає змогу:

- 1) спостерігати за станом аграрного підприємства (сівозміна в розрізі сезонів, витрати, доходи та маржинальність, площа і стан кожного типу культур, статус виконання робіт);
- 2) слідкувати за динамікою виконання робіт (план-факт виконання агрооперацій, фільтр за культурами, видами робіт на полях, вибір сезону і типу прямих витрат тощо);
- 3) набір інструментів для аналізу планових і фактичних витрат, а також моніторингу накопичених витрат (середня вартість за 1 га, планові витрати, фактичні витрати, фільтри по культурам, видам робіт і полях тощо).

Активно просуваються також інформаційно-аналітичний сервіс – 1С AGRO WEB. Це система управлінського обліку агропідприємства – яка адаптувала досвід найбільших аграрних підприємств під потреби клієнтів. Тут акцентується

на максимально повних обсягах даних в інтуїтивно зрозумілому інтерфейсі. Це дані що необхідні для аналізу ефективності роботи компанії, включаючи собівартість гектара поля і динаміку витрат тощо.

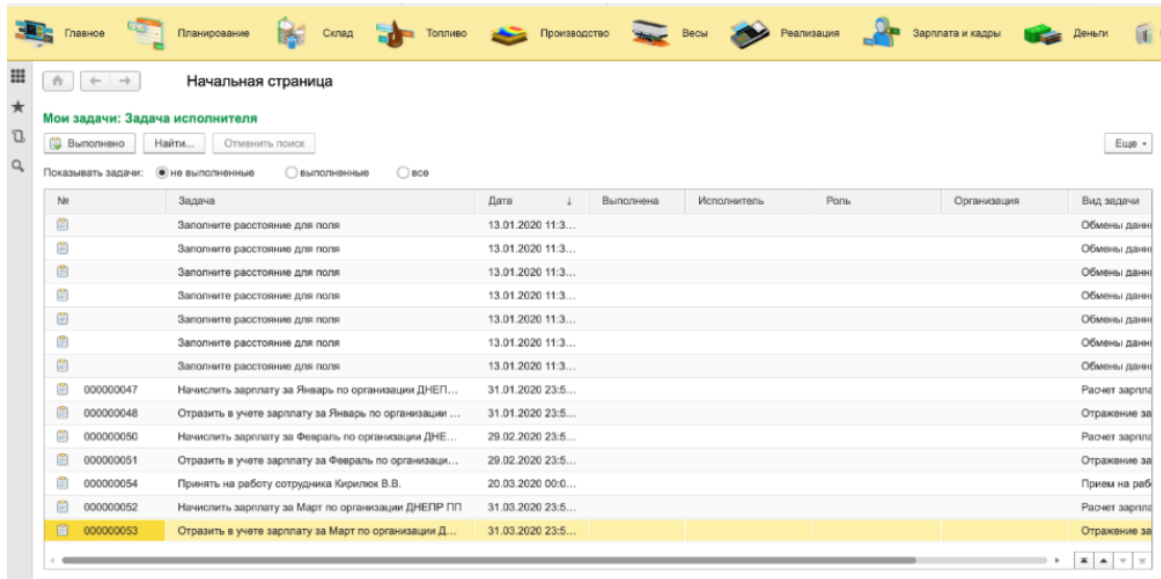


Рис. 1.6. Робоче вікно 1C AGRO WEB

Розгляд цих інформаційних систем дає підстави стверджувати що основну увагу приділено бізнес-процесам та управління ними. Однак завдання що стосуються планування використання ресурсів підприємства та, зокрема, планування робіт із застосуванням відповідних машин для виконання польових робіт (комбайнів для збирання врожаю культур) нажаль відсутні.

РОЗДІЛ 2

ПОСТАНОВКА ЗАДАЧІ ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Технологія структурного аналізу та проектування ІС

В методології структурного аналізу традиційно використовують моделі, що показують:

- функції, які система повинна виконувати;
- процеси, що забезпечують виконання функцій;
- дані, що необхідні під час виконання функцій, і відношення між ними;
- організаційні структури, що забезпечують виконання функцій;
- матеріальні та інформаційні потоки, що виникають у ході виконання функцій.

Як інструментальні засоби структурного аналізу і проектування найбільш часто і ефективно застосовуються наступні.

BFD (Business Function Diagrams) – діаграми бізнес-функцій (функціональні специфікації).

DFD (Data Flow Diagrams) – діаграми потоків даних у нотаціях Гейна – Сарсона, Йордана та інших, що забезпечують аналіз і функціональне проектування інформаційних систем.

ERD (Entity-Relationship Diagrams) – діаграми "сутність – зв'язок" у нотаціях Чена и Баркера.

STD (State Transition Diagrams) – діаграми переходів станів, які засновані на розширеннях Уорда – Меллора (Ward – Mellor) і Хатлі (Hatley) для проектування систем реального часу.

SSD (System Structure Diagrams) – діаграми структури програмного додатка.

FDD (Functional Decomposition Diagrams) – діаграми функціональної декомпозиції.

Сімейство **IDEF** (*Integration Definition for Function Modeling*):

IDEF0 – методологія функціонального моделювання, що є складовою частиною SADT і що дозволяє описати бізнес-процес у вигляді ієрархічної системи взаємопов'язаних функцій.

IDEF1 – методологія аналізу і вивчення взаємозв'язків між інформаційними потоками в рамках комерційної діяльності підприємства.

IDEF1X – методологія інформаційного моделювання, заснована на концепції "сутність – зв'язок" Ченом. Застосовується для розробки реляційних баз даних і використовує умовний синтаксис, що спеціально розроблений для зручної побудови концептуальної схеми і забезпечує універсальне подання структури даних в рамках підприємства, незалежне від кінцевої реалізації бази даних і апаратної платформи;

IDEF3 – методологія моделювання потоків робіт підприємства, що дозволяє подати їх сценарії за допомогою опису послідовності змін властивостей об'єкта в рамках даного процесу.

IDEF4 – методологія об'єктно-орієнтованого проектування для підтримки проектів, пов'язаних з об'єктно-орієнтованими реалізаціями.

IDEF5 – методологія, що забезпечує наочне подання даних, отриманих у результаті обробки онтологічних запитів, у простій, графічній формі.

Усі перелічені засоби містять графічні і текстові засоби моделювання: перші – для зручності відображення основних компонентів моделі, другі – для забезпечення точного визначення її компонентів і зв'язків.

Методології структурного аналізу і проектування можна класифікувати наступним чином (рис. 6.1).

За відношенням до шкіл виділяють методології програмної інженерії (*Software Engineering, SE*) та інформаційної інженерії (*Information Engineering, IE*).

SE є нисхідним поетапним підходом до розробки ПЗ, що починається із загального погляду на його функціонування. Потім проводиться декомпозиція на підфункції. Процес повторюється доти, поки підфункції не стануть достатньо малими для можливості їх кодування. У результаті виходить ієрархічна,

структурована модульна програма. SE подає універсальні методології розробки ПЗ.

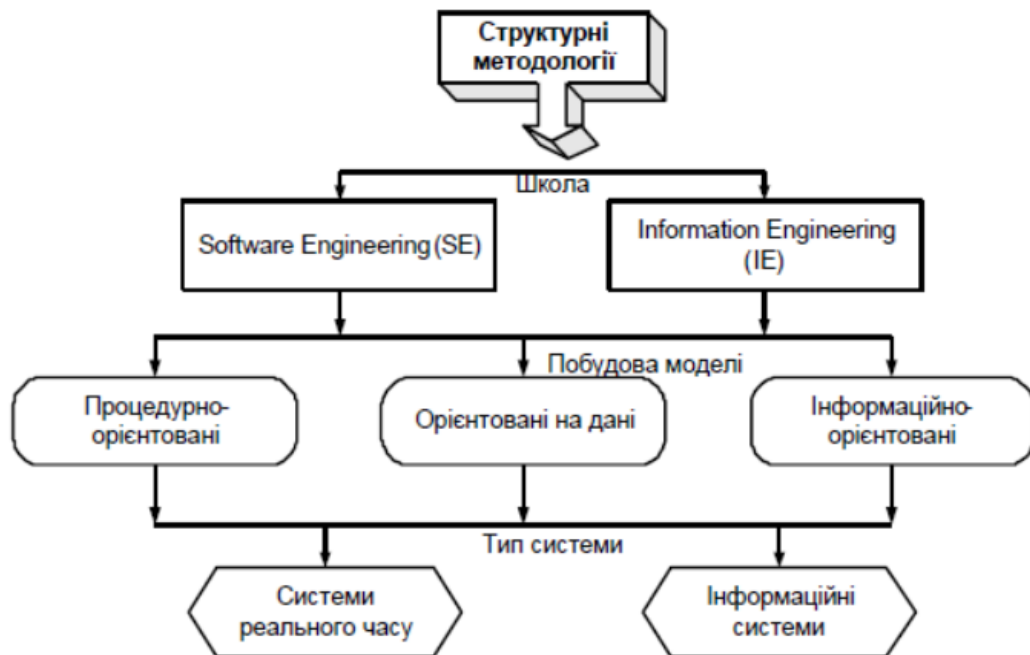


Рис. 2.1. – Класифікація структурних методологій

ІЕ представляє нові методології побудови систем взагалі і включає етапи більш високого рівня (наприклад, стратегічне планування). На етапі проектування систем за методологією SE і IE аналогічні.

2.2. Вимоги до інформаційно-аналітичної системи планування робіт

Для формування завдань та розробки головних блоків інформаційно-аналітичної системи потрібно чітко виокремити особливості виробничих процесів, а також означити процеси та явища які ідеалізуються. Тому опишемо особливості технологічного процесу збирання врожаю коренеплодів цукрових буряків.

Особливістю досягання цукрових буряків є те, що приріст маси (Δm) коренеплодів відбувається в осінній період і може тривати до початку заморозків (часу (τ_{ϕ}^3) завершення фізичної стиглості ґрунту, або часу (τ^{-5}) виникнення

заморозків нижче -5°C). Надто ранні терміни початку (τ_{np}) бурякозбиральних робіт, за яких ще відбувається поточний приріст маси (m_n) коренеплідів, зумовлюють порівняно менший збір урожаю Q_{ϕ} (рис. 2.2).

Відповідно до цього справедливою буде умова – $\Delta m_1 > \Delta m_2 > \Delta m_3$ та $m_{n1} < m_{n2} < m_{n3}$ у результаті виконання бурякозбиральних робіт за різних τ_{np1} , τ_{np2} , τ_{np3} отримують $E_1 < E_2 < E_3$. Зниження Q_{ϕ} для варіанта часу початку робіт за τ_{np1} порівняно з τ_{np2} та τ_{np3} пояснюється зниженням інтенсивності (Δm) приросту маси коренеплідів. Відповідно до цього мінімальні біологічні втрати врожаю досягаються за умови пізніх термінів τ_{np} :

$$Q_{\phi} = f(\tau_{np}, \underbrace{W_{\text{дооб}}, S^n}_{t_{mn}}, m_n, \Delta m). \quad (2.1)$$

Темпи приросту Δm впливають на планування термінів початку робіт зі збирання врожаю. Користуючись відомими даними досліджень [6], закономірність Δm для різних років є змінною.

