АНОТАЦІЯ

*Зм.*

*Арк.*

*Прізвище*

*Підпис*

*Дата*

*Арк.*

2

ДП ІС-0100.1393-с.ПЗ

*Розроб.*

*Іванов І.І.*

*Перевірив.*

*.*

*Баня Є.М.*

*Н. кон.*

*Сперкач М.О.*

*Затв.*

*Баня Є.М.*

Тема згідно з наказом

*Літ.*

*Аркушів*

–

*НТУУ «КПІ ім. Ігоря Сікорського» ФІОТ*

*кафедра АСОІУ гр. ІС-31*

**Структура та обсяг роботи.** Пояснювальна записка дипломного проекту складається з шести розділів, містить ?? рисунків, ?? таблиць, ?? додатків, ?? джерел.

Дипломний проект присвячений розробці комплексу задач … . Цілі та задачі розробки.

У розділі інформаційного забезпечення …

Розділ математичного забезпечення присвячений…

Програмне забезпечення …

У технологічному розділі …

|  |
| --- |
| ТУТ МАЄ БУТИ ПЕРЕЛІК КЛЮЧОВИХ СЛІВ: ВЕЛИКИМ ЛІТЕРАМИ, ЧЕРЕЗ КОМУ. |
|  |

ABSTRACT

*Той самий текст, що й в анотації, але іноземною мовою (англійська, німецька, НЕ російська)*

ЗМІСТ

[Вступ 7](#_Toc303777083)

[1 Загальні положення 7](#_Toc303777084)

[1.1 Опис предметного середовища 7](#_Toc303777085)

[1.1.1 Опис процесу діяльності 7](#_Toc303777086)

[1.1.2 Опис функціональної моделі 7](#_Toc303777087)

[1.2 Огляд наявних аналогів 7](#_Toc303777088)

[1.3 Постановка задачі 7](#_Toc303777089)

[1.3.1 Призначення розробки 7](#_Toc303777090)

[1.3.2 Цілі та задачі розробки 7](#_Toc303777091)

Висновок до розділу

[2 Інформаційне забезпечення 7](#_Toc303777092)

[2.1 Вхідні дані 7](#_Toc303777093)

[2.2 Вихідні дані 7](#_Toc303777094)

[2.3 Опис структури бази даних 7](#_Toc303777095)

[2.4 Структура масивів інформації 7](#_Toc303777096)

[Висновок до розділу 7](#_Toc303777102)

[3 Математичне забезпечення 7](#_Toc303777097)

[3.1 Змістовна постановка задачі 7](#_Toc303777098)

[3.2 Математична постановка задачі 7](#_Toc303777099)

[3.3 Обґрунтування методу розв’язання 7](#_Toc303777100)

[3.4 Опис методу розв’язання 7](#_Toc303777101)

[Висновок до розділу 7](#_Toc303777102)

[4 Програмне та технічне забезпечення 7](#_Toc303777103)

[4.1 Засоби розробки 7](#_Toc303777104)

[4.2 Вимоги до технічного забезпечення 7](#_Toc303777105)

[4.2.1 Загальні вимоги 7](#_Toc303777106)

[4.2.2 Опис локальної обчислювальної мережі 7](#_Toc303777107)

[4.3 Архітектура програмного забезпечення 7](#_Toc303777108)

[4.3.1 Діаграма класів 7](#_Toc303777109)

[4.3.2 Діаграма послідовності 7](#_Toc303777110)

[4.3.3 Діаграма компонентів 7](#_Toc303777111)

[4.3.4 Специфікація функцій 7](#_Toc303777112)

[4.4 Опис звітів 7](#_Toc303777113)

[Висновок до розділу 7](#_Toc303777102)

[5 Технологічний розділ 7](#_Toc303777114)

[5.1 Керівництво користувача 7](#_Toc303777115)

[5.2 Випробування програмного продукту 7](#_Toc303777116)

[5.2.1 Мета випробувань 7](#_Toc303777117)

[5.2.2 Загальні положення 7](#_Toc303777118)

[5.2.3 Результати випробувань 7](#_Toc303777119)

[Висновок до розділу 7](#_Toc303777102)

[загальні висновки 7](#_Toc303777123)

[Перелік посилань 7](#_Toc303777124)

[ДОДАТОК A Тексти програмного коду 7](#_Toc303777126)

Вступ

На сьогоднішній день однією з відомих і важливих областей прикладної математики є теорія розкладів. Висока популярність даної області зумовлена, в першу чергу, існуванням великої кількості прикладних задач, які потребують ефективних методів розв’язання. В таких предметних областях як: виробництво, управління персоналом, управління проектами, транспорт, обчислювальні системи, методи розв’язання задач складання розкладів і календарного планування посідають ключове місце.

Втім, сьогодні існує багато задач, які не мають точних методів розв’язання, або точний метод існує, але він не є ефективним (його обчислювальна складність є занадто високою при реальних розмірностях вхідних даних). Для ефективного розв’язання таких задач було винайдено новий клас алгоритмів – ПДС-алгоритми (детальніше ПДС-алгоритми будуть описані в підрозділі опису предметного середовища 1-го розділу).

Беручи до уваги складність побудови алгоритмів вирішення задач складання розкладів а також те, що сьогодні є високий попит в ефективному вирішенні таких задач, існує значна потреба в інформаційній системі, яка б допомагала науковцям у розробці ПДС-алгоритмів.

Даний дипломний проект присвячений розробці системи підтримки розробки та аналізу ПДС-алгоритмів побудови розкладів робіт на паралельних верстатах однакової продуктивності.

**Практичне значення одержаних результатів.** Розроблено точний алгоритм побудови оптимальних за критерієм максимізації мінімального моменту початку роботи верстатів розкладів робіт з довільними директивними термінами на паралельних верстатах однакової продуктивності.

# Загальні положення

## **Опис предметного середовища**

Об’єктом автоматизації є наукова група, що розробляє ПДС-алгоритми побудови розкладів робіт з довільними директивними термінами на паралельних верстатах однакової продуктивності.

**ПДС-алгоритмом** називається алгоритм, що складається з поліноміальної складової і експоненціального підалгоритму чи поліноміальної апроксимації точного алгоритму, який може включати умови декомпозиції початкової задачі на підзадачі меншої розмірності [1].

Поліноміальна складова породжується логіко-аналітичними умовами (p-умовами), виконання яких допустимим рішенням, отриманим в результаті реалізації поліноміальної складової ПДС-алгоритму, визначає його