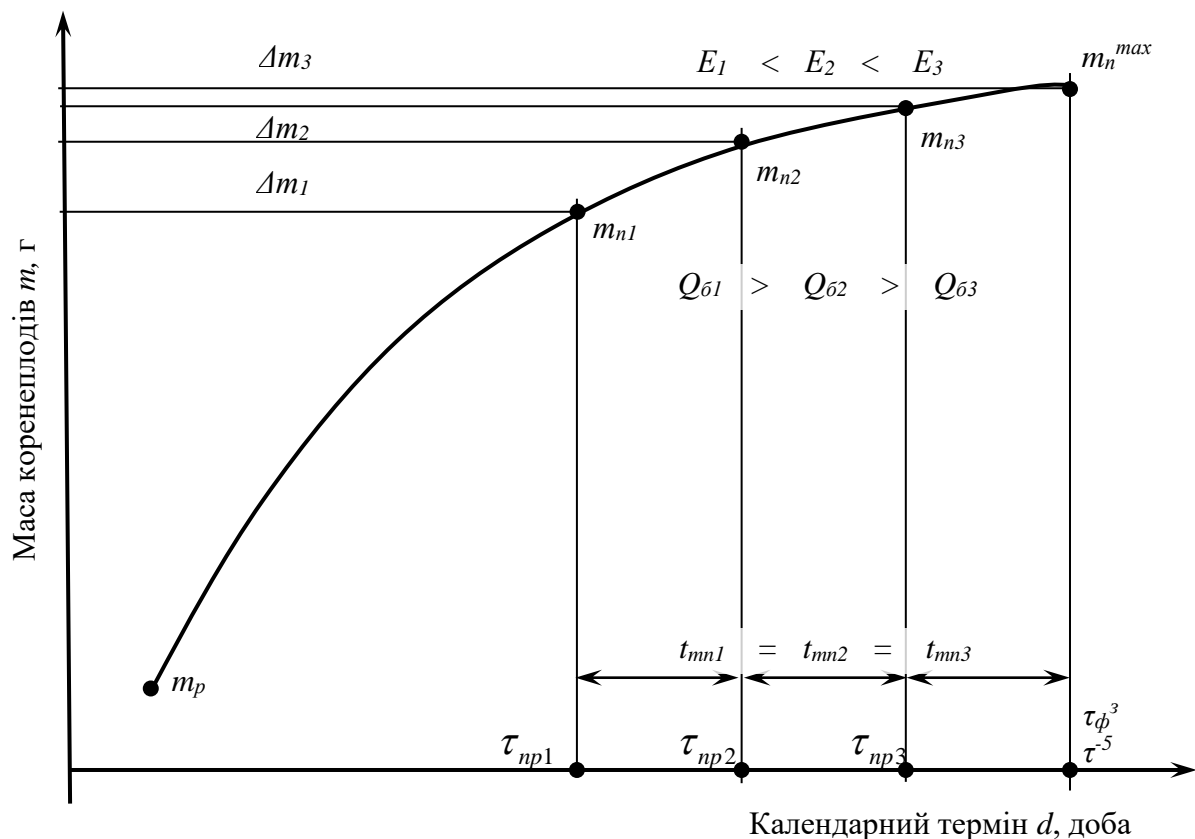


Рис. 2.2. Предметні передумови формування термінів робіт у проектах

Зокрема відомо, що закономірність Δm класифікувати на три види інтенсивності зміни в часі. Однак характерна крива приросту Δm має чітко виражену закономірність перехідного процесу (рис. 2.3).

Аналіз закономірностей зміни Δm та t_{n3} дає підстави стверджувати, що бурякозбиральні роботи у проектах збирання врожаю необхідно узгоджувати з базовими подіями проектного середовища, а також динамікою їх зміни. Як видно з рис. 2.3 τ_{np} необхідно обґрунтовувати відповідно до τ_{ϕ}^3 (або τ^{-5}), за якого $\Delta m = 0$, а також з огляду на забезпечення умови $t_{mn} = t_{n3}$.

Однак необхідно зазначити, що динаміка t_{n3} для різних років реалізації проектів ЗЦБ є також змінною.

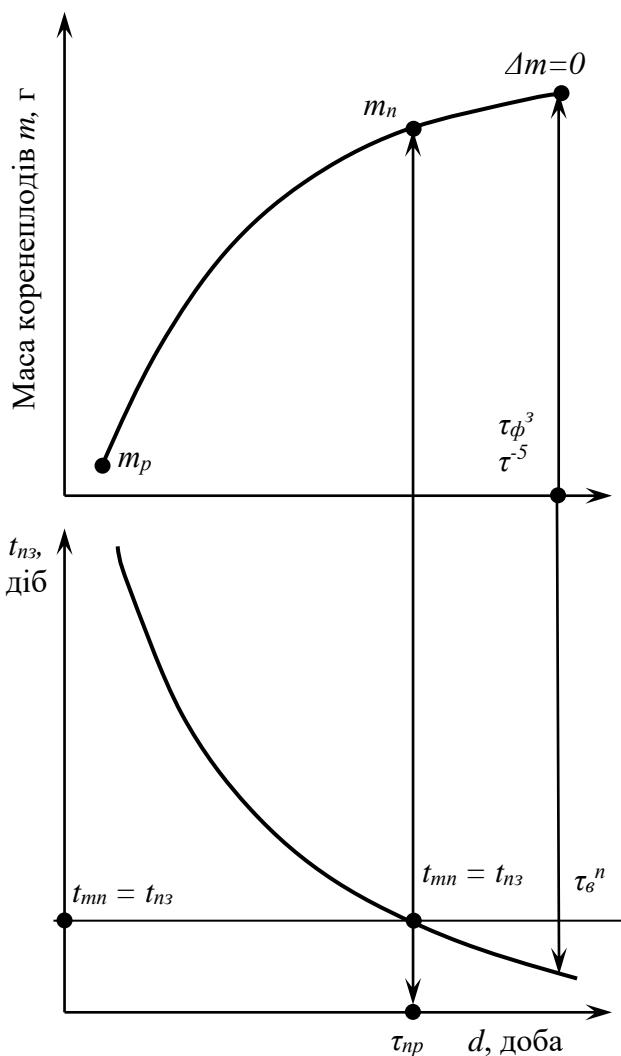


Рис. 2.3. Графічна інтерпретація зміни біологічної та агрометеорологічної складових проектного середовища: t_{mn} – технологічно потрібний фонд часу, днів; τ_{ϕ}^n – час початку технологічних втрат, доба

Відповідно до цього τ_{np} характеризуватиметься стохастичністю. Врахування особливостей об'єктивного розвитку умов проектного середовища для розглянутих варіантів t_{n31} та t_{n32} уможливило формулювання вимог до інформаційно-аналітичної системи планування відповідних робіт. Очевидно, що для τ_{np1} та τ_{np2} справедливою буде умова:

$$\begin{aligned}
 t_{n31} &= t_{n32}; \\
 m_{n1} &< m_{n2}; \\
 Q_{\delta 1} &> Q_{\delta 2}; \\
 E_1 &< E_2.
 \end{aligned}
 \tag{2.2}$$

Аналіз закономірності m_n дав змогу встановити, що обґрунтування τ_{np} для окремих випадків необхідно здійснювати на підставі Δm . Це зумовлено особливостями розвитку агрометеорологічної та біологічної складових проектного середовища ЗЦБ. Виходячи з викладеного, обґрунтування τ_{np} у проектах ЗЦБ із заданими $W_{\text{доб}}$ та S^n необхідно здійснювати за одночасного оцінювання наступних показників: t_{n3} , Δm , τ_{ϕ}^3 та τ^{-5} .

РОЗДІЛ 3

ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ТА ФОРМУВАННЯ ПОЧАТКОВИХ ДАНИХ

3.1. Опис інструментів та мови програмування

Microsoft Visual Studio — Серія продуктів фірми Майкрософт, які включають інтегроване середовище розробки програмного забезпечення та низку інших інструментальних засобів. Ці продукти дозволяють розробляти як консольні програми, так і програми з графічним інтерфейсом, в тому числі з підтримкою технології Windows Forms, а також веб-сайти, веб-застосунки, веб-служби як в рідному, так і в керованому кодах для всіх платформ, що підтримуються Microsoft Windows, Windows Mobile, Windows Phone, Windows CE, .NET Framework, .NET Compact Framework та Microsoft Silverlight.

Visual Studio включає один або декілька з наступних компонентів:

- Visual Basic .NET, а до його появи — Visual Basic
- Visual C++
- Visual C#
- Visual F# (входить до складу Visual Studio 2010);
- Visual Studio Debugger

Багато варіантів постачання також включають:

- Microsoft SQL Server або
- MSDE Visual Source Safe — файл-серверна система управління

версіями

У минулому, до складу Visual Studio також входили продукти:

- Visual InterDev
- Visual J++
- Visual J#
- Visual FoxPro
- Visual Source Safe – файл-серверна система управління версіями.

Для побудови програми у Visual Studio широко використовують засоби, які надає система. Є дві частини побудови: перша – проектування інтерфейсу з використанням стандартних елементів (компонент) та маніпулювання їхніми розмірами й розташуванням; друга – написання фрагментів програмного коду для виконання завдання. Visual Studio самостійно записує деякі частини програми без зовнішнього втручання, розробникові треба кодувати лише суто свою задачу. Крім того, Visual Studio формує для майбутньої програми потрібну інформацію в файлах. На всіх етапах розробки програми можна бачити її інтерфейс, перевірити програму шляхом виконання, змінювати властивості компонент, на яких побудована програма, вилучати чи додавати візуальні компоненти.

На початку роботи з системою Visual Studio доцільно виконати її мінімальне налаштування.

По-перше, треба підготувати папку (каталог), де будуть зберігатися всі файли майбутньої програми. Це ліпше робити до запуску Visual Studio.

Нову папку створити за звичайними правилами операційної системи Windows. Наприклад, запустити інструмент Мій комп'ютер (MyComputer), розташований на робочому столі системи, відшукати потрібний диск та вже наявну папку, відкрити її, після чого через меню вибрати команди Файл→Створити→Папка. Ввівши з клавіатури потрібне ім'я папки та натиснути на клавішу Enter, матимемо готову порожню папку.

Свою папку можна створити і пізніше, під час побудови програми. У разі першого запам'ятовування відшукати у стандартному вікні запам'ятовування файлів потрібні диск та папку і за допомогою кнопки вікна Створення нової папки так само, як і в попередньому випадку, ввести з клавіатури ім'я папки.

Подруге, потрібно вибрати і налаштувати деякі важливі параметри Visual Studio, які постійно впливають на подальшу роботу. Вище зазначено, що в меню треба послідовно вибрати пункти Tools→EnvironmentOptions...→ закладку Preferences і за допомогою мишки поставити позначку в поля Showcompiler та Minimizeonrun. Таку ж позначку потрібно поставити в полі Breakonexception.

3.2. Відображення технологічних операцій моделлю

Початковою базою знань для системно-подієвого відображення умов проектного середовища у статистичній імітаційній моделі процесів ЗЦБ є множина залежностей характеристик та моделей ризику агрометеорологічної і предметної складових [3].

Відображення бурякозбиральних робіт у віртуальному проекті ЗЦБ відбувається у чисельному форматі. З метою встановлення закономірностей зміни інтегрованих функціональних показників проектів за різних планових термінів початку (τ_{np}) робіт, моделювання виконується для календарного періоду з 1 вересня (243-й день з початку року) до 1 грудня (304-й день) (рис. 3.1).

Для відображення термінів робіт у моделі технологічних процесів ЗЦБ, що входять до складу інформаційно-аналітичної системи, у чисельному форматі формується календарна вісь, на якій відкладається τ_{np} , час (τ^{-5}) початку заморозку (нижче -5°C) та час (τ_{ϕ^3}) завершення фізичної стиглості ґрунту (див. рис. 3.1, а). Відображення погожих та непогожих проміжків часу для цього періоду здійснюється позначеннями 1 та 0 відповідно.

Генерування псевдовипадкових показників, що відображають стохастичність агрометеорологічної складової проектного середовища ЗЦБ, здійснюється за наступними розподілами: 1) тривалості погожих (t_{nn}) та непогожих (t_{nn}) проміжків; 2) моментів τ^{-5} і τ_{ϕ^3} . Відкладаючи ці події на календарну вісь (див. рис. 3.1, б) та проектуючи на ній τ_{np} , отримують можливість відобразити природно зумовлений фонд часу на виконання бурякозбиральних робіт у проектах ЗЦБ. Користуючись статистичними закономірностями, здійснюють її відображення у моделі віртуального проекту ЗЦБ (рис. 3.1, в).

Перебіг бурякозбиральних робіт у проектах ЗЦБ в j -у добу відображають на підставі віднімання добової продуктивності ($W_{доб}$) відповідного бурякозбирального комбайна від площі поля (S_{j-1}), яка залишилась необробленою у попередній ($j-1$) день:

$$S_j = S_{j-1} - W_{\text{доб}}, \quad (3.12)$$

де S_j - незібрана площа цукрових буряків на кінець поточної j -ї доби, га.

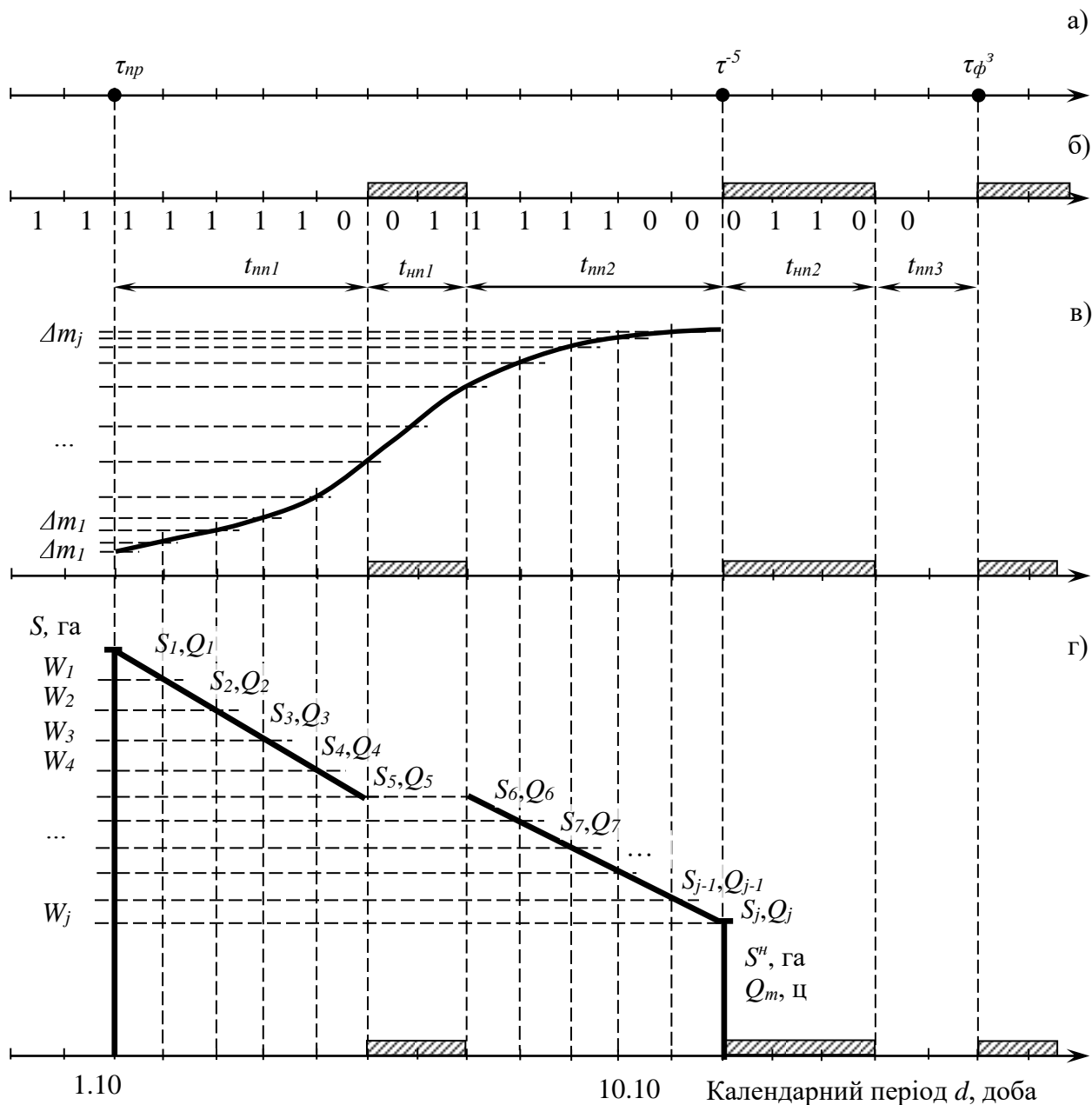


Рис. 3.1. Графічна інтерпретація відображення бурякозбиральних робіт імітаційною моделлю: а) календарні терміни початку та завершення робіт; б) погожі та непогожі проміжки; в) календарний приріст маси коренеплодів цукрових буряків; г) перебіг робіт у проектах; S_j – площа культури, що зібрана в j -у добу, га; $Q_{\phi j}$ – обсяг врожаю цукрових буряків, що зібраний в j -у добу, ц; S^h – площа, на якій врожай культур уражено заморозками, га; Q_m – обсяг технологічних втрат врожаю цукрових буряків, ц

Очевидно, що S_j є початковою площею для виконання робіт у наступну добу. Для непогожих проміжків календарного періоду робіт $W_{доб} = 0$ га/добу, тому $S_j = S_{j-1}$ і т.д. Запізнення з виконанням робіт до моменту τ^5 , або τ_{ϕ}^3 зумовлює технологічні втрати у проектах ЗЦБ.

Встановлення таких інтегрованих функціональних показників проектів ЗЦБ, як валовий обсяг (Q_{ϕ}) зібраного врожаю цукрових буряків та обсяг (Q_m) технологічних втрат, за планового τ_{np} є підставою оцінення узгодженості обсягу, темпу та часу початку робіт з умовами проектного середовища, а відтак і обґрунтування реакцій на ВТР у цих проектах.

Визначення Q_{ϕ} здійснюється за допомогою щоденного додавання обсягів зібраного врожаю культури: $\left(m_n + \sum_1^j \Delta m_1\right) \cdot S_1 \cdot 0,9$ – перший день робіт у проектах ЗЦБ, $\left(m_n + \sum_1^j \Delta m_2\right) \cdot S_2 \cdot 0,9$ – другий день і т.д. Сума цих добутоків відображає в Q_{ϕ} у центнерах:

$$Q_{\phi} = \sum_1^j \left(\left(m_n + \sum_1^j \Delta m_j \right) \cdot S_j \cdot 0,9 \right), \quad (3.1)$$

де m_n – поточна маса коренеплодів цукрових буряків станом на τ_{np} , г; 0,9 – коефіцієнт переведення ваги зібраного врожаю цукрових буряків з грамів у центнери (з розрахунку 90 тис. коренеплодів на 1 га)].

У результаті моделювання бурякозбиральних робіт у віртуальному проекті ЗЦБ отримують інтегровані функціональні показники, до складу яких відносимо: 1) обсяг (Q_{ϕ}) фактично зібраного врожаю коренеплодів цукрових буряків; 2) обсяг (Q_{δ}) біологічних втрат; 3) обсяг (Q_m) технологічних втрат; 4) обсяг фактично (S_{ϕ}) зібраних площ культури; 5) ймовірність (P_{Qm}) виникнення технологічних втрат; 6) ймовірність ($P_{Q\delta}$) виникнення біологічних втрат; 7) густину розподілу (моделі ризику) Q_{ϕ} , Q_{δ} та Q_m .

3.3. Розробка блок-схеми алгоритму прикладної програми

За допомогою статистичної імітаційної моделі процесу механізованого збирання цукрових буряків та математичного опрацювання її результатів встановлюються фізичні характеристики одиничного комбайна при виконанні збирального процесу для відповідних характеристик виробничої програми СГП.

Умовою одержання об'єктивних результатів моделювання є використання сталих статистичних характеристик та параметрів розподілу: 1) площі ріллі СГП (S_p); 2) часу завершення (τ_{ϕ^3}) фізичної стиглості ґрунту у осінній період; 3) тривалості погожих і непогожих проміжків часу 4) часу зниження середньодобової температури нижче -2°C (τ^z); 5) часом завершення фізичної стиглості ґрунту (τ_{ϕ^3}).

Імітаційне моделювання виконувалось поетапно: 1) розроблялась блок-схема та алгоритм імітаційної моделі, а також програмне забезпечення для реалізації його на ПК; 2) виконувалось попереднє моделювання збирального процесу з метою валідації моделі та перевірки її адекватності реальному процесу; 3) обґрунтовувалась потрібна кількість реалізацій моделі процесу механізованого збирання цукрових буряків; 4) виконувались комп'ютерні експерименти (статистичне імітаційне моделювання); 5) математично опрацьовувались результати моделювання та обґрунтовувались залежності показників системної ефективності одиничного комбайна від характеристик виробничої програми СГП.

Імітаційне моделювання бурякозбирального процесу та фіксування характеристик роботи одиничного комбайна впродовж осіннього періоду виконували для наступних років (N_p - число реалізацій моделі) [7] для варіантів тривалості роботи 7, 10.5, 14год. а також для різного терміну початку бурякозбиральних робіт(при прирості маси кореня менше 3гр, 2.5гр, 2гр). На підставі математичного опрацювання результатів моделювання визначалися статистичні характеристики фізичних показників бурякозбирального процесу та обґрунтовувалась потрібна кількість реалізацій моделі на ПК за умови допустимої

відносної похибки 0,05 та довірчої ймовірності (0,95), яка є достатньою для інженерних розрахунків [7].

Блок-схема та алгоритм імітаційного моделювання розроблялись на підставі аналізу причинно-наслідкових зв'язків процесу механізованого збирання цукрових буряків одиничним комбайном для осіннього сезону. Блок-схема складається із 5 основних блоків (рис. 3.2).

Перший, та другий блоки призначені для занесення в пам'ять ПК початкових даних: параметрів розподілів кількісних показників імовірнісних чинників; характеристик виробничої програми СГП тощо.

Блок 3 це сам блок відображення функціонування виробничої системи в якому проводиться безпосереднє моделювання

Блоки 4,5 призначені виведення даних моделювання в графічному вигляді на екран монітора.

Розглянемо детальніше блок 3 «Блок імітаційного моделювання». Блоки 3.1 і 3.2 призначені для отримання початкових параметрів моделювання і запуску моделювання для заданої площі збирання.

В блоках 3.3, 3.4 обнулюються значення бігучих змінних а також, генерують агрометеорологічні умови поточного сезону(початкова маси кореня, дата завершення фізичної стиглості ґрунту, тривалість погожих та непогожих проміжків). Встановлена потрібна кількість реалізацій (ітерацій) (N_p) моделі уможливорює об'єктивне відображення імовірнісних характеристик некерованих чинників процесу.

Блок 3.5 формує погоду на сезон з врахуванням згенерованих в попередньому блоці масивів даних.

Блок 3.6 аналізує масиви вихідних даних моделювання та виконує імітацію процесу збирання цукрових буряків для. В результаті моделювання у змінних масивах фіксують обсяги несвоєчасно оброблених площ, обсяги фактично оброблених площ.

Блок 3.7 призначений для встановлення статистичних характеристик фізичних показників процесу збирання цукрових буряків за результатами N_p

ітерацій моделі для одиничного комбайна та відповідних характеристик виробничої програми СГП.

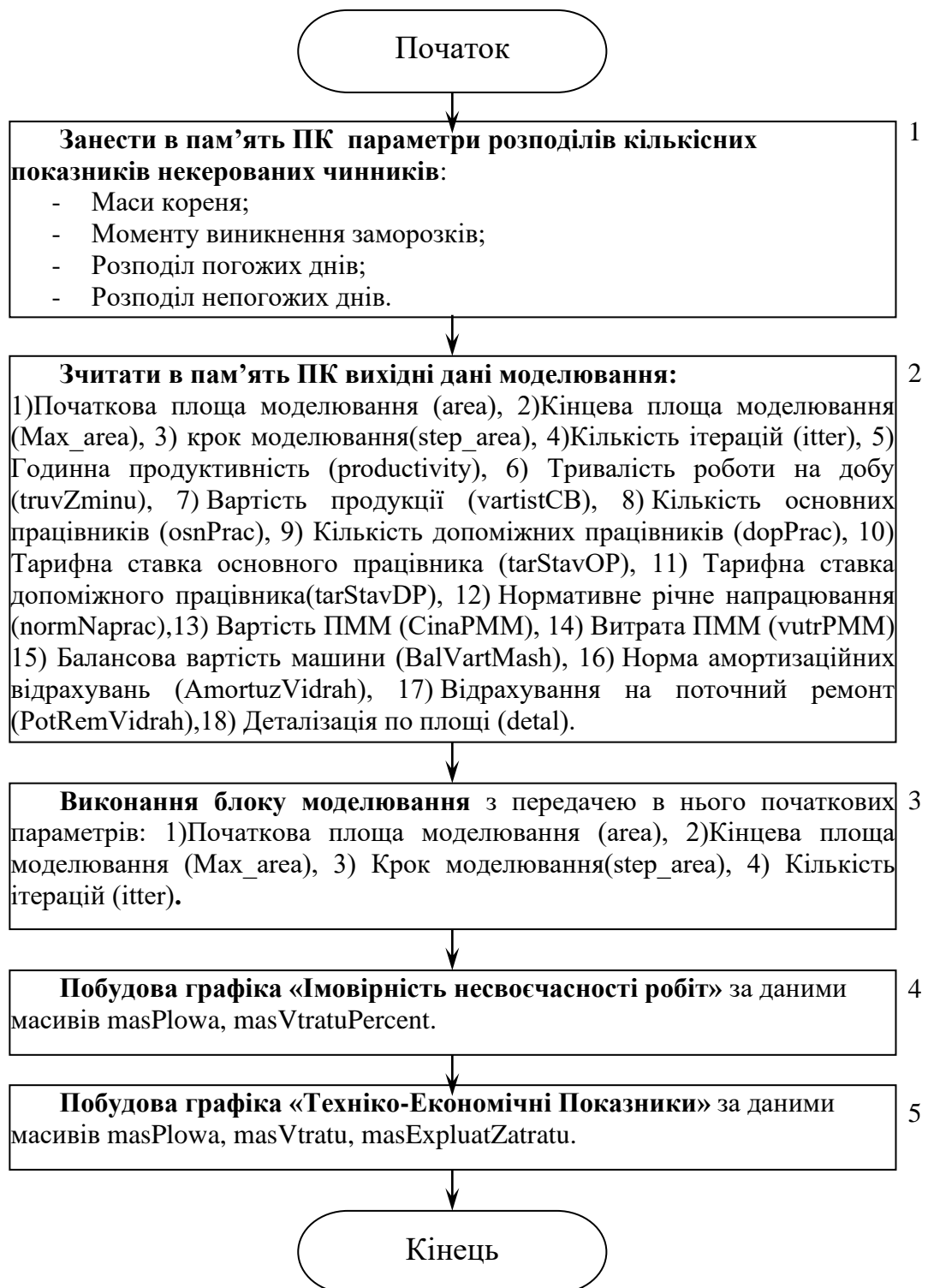


Рис. 3.2. Блок-схема алгоритму імітаційної моделі процесу механізованого збирання цукрових буряків (загальний вигляд)

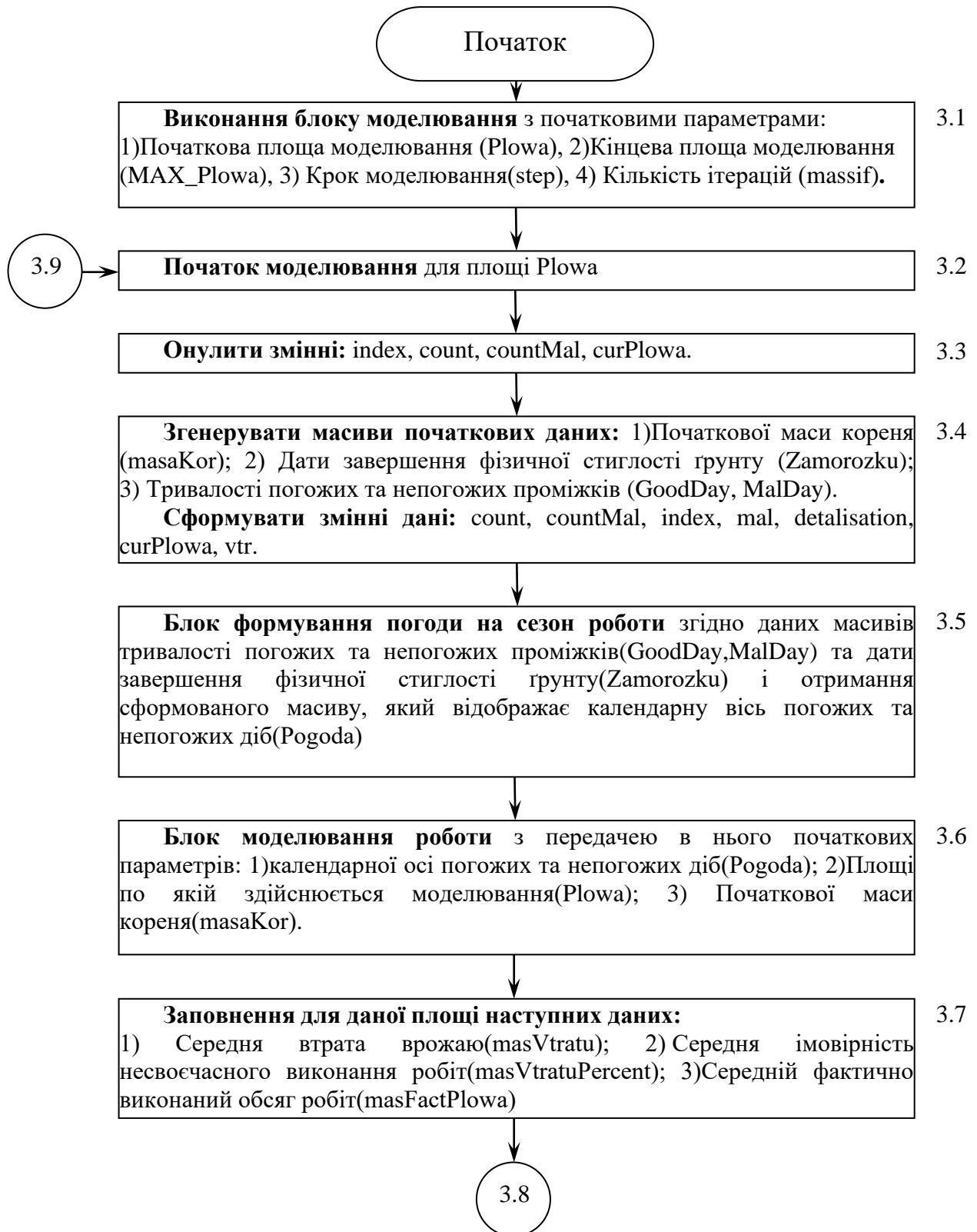


Рис. 3.2. Блок-схема алгоритму відображення функціонування виробничої системи

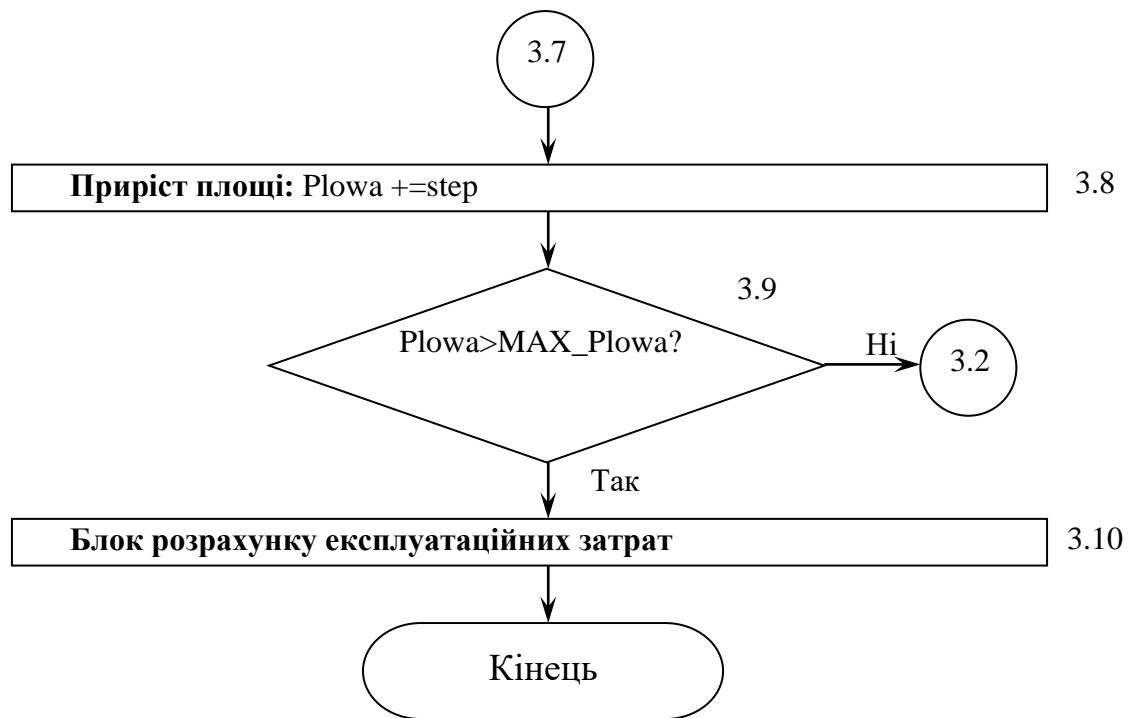


Рис. 3.2. Блок-схема алгоритму відображення функціонування виробничої системи (продовження).

Блоки 3.8, 3.9 інкрементують програму виробництва СГП та перевіряють її належність в межах моделювання.

Блок 3.10 виконує розрахунок експлуатаційних затрат.

Деталізуємо блок 3.6 «Блок імітації роботи комбайна на полі» рис. 3.2.

Блоки 3.6.1-3.6.3 призначенні для отримання вхідних параметрів, обнулити біжучі змінні, вирахувати добову продуктивність.

В блоках 3.6.4, 3.6.5 проводимо моделювання для i -ої доби та перевіряємо вчасність виконання обробітку.

В блоках 3.6.6-3.6.9 відбувається моделювання приросту маси кореня, перевірка на можливість виконання робіт та саме моделювання роботи.

Блок 3.6.10 визначає обсяг незібраного врожаю.

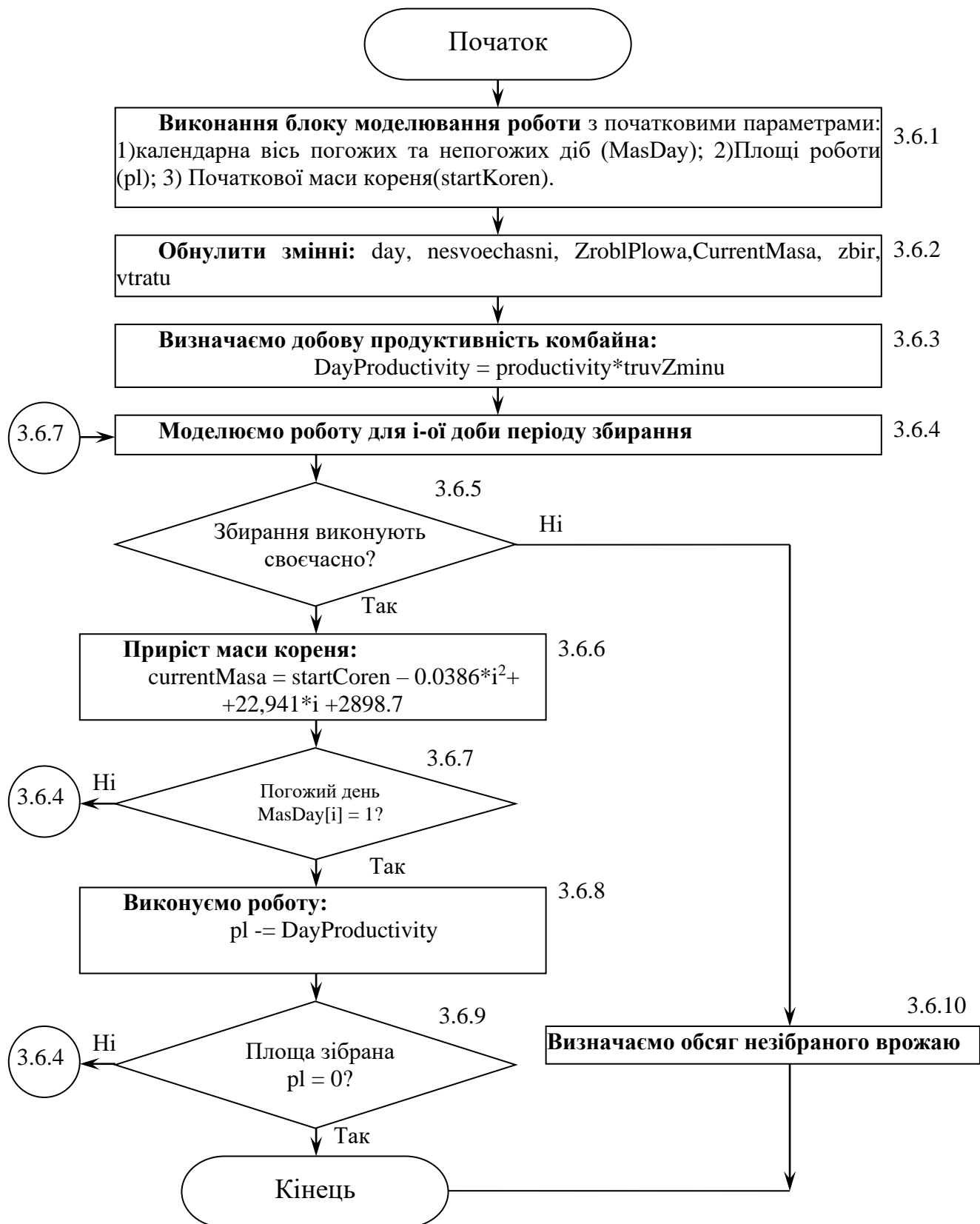


Рис. 3.2. Блок-схема алгоритму імітаційної моделі процесу механізованого збирання цукрових буряків (Блок імітації роботи комбайна на полі)

РОЗДІЛ 4.

ПРАКТИЧНЕ ВИКОРИСТАННЯ ПРОГРАМНОГО ДОДАТКУ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ ТЕХНОЛОГІЧНИХ ОПЕРАЦІЙ

4.1. Імітаційне моделювання технологічних операцій виробництва

На основі розробленої методики була реалізована комп'ютерна програма статистичної імітаційної моделі фрагмент коду, якої наведено в додатках (дод. А). Дана програма (рис. 4.1) дозволяє змоделювати агрометеорологічні умови періоду збирання цукрових буряків та відповідно до цих умов роботу бурякозбирального комбайна. Для процесу моделювання як базовий взято комбайн КС-6Б-10

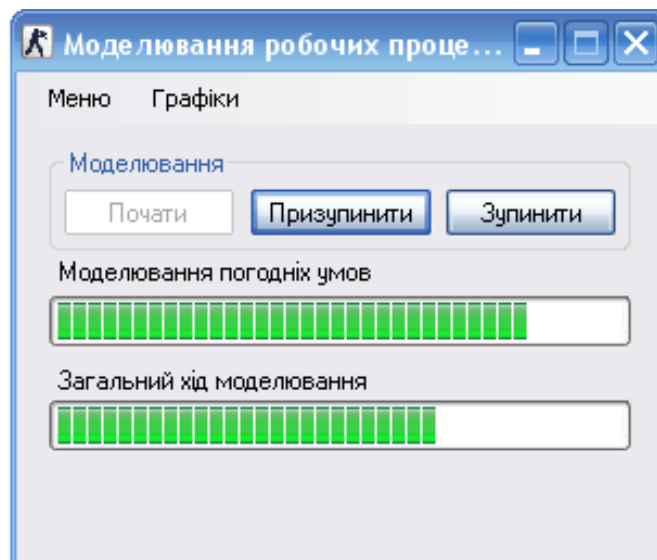


Рис. 4.1. Головне вікно прикладної програми імітаційної моделі

В результаті моделювання вдалося отримати дані на основі яких можна вивести основні характеристики роботи бурякозбирального комбайна: залежність ймовірності своєчасної роботи від сезонної площі комбайна, залежність фактично обробленої площі комбайном від його сезонного навантаження, розподіл незібраної площі.

Так під час роботи імітаційної моделі збирання цукрових буряків встановили залежність ймовірності своєчасного виконання робіт від сезонної площі для різних варіантів тривалості роботи та терміну початку збирання.

Імовірність своєчасного збирання врожаю для випадку коли початок збирання припадає на час зменшення добового приросту маси кореня Δm менше 2 гр показано на рис. 4.2.

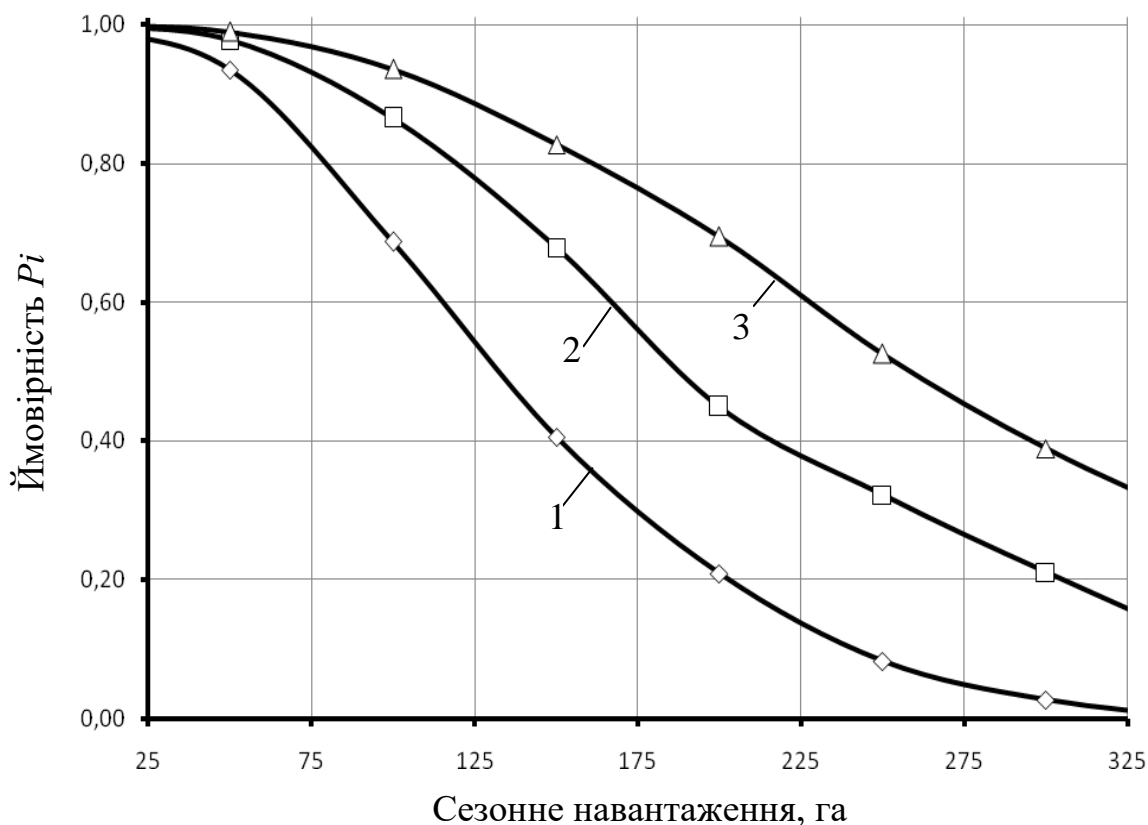


Рис. 4.2. Залежність своєчасної роботи комбайна від його сезонної площі: 1 – 7 год; 2 – 10,5 год; 3 – 14 год.

Також під час моделювання роботи бурякозбирального комбайна отримали залежність фактично обробленої площі від сезонної програми (рис. 4.10).

Як видно з даної залежності із збільшенням сезонного навантаження комбайна фактично-зроблена площа зростає до певної межі і тримається на цьому рівні це свідчить про вплив на тривалість збирання не тільки фізичних характеристик комбайна а й агрометеорологічних умов.

За даними імітаційної моделі отримали дані для побудови розподілу незібраної площі для таких початкових умов моделювання: початок збирання – приріст маси менше 2 гр, тривалість роботи 14 год, сезонне навантаження 120 га.

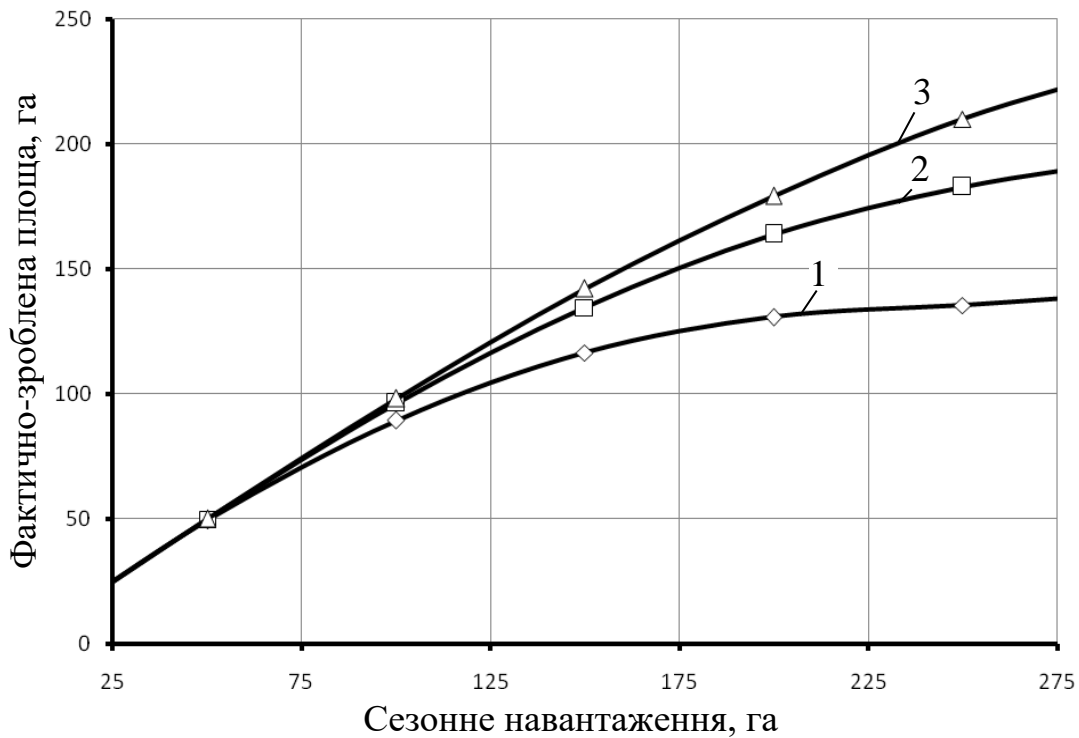


Рис. 4.3. Залежність фактично-зробленої площі від сезонного навантаження бурякозбирального комбайна: 1 – 7 год; 2 – 10,5 год; 3 – 14 год.

4.2. Результати підтримки прийняття рішень із оптимізації сезонної площі для бурякозбирального комбайном

Згідно з наведеною в р.3. методикою встановимо оптимальну сезонну площі для початкових умов виконання процесу: початок збирання – приріст кореня менше 2 гр, тривалість роботи – 14 год. для діапазону сезонного навантаження площі від 10 до 120 га з кроком 10 га.

Для прикладу розрахуємо питомі сукупні витрати коштів на виконання процесу збирання цукрових буряків для сезонного навантаження 110 га.

Згідно із результатами моделювання, враховуючи погодні умови, отримуємо залежно від сезонного навантаження обсяг фактично зроблених площ (табл. 4.1).

Таблиця 4.1. Результати визначення фактично обробленої площі бурякозбиральним комбайном КС-6Б-10 «Тернопіль»

| | | | | | | | | | | | | |
|-----------------------------|----|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|
| Сезонне навантаження, га | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 | 110 | 120 |
| Фактично-зроблена площа, га | 10 | 19,99 | 29,98 | 39,95 | 49,89 | 59,78 | 69,61 | 79,32 | 88,95 | 98,29 | 107,48 | 116,65 |

Відповідно до обсягу фактично виконаних робіт агрегатом питомі витрати коштів на оплату праці [7]:

$$C_1 = \frac{(160 \cdot 1 + 98.5 \cdot 1)}{1.1} = 235.0 \text{ грн/га.}$$

Вартість паливо-мастильних матеріалів визначається за формулою

$$C_2 = 25.5 \cdot 52 = 1326.0 \text{ грн/га.}$$

Відрахування на амортизацію:

$$C_3 = \frac{815000 \cdot 1 \cdot 15}{100 \cdot 107.48} = 1137.4 \text{ грн/га.}$$

Значення коефіцієнта k_r відображає відношення фактично виконаного обсягу робіт, до планового річного навантаження:

$$k = \frac{107,48}{1.1 \cdot 180} = 0.54.$$

Відрахування на поточний ремонт та технічне обслуговування:

$$C_4 = \frac{815000 \cdot 1 \cdot 12 \cdot 0.54}{100 \cdot 1.1 \cdot 180} = 266.7 \text{ грн/га}$$

Питомі експлуатаційні витрати ($B_{\text{тн}}$) на виконання бурякозбирального процесу

$$B_{\text{тн}} = 235.0 + 1326.0 + 1137.4 + 266.7 = 2965.1 \text{ грн/га}$$

Вартісне оцінення втрат врожаю ($B_{\text{тл}}$):

$$B_{\text{тл}} = \frac{572.75 \cdot 2.52 \cdot 155}{107.5} = 2081.35 \text{ грн./га}$$

Питомі сукупні витрати коштів на виконання процесу включають – питомі експлуатаційні витрати ($B_{\text{тн}}$) процесу та питомі втрати ($B_{\text{тл}}$) врожаю культури через несвоєчасне її збирання:

$$E = 2965.1 + 2081.35 = 5046.5 \text{ грн/га.}$$

Провівши розрахунки аналогічним чином для інших варіантів сезонного навантаження отримаємо дані які зведені в табл. 4.2.

Таблиця 4.2. Результати оцінення питомих експлуатаційних витрат, питомих втрат та питомих сукупних витрат на виконання процесу

| Сезонне навантаження, га | Питомі експлуатаційні витрати, грн/га. | Питомі втрати, грн/га | Питомі сукупні витрати коштів, грн/га |
|--------------------------|--|-----------------------|---------------------------------------|
| 10 | 18761,7 | 0,0 | 18761,7 |
| 20 | 9896,1 | 19,0 | 9915,2 |
| 30 | 6965,8 | 47,2 | 7013,0 |
| 40 | 5520,3 | 110,8 | 5631,1 |
| 50 | 4667,9 | 177,0 | 4844,9 |
| 60 | 4113,8 | 318,2 | 4432,0 |
| 70 | 3729,0 | 502,5 | 4231,5 |
| 80 | 3451,0 | 740,1 | 4191,1 |
| 90 | 3242,9 | 1039,0 | 4281,9 |
| 100 | 3086,7 | 1535,8 | 4622,6 |
| 110 | 2965,1 | 2081,4 | 5046,5 |
| 120 | 2868,2 | 2581,5 | 5449,7 |

Отже, оптимальною сезонною програмою для заданих умов буде навантаження в розмірі $S_{opt} = 80$ га при якому досягаємо найменших питомих сумарних затрат коштів $E_{min} = 4191,1$ грн/га.

РОЗДІЛ 5.

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

5.1. Структурно-функціональний аналіз технологічного процесу

Розробка та вживання ефективних заходів запобігання аварійним і травмонебезпечним ситуаціям можливі лише при завчасному виявленні тих небезпек, з яких починаються процеси їх формування. Оскільки небезпечні умови не завжди завчасно можна виявити, а для вивчення небезпечних дій іноді потрібно багато часу, щоб зібрати статичний матеріал, то і методи виявлення цих небезпек повинні бути відповідно диференційовані (табл. 5.1) [5].

Таблиця 5.1. Моделі формування та виникнення травмонебезпечних і аварійних ситуацій

| Вид робіт, виробничий підрозділ, робоче місце, виробниче обладнання, склад агрегату | Виробнича безпека | | | Можливі наслідки | Заходи запобігання небезпечним ситуаціям |
|--|---|-------------------------|--------------------------|------------------|---|
| | Небезпечна умова (НУ) | Небезпечна дія (НД) | Небезпечна ситуація (НС) | | |
| Виконання робіт із електрообладнанням | Не вимкнено живлення. Відсутність заземлення. | Нехтування правилами ТБ | Ураження струмом | Травма (Т) | Проведення повторного інструктажу з ТБ. Розробка нових способів захисту. Встановлення заземлення. |
| <pre> graph TD ND[НД] --> NS[НС] NU[НУ] --> NS NS --> T[Т] </pre> | | | | | |

Відповідно до аналізу небезпечних умов, які існують у виробничому процесі виокремлено такі наступні за характером дії на працівника їх групи [5]:

– характеризують стан або рівень безпеки обладнання, які використовуються.

- сприяють виникненню технологічних помилок обслуговуючого персоналу впродовж виробничого процесу;
- створювати умови та можливість проникнення працівника в небезпечну зону;
- приводять до виникнення небезпечних дій (внаслідок низького рівня професійної підготовки працівників та організації навчання з охорони праці).

Моделі формування та виникнення травмонебезпечних і аварійних ситуацій в комп'ютерному кабінеті представлено у вигляді моделі формування та виникнення травмонебезпечних і аварійних ситуацій – табл. 5.1.

5.2. Розрахунок освітлення приміщення комп'ютерного кабінету

Освітленість виробничих приміщень може бути штучною і природною. Природне освітлення при правильному обладнанні найбільш сприятливе для людини. Основні вимоги для освітлення наступні:

- освітлення повинне бути достатнім для швидкого і легкого розпізнання об'єктів роботи;
- освітлення повинно бути рівномірне без різких тіней;
- джерело світла не повинно осліплювати працівника;
- рівень освітленості не повинен обмежуватись часом.

Природне освітлення забезпечується обладнанням вікон (бокове освітлення) фонарів і світильних покриттів приміщень (верхнє освітлення). Природне освітлення нормується коефіцієнтом природної освітленості. Коефіцієнт природної освітленості – це процентне відношення фактичної освітленості F_v в будь-якій точці приміщення до освітленості F_n розсіяної світлом небозводу точки, яка лежить на відкритій місцевості. Розрахунок природного освітлення через бокові вікна по нормам освітленості ведеться для самої дальньої від вікон точки, тобто знаходять мінімальне значення стик коефіцієнта природної освітленості [5]:

$$e_{\min} = \frac{F_b}{F_H} \cdot 100. \quad (5.1)$$

Значення коефіцієнта природної освітленості визначається не менше чим в п'яти точках. Значення коефіцієнта природної освітленості для сільськогосподарських виробничих приміщень в даному випадку ремонтній майстерні, беремо $e_{\min} = 5\%$.

Розрахунок природного освітлення зводиться до визначення площі світлових променів.

Сумарну площу світлових променів $\sum F_o (m^2)$ по коефіцієнту природної освітленості для бокових променів визначаємо по формулі:

$$\sum F_o = \frac{F_H \cdot e_{\min} \cdot r_o \cdot K}{100 \cdot \tau \cdot \Gamma_1}, \quad (5.2)$$

де F_H – площа підлоги, m^2 ; e_{\min} – величина мінімального коефіцієнта природного освітленості; τ – загальний коефіцієнт світловикористання віконного отвору із врахуванням його забруднення, $\tau = 0,25$; r_o – світлова характеристика вікна, $r_o = 9,5$; Γ_1 – коефіцієнт, який враховує підвищення освітленості за рахунок світла, яке відбивається від стін і стелі, $\Gamma_1 = 1,2$; K – коефіцієнт, який враховує затінення вікон сусідніми приміщеннями і загорожею, $K = 1$.

$$\sum F_o = \frac{36 \cdot 0,5 \cdot 9,5 \cdot 1}{100 \cdot 0,25 \cdot 1,2} = 5,7 m^2.$$

Кількість світлових променів визначимо:

$$N = \frac{\sum F_o}{F_o}, \quad (5.3)$$

де F_o – площа вікна згідно стандарту, m^2 .

$$N = \frac{5,7}{6} = 0,95.$$

Приймаємо кількість вікон – одне вікно.

При розрахунку природного освітлення найбільш поширеним і простим є метод світлового потоку. При цьому методі розраховуємо світловий потік F_{λ} (Лк), який повинна випромінювати кожна лампа (при заданій кількості ламп).

$$F_{л} = \frac{k \cdot S_n \cdot E}{n_{л} \cdot \eta \cdot r^2}, \quad (5.4)$$

де k – коефіцієнт запасу, $k = 1,3$; S_n – площа підлоги, m^2 ; $S_n = 36 m^2$. E – нормативна освітленість, $E = 300$ Лк; $n_{л}$ – кількість встановлених ламп, $n_{л} = 6$ од; η – коефіцієнт використання світлового потоку, $\eta = 0,25$; r – коефіцієнт нерівномірності освітленості, $r = 0,545$.

Коефіцієнт запасу (K) враховує можливість забруднення світильників пилом, що залежить від характеру виробництва.

Розрахунок штучного освітлення починаємо із визначення висоти розташування світильника і їх кількості. Висоту h_H (м) розташування світильників над робочим місцем знаходимо за формулою:

$$h_H = H - (h_1 + h_2), \quad (5.5)$$

де H – висота приміщення, м; h_1 – віддаль від підлоги до освітлювальної поверхні, м; h_2 – віддаль від стелі до світильника, м.

$$h_H = 4,5 - (2,2 + 1,5) = 0,8 \text{ м.}$$

При симетричному розміщенні світильників по вершинах квадратів їх кількість визначається за формулою:

$$n_c = \frac{S_n}{l^2}, \quad (5.6)$$

де l – віддаль між світильниками, м.

Підставивши значення отримаємо:

$$n_c = \frac{36}{9} = 4 \text{ од.}$$

Тоді світловий потік буде становити

$$F_{л} = \frac{1,3 \cdot 36 \cdot 300}{4 \cdot 0,25 \cdot 0,545} = 2576,2 \text{ Лк.}$$

При світловому потоці 2576,2 Лк для заданої лампи вибираємо тип і потужність.

Вибираємо тип лампи – люмінесцентну, потужністю 40Вт.

5.3. Безпека в надзвичайних ситуаціях

Забезпечення захисту населення і території у разі загрози і виникнення надзвичайних ситуацій є одним з найважливіших завдань держави.

Захист населення є системою загальнодержавних заходів, які реалізуються центральними і місцевими органами виконавчої влади, виконавчими органами влад, органами управління з питань надзвичайних ситуацій та цивільного захисту населення, підпорядкованими їм системами, та підприємств, що забезпечують виконання організаційних, інженерно – технічних, санітарно – гігієнічних, проти епідемічних та інших заходів у сфері запобігання та ліквідації наслідків надзвичайних ситуацій.

Загрози життєво важливих інтересів громадян, держави, суспільства поділяють на зовнішні та внутрішні, виконують під час надзвичайних ситуацій техногенного та природного характеру та воєнних конфліктах.

Принципи захисту впливають з основних положень Женевської конвенції щодо захисту жертв війни та додаткових протоколів до неї, можливого характеру воєнних дій, реальних можливостей держави щодо створення матеріальної бази захисту. З метою захисту населення, зменшення втрат та шкоди економіці в разі виникнення надзвичайних ситуацій має право проводитись спеціальний комплекс заходів.

Оповіщення та інформування, яке досягається завчасним створенням і підтримкою в постійній готовності загальнодержавної, територіальних та об'єктивних систем оповіщення населення.

ВИСНОВКИ І РЕКОМЕНДАЦІЇ

1. Сьогодні значна частина агрохолдингів прийшла до такого етапу свого розвитку, що конкуренція на ринку АПК змушує їх використовувати не тільки продуктивні технології виробництва продукції із застосуванням дедалі нових сортів, добрив та хімічного захисту, але й використання сучасних ІТ для ефективного використання наявних ресурсів та управління виробництвом загалом. Переважно, такі системи є платними і доволі важко досяжні для менших господарств. Вони також є дещо узагальненими та не розглядають особливості виробничих процесів на рівні окремих технологічних операцій.

2. Встановлені на підставі моделювання в інформаційно-аналітичній системі фізичні показники процесу збирання цукрових буряків (обсяги втрат через несвоєчасно зібрані площі), дали змогу на підставі вартісного критерію обґрунтувати оптимальну сезонну площу для комбайна. Встановлені залежності є важливою підставою для оцінення показників ефективності бурякозбирального процесу і, на цій підставі, прийняття рішень щодо узгодження характеристик виробничої програми СГП із параметрами парку комбайнів.

3. Імітаційне моделювання і вартісне оцінення втрат біологічної врожайності сільськогосподарських культур є головною підставою виконання оптимізаційних розрахунків, результати яких переконують в тому, із збільшенням тривалості роботи (7, 10,5, 14) одиничного комбайна та незмінної структури площ підприємства оптимальна виробнича площа збільшується. Також при однаковій тривалості роботи на добу із збільшення межі добового приросту маси кореня оптимальне сезонне навантаження також збільшується.

4. На підставі результатів моделювання та опрацювання головних функціональних показників встановлено оптимальну сезонну площу комбайна КС-6Б яка для умов – приріст кореня менше 2 гр, тривалість роботи – 14 год. діапазону сезонної площі 10-120 га із кроком 10 становить 80 га із питомими сукупними витратами коштів – 2735,3 грн/га.

БІБЛІОГРАФІЧНИЙ СПИСОК

1. Голуб Б.М. С#. Концепція та синтаксис. Навч. посібник / Б.М. Голуб, Львів: Видавничий центр ЛНУ імені Івана Франка, 2016. 136 с.
2. Карабиньош С., Новицький А. Підготовка бурякозбиральної техніки до польових робіт // Пропозиція. 2006. №10. 164 с.
3. Комплексне ІТ-рішення для агровиробників. URL: <https://www.soft.farm/uk> (дата звернення: 14.04.2024 р.).
4. Комплексна система ІТ-рішень для управління агробізнесом. URL: <https://agrichain.com.ua/> (дата звернення: 14.04.2024 р.).
5. Лехман С.Д. та ін. Запобігання аварійності і травматизму у сільському господарстві / С.Д. Лехман, В.І. Рубльов, Б.І. Рябцев. К.: Урожай, 1993. 272 с.
6. Лихочвор В.В. Рослинництво. Технології вирощування сільськогосподарських культур. Львів: НВФ “Українські технології”, 2012. 800 с.
7. Сидорчук О., Луб П., Татомир А., Бурилко А. Метод визначення втрат врожаю сільськогосподарських культур внаслідок несвоєчасності механізованих процесів рільництва // Матеріали V ювілейної Міжнар. наук.-техн. конф. “Механізація і енергетика сільського господарства “МОТРОЛ 2005”. Одеса, 2005. Том.7. С. 87-91.
8. Спірін О. М. Зміст навчального матеріалу спецкурсу "Хмарні інформаційно-аналітичні технології у науково-дослідному процесі" / О. М. Спірін, О. А. Одуд // Інформаційні технології і засоби навчання. 2016. Т. 52, Вип. 2. С. 108-120.
9. Стеценко І.В. Моделювання систем: навч. посіб. [Електронний ресурс, текст] / І.В. Стеценко ; М-во освіти і науки України, Черкас. держ. технол. ун-т. Черкаси : ЧДТУ, 2020. 399 с.
10. Тимченко А.А. Основи системного проектування та системного аналізу складних об'єктів: Підручник для студентів вищих закладів освіти / За

ред. В.І. Бикова. К.: Либідь, 2010. 270 с.

11. Томашевський В. М. Моделювання систем. К.: Видавнича група BVH, 2015, 352 с.
12. Шилдт Г. С# 4.0: повне керівництво. М.: ТЗОВ "І.Д. Вільямс", 2011. 1056 с.
13. Big Data information technology and data space architecture / N. Shakhovska, O. Veres, Y. Bolubash, L. Bychkovska // Sensors & Transducers. 2015. Vol. 195, Is.12. P. 69-77.
14. Demydenko, M. A. (2016). Decision support systems [Systemy pidtrymky pryiniattia rishen], Natsionalnyi hirnychiy universytet, Dnipro, 104 s. URL: <http://nmu.org.ua> [in Ukrainian]
15. Bratushka, S. M., Novak S. M., Khailuk S. O. (2020). Decision support systems [Systemy pidtrymky pryiniattia rishen], DVNZ «UABS NBU», Sumy, 265 s. [in Ukrainian]
16. Sytnyk, V. F (2022). Decision support systems [Systemy pidtrymky pryiniattia rishen], KNEU, Kyiv, 614 s. [in Ukrainian]
17. OLAP i Business Intelligence. URL: <http://www.olap.ru/home.asp?catId=60&catPage=56> (14.04.2024 p.).

ДОДАТКИ

Додаток А

ФРАГМЕНТ КОДУ ГОЛОВНОГО ВІКНА ІМІТАЦІЙНОЇ МОДЕЛІ - Form1.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO;
using System.Threading;
using System.Globalization;

namespace Dip
{
    public partial class Form1 : Form
    {
        public struct dani
        {
            public int area, Max_area, step_area, itter;
            public dani(int area, int Max_area, int step_area, int itter)
            {
                this.area = area;
                this.Max_area = Max_area;
                this.step_area = step_area;
                this.itter = itter;
            }
        }

        //-----Початкові дані-----
        public int area = 50, Max_area = 150, step_area = 10, itter = 5, detal = 90;
        //public int itter=5;//, massif =10;
        public double productivity = 1.1, truvZminu = 14, vartistCB = 25;
        //42857 koreniv na lga
        public int osnPrac = 1, dopPrac = 1, normNaprac = 180, start = 258;
        public double CinaPMM = 5.5, BalVartMash = 215000, vutrPMM = 27, AmortuzVidrah = 15,
PotRemVidrah = 12, koevVidrah;
        public double tarStavOP = 12, tarStavDP = 7.5, zibrMasa;
        //-----
        public bool sleep = false, modDone = false;
        StreamWriter sw;
        int[] Pogoda = new int[85];
        public float[] masVtratuPercent;
        public int[] masPlowa;
        public float[] masExploatZatratu, masFactPlowa, masVtratu;
        Random rd = new Random();
        Thread WorkThread;

        public Form1()
        {
            InitializeComponent();
            //-----buttons-----
            Pause.Enabled = false;
            Stop.Enabled = false;
            Thread.CurrentThread.CurrentCulture = new CultureInfo("en-US");
        }

        // розподіл Вейбула
        public double veyb(double a, double b, double c, int min, int max)
        {
            double temp;
            do
            {
                temp = 0;
                temp = (c + a * Math.Exp((1 / b) * (Math.Log(-(Math.Log(rd.NextDouble())))))));
            }
        }
    }
}

```

```

    } while (temp < min || temp > max);
    return temp;
}

// Моделювання
//void model(int Plowa, int MAX_Plowa, int step, int massif)
void model(object o)
{
    try
    {
        dani d1 = (dani)o;
        int Plowa = d1.area;
        int MAX_Plowa = d1.Max_area;
        int step = d1.step_area;
        int massif = d1.itter;
        int count, countMal, index = 0;
        bool detalisation;
        double curPlowa, vtr;
        sw = new StreamWriter("pog.txt");
        double masaKor = 0;
        while (Plowa <= MAX_Plowa)
        {
            detalisation = false;
            if (Plowa == detal) detalisation = true;
            count = 0; countMal = 0; curPlowa = 0; zibrMasa = 0;
            double vtrTemp = 0, tmp = 0;
            for (int i = 0; i < massif; i++)
            {
                masaKor = veyb(156.044, 1.622, 74, 74, 426);
                int Zamorozku = 0;
                for (int z = 0; z < massif; z++)
                {
                    prPogUmovu.Value = 0;
                    Zamorozku = (int)veyb(24.755, 1.545, 277, 277, 332);
                    for (int pog = 0; pog < massif; pog++)
                    {
                        int[] GoodDay = new int[25];
                        for (int G = 0; G < GoodDay.Length; G++)
                        {
                            GoodDay[G] = (int)veyb(5.665, 1.148, 1, 1, 21);
                        }
                        int[] MalDay = new int[25];
                        for (int M = 0; M < MalDay.Length; M++)
                        {
                            MalDay[M] = (int)veyb(2.531, 1.080, 1, 1, 14);
                        }
                        Pogoda = FormPogodu(GoodDay, MalDay, Zamorozku);
                        curPlowa += robota(Pogoda, Plowa, masaKor, detalisation, out vtr,
out zibrMasa);

                        tmp += zibrMasa;
                        vtrTemp += vtr;

                        count++;
                        if (vtr > 0) countMal++;
                        prPogUmovu.PerformStep();
                        prFull.PerformStep();
                        while (sleep)
                        {
                            Thread.Sleep(100);
                        }
                    }
                }
            }
            //sw.WriteLine("{0} \t {1}\t {2:f2} ",countMal , count, curPlowa / count);

            masVtratu[index] = (float)((vtrTemp / count / Plowa) * vartistCB);
            masPlowa[index] = Plowa;
            masVtratuPercent[index] = 100f - (float)countMal * 100 / count;
            masFactPlowa[index] = (float)curPlowa / count;
        }
    }
}

```

```

        //sw.WriteLine("{0}",tmp/count/Plowa);//фактично зібраний врожай
        //sw.WriteLine("{0:f2}\t{1}", masVtratuPercent[index],masPlowa[index]);
        //sw.WriteLine("{0:f2}", masVtratuPercent[index]);
        //sw.WriteLine("{0:f2}", masFactPlowa[index]);
        sw.WriteLine("{0}\t{1}", masVtratu[index], masFactPlowa[index]);
        //sw.WriteLine("площа: {0},   Втрати :{1:f2}",masPlowa[index]-
masFactPlowa[index],masVtratu[index]);
        ++index;
        Plowa += step;
    }
    sw.WriteLine("-----");
    TEP();
    modDone = true;
}
finally
{
    sw.Close();
    //----buttons-----
    Start.Enabled = true;
    Pause.Enabled = false;
    Stop.Enabled = false;
}
    MessageBox.Show("Моделювання завершено!", "OK", MessageBoxButtons.OK,
    MessageBoxIcon.None);
}
// формування ряду погожих і непогожих днів
int[] FormPogodu(int[] pogogiDni, int[] nepogogiDni, int zamoroz)
{
    int[] rez = new int[zamoroz - start];
    int position = 0;
    int temp;
    for (int j = 0; j < pogogiDni.Length && position < rez.Length; j++)
    {
        temp = 0;
        while (temp < pogogiDni[j])
        {
            if (position == rez.Length) break;
            rez[position] = 1;
            temp++;
            position++;
        }
        temp = 0;
        while (temp < nepogogiDni[j])
        {
            if (position == rez.Length) break;
            rez[position] = 0;
            temp++;
            position++;
        }
    }
    return rez;
}

double robota(int[] MasDay, double pl, double startKoren, bool det, out double vtratu, out
double zbir)
{
    int day = 0, nesvoechasni = 0;
    vtratu = 0; zbir = 0;
    int truvSezony = MasDay.Length;
    double ZroblPlowa = 0, CurrentMasa = 0;//, zbir = 0;
    double DayProductivity = productivity * truvZminu;

    for (int i = 0; i < truvSezony; i++)
    {
        if (pl > 0)
        {
            CurrentMasa = startKoren - 0.0386 * (i + start) * (i + start) + 22.941 * (i +
start) - 2898.7;

```



```

        if (MasDay[i] == 1)
        {
            ZroblPlowa += (pl > DayProductivity) ? DayProductivity : pl;
            zbir += ((pl > DayProductivity) ? DayProductivity : pl) * CurrentMasa *
80000 / 100000;
            pl -= (pl > DayProductivity) ? DayProductivity : pl;
        }
        day++;
    }
    else break;
}
if (det) sw.WriteLine(pl);
if (pl > 0)
{
    //втрата в ц.
    vtratu = pl * CurrentMasa * 80000 / 100000;
}
nesvoechasni = (int)((pl / (productivity * truvZminu) > (int)(pl / (productivity *
truvZminu))) ?
    (int)(pl / (productivity * truvZminu)) + 1.0 : (int)(pl / (productivity *
truvZminu)));
    //sw.WriteLine("{0} днів\t{1} днів\t{2} га\t{3:n2} ц\t{4:n2} ц\t{5} га", day,
nesvoechasni,
    // ZroblPlowa,(zbir/100000),vtratu/100000, (pl<=0)?0:pl);
    //sw.WriteLine("Урожайність {0:n2} ц/га", CurrentMasa * 42857 / 100000);
    return ZroblPlowa;
}

void TEP()
{
    masExpluatZatratu = new float[masPlowa.Length];

    for (int i = 0; i < masExpluatZatratu.Length; ++i)
    {
        koevVidrah = masFactPlowa[i] / (productivity * normNaprac);
        masExpluatZatratu[i] = (float)((osnPrac * tarStavOP + dopPrac * tarStavDP) /
productivity + (CinaPMM * vutrPMM) + (BalVartMash * AmortuzVidrah / (100 * masFactPlowa[i])) +
        (BalVartMash * PotRemVidrah * koevVidrah / (100 * productivity * normNaprac)));
        sw.WriteLine("{0:f2}", masExpluatZatratu[i]);
    }
}

private void Start_Click(object sender, EventArgs e)
{
    //----ініціалізація ProgresBar-----
    prFull.Maximum = ((Max_area - area) / step_area + 1) * itter * itter * itter;
    prFull.Value = 0;
    //-TOTAL-
    prPogUmovu.Maximum = itter;
    //-----
    masVtratuPercent = new float[(Max_area - area) / step_area + 1];
    masPlowa = new int[(Max_area - area) / step_area + 1];
    masFactPlowa = new float[(Max_area - area) / step_area + 1];
    masVtratu = new float[(Max_area - area) / step_area + 1];
    dani d1 = new dani(area, Max_area, step_area, itter);
    WorkThread = new Thread(new ParameterizedThreadStart(model));
    WorkThread.IsBackground = true;
    WorkThread.Start(d1);
    //model(area, Max_area, step_area, itter);
    modDone = false;
    //-----buttons-----
    Start.Enabled = false;
    Pause.Enabled = true;
    Stop.Enabled = true;
}

private void Pause_Click(object sender, EventArgs e)
{
    sleep = !sleep;
}

```

```

    Pause.Text = (sleep) ? "Продовжити" : "Призупинити";
}

private void Stop_Click(object sender, EventArgs e)
{
    WorkThread.Abort();
    Start.Enabled = true;
    sleep = false;
    Pause.Text = "Призупинити";
    prFull.Value = 0;
    prPogUmovu.Value = 0;
}

private void вихідніДаніToolStripMenuItem_Click(object sender, EventArgs e)
{
    Data d1 = new Data(this);
    d1.Show();
}

private void ймовірністьВтратToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (!modDone)
        MessageBox.Show("Моделювання не проведене!", "Увага", MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    else
    {
        imovirnist g = new imovirnist(this);
        g.Show();
    }
}

private void технікоЕкономічніПоказникиToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (!modDone)
        MessageBox.Show("Моделювання не проведене!", "Увага", MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    else
    {
        TEP t1 = new TEP(this);
        t1.Text = "Тривалість роботи: " + truvZminu + " год." + " Початок збирання: " +
        start + "день";
        t1.Show();
    }
}
}
}
}

```

ФРАГМЕНТ КОДУ ВІКНА ФОРМИ – Form1.Designer.cs

```

namespace Dip
{
    partial class Form1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed; otherwise,
false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            System.ComponentModel.ComponentResourceManager resources = new
System.ComponentModel.ComponentResourceManager(typeof(Form1));
            this.menuStrip1 = new System.Windows.Forms.MenuStrip();
            this.менюToolStripMenuItem = new System.Windows.Forms.ToolStripItem();
            this.вихідніДаніToolStripMenuItem = new System.Windows.Forms.ToolStripItem();
            this.графікиToolStripMenuItem = new System.Windows.Forms.ToolStripItem();
            this.ймовірністьВтратToolStripMenuItem = new System.Windows.Forms.ToolStripItem();
            this.технікоЕкономічніПоказникиToolStripMenuItem = new
System.Windows.Forms.ToolStripItem();
            this.Start = new System.Windows.Forms.Button();
            this.prFull = new System.Windows.Forms.ProgressBar();
            this.label1 = new System.Windows.Forms.Label();
            this.prPogUmovu = new System.Windows.Forms.ProgressBar();
            this.label2 = new System.Windows.Forms.Label();
            this.groupBox1 = new System.Windows.Forms.GroupBox();
            this.Stop = new System.Windows.Forms.Button();
            this.Pause = new System.Windows.Forms.Button();
            this.menuStrip1.SuspendLayout();
            this.groupBox1.SuspendLayout();
            this.SuspendLayout();
            //
            // menuStrip1
            //
            this.menuStrip1.Items.AddRange(new System.Windows.Forms.ToolStripItem[] {
            this.менюToolStripMenuItem,
            this.графікиToolStripMenuItem});
            this.menuStrip1.Location = new System.Drawing.Point(0, 0);
            this.menuStrip1.Name = "menuStrip1";
            this.menuStrip1.Size = new System.Drawing.Size(298, 24);
            this.menuStrip1.TabIndex = 0;
            this.menuStrip1.Text = "menuStrip1";
            //
            // менюToolStripMenuItem
            //

```

```

        this.менюToolStripMenuItem.DropDownItems.AddRange(new
System.Windows.Forms.ToolStripItem[] {
        this.вихідніДаніToolStripMenuItem});
        this.менюToolStripMenuItem.Name = "менюToolStripMenuItem";
        this.менюToolStripMenuItem.Size = new System.Drawing.Size(48, 20);
        this.менюToolStripMenuItem.Text = "&Меню";
        //
        // вихідніДаніToolStripMenuItem
        //
        this.вихідніДаніToolStripMenuItem.Name = "вихідніДаніToolStripMenuItem";
        this.вихідніДаніToolStripMenuItem.Size = new System.Drawing.Size(160, 22);
        this.вихідніДаніToolStripMenuItem.Text = "Початкові дані";
        this.вихідніДаніToolStripMenuItem.Click += new
System.EventHandler(this.вихідніДаніToolStripMenuItem_Click);
        //
        // графікиToolStripMenuItem
        //
        this.графікиToolStripMenuItem.DropDownItems.AddRange(new
System.Windows.Forms.ToolStripItem[] {
        this.ймовірністьВтратToolStripMenuItem,
        this.технікоЕкономічніПоказникиToolStripMenuItem});
        this.графікиToolStripMenuItem.Name = "графікиToolStripMenuItem";
        this.графікиToolStripMenuItem.Size = new System.Drawing.Size(59, 20);
        this.графікиToolStripMenuItem.Text = "Графіки";
        //
        // ймовірністьВтратToolStripMenuItem
        //
        this.ймовірністьВтратToolStripMenuItem.Name = "ймовірністьВтратToolStripMenuItem";
        this.ймовірністьВтратToolStripMenuItem.Size = new System.Drawing.Size(235, 22);
        this.ймовірністьВтратToolStripMenuItem.Text = "Ймовірність свочасних робіт";
        this.ймовірністьВтратToolStripMenuItem.Click += new
System.EventHandler(this.ймовірністьВтратToolStripMenuItem_Click);
        //
        // технікоЕкономічніПоказникиToolStripMenuItem
        //
        this.технікоЕкономічніПоказникиToolStripMenuItem.Name =
"технікоЕкономічніПоказникиToolStripMenuItem";
        this.технікоЕкономічніПоказникиToolStripMenuItem.Size = new System.Drawing.Size(235,
22);
        this.технікоЕкономічніПоказникиToolStripMenuItem.Text = "Техніко-Економічні показники";
        this.технікоЕкономічніПоказникиToolStripMenuItem.Click += new
System.EventHandler(this.технікоЕкономічніПоказникиToolStripMenuItem_Click);
        //
        // Start
        //
        this.Start.Location = new System.Drawing.Point(6, 18);
        this.Start.Name = "Start";
        this.Start.Size = new System.Drawing.Size(80, 23);
        this.Start.TabIndex = 0;
        this.Start.Text = "Почати";
        this.Start.UseVisualStyleBackColor = true;
        this.Start.Click += new System.EventHandler(this.Start_Click);
        //
        // prFull
        //
        this.prFull.ForeColor = System.Drawing.Color.Green;
        this.prFull.Location = new System.Drawing.Point(15, 151);
        this.prFull.Maximum = 500;
        this.prFull.Name = "prFull";
        this.prFull.Size = new System.Drawing.Size(268, 23);
        this.prFull.Step = 1;
        this.prFull.Style = System.Windows.Forms.ProgressBarStyle.Continuous;
        this.prFull.TabIndex = 1;
        //
        // label1
        //
        this.label1.AutoSize = true;
        this.label1.Location = new System.Drawing.Point(15, 135);
        this.label1.Name = "label1";

```

```

this.label1.Size = new System.Drawing.Size(150, 13);
this.label1.TabIndex = 4;
this.label1.Text = "Загальний хід моделювання";
//
// prPogUmovu
//
this.prPogUmovu.BackColor = System.Drawing.SystemColors.Control;
this.prPogUmovu.Location = new System.Drawing.Point(15, 103);
this.prPogUmovu.Maximum = 10;
this.prPogUmovu.Name = "prPogUmovu";
this.prPogUmovu.Size = new System.Drawing.Size(268, 23);
this.prPogUmovu.Step = 1;
this.prPogUmovu.TabIndex = 3;
//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(16, 85);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(151, 13);
this.label2.TabIndex = 2;
this.label2.Text = "Моделювання погодніх умов";
//
// groupBox1
//
this.groupBox1.Controls.Add(this.Stop);
this.groupBox1.Controls.Add(this.Pause);
this.groupBox1.Controls.Add(this.Start);
this.groupBox1.Location = new System.Drawing.Point(15, 35);
this.groupBox1.Name = "groupBox1";
this.groupBox1.Size = new System.Drawing.Size(268, 47);
this.groupBox1.TabIndex = 1;
this.groupBox1.TabStop = false;
this.groupBox1.Text = "Моделювання";
//
// Stop
//
this.Stop.Location = new System.Drawing.Point(182, 18);
this.Stop.Name = "Stop";
this.Stop.Size = new System.Drawing.Size(80, 23);
this.Stop.TabIndex = 2;
this.Stop.Text = "Зупинити";
this.Stop.UseVisualStyleBackColor = true;
this.Stop.Click += new System.EventHandler(this.Stop_Click);
//
// Pause
//
this.Pause.Location = new System.Drawing.Point(92, 18);
this.Pause.Name = "Pause";
this.Pause.Size = new System.Drawing.Size(85, 23);
this.Pause.TabIndex = 1;
this.Pause.Text = "Призупинити";
this.Pause.UseVisualStyleBackColor = true;
this.Pause.Click += new System.EventHandler(this.Pause_Click);
//
// Form1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(298, 225);
this.Controls.Add(this.groupBox1);
this.Controls.Add(this.label2);
this.Controls.Add(this.prPogUmovu);
this.Controls.Add(this.label1);
this.Controls.Add(this.prFull);
this.Controls.Add(this.menuStrip1);
this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.FixedSingle;
this.Icon = ((System.Drawing.Icon)(resources.GetObject("$this.Icon")));
this.Location = new System.Drawing.Point(300, 0);

```

```
        this.MainMenuStrip = this.menuStrip1;
        this.MaximizeBox = false;
        this.Name = "Form1";
        this.Text = "Моделювання робочих процесів";
        this.menuStrip1.ResumeLayout(false);
        this.menuStrip1.PerformLayout();
        this.groupBox1.ResumeLayout(false);
        this.ResumeLayout(false);
        this.PerformLayout();
    }

#endregion

private System.Windows.Forms.MenuStrip menuStrip1;
private System.Windows.Forms.ToolStripMenuItem менюToolStripMenuItem;
private System.Windows.Forms.ToolStripMenuItem вихідніДаніToolStripMenuItem;
private System.Windows.Forms.Button Start;
private System.Windows.Forms.ProgressBar prFull;
private System.Windows.Forms.Label label1;
private System.Windows.Forms.ProgressBar prPogUmovu;
private System.Windows.Forms.Label label2;
private System.Windows.Forms.GroupBox groupBox1;
private System.Windows.Forms.Button Pause;
private System.Windows.Forms.Button Stop;
private System.Windows.Forms.ToolStripMenuItem графікиToolStripMenuItem;
private System.Windows.Forms.ToolStripMenuItem ймовірністьВтратToolStripMenuItem;
private System.Windows.Forms.ToolStripMenuItem технікоЕкономічніПоказникиToolStripMenuItem;
    }
}
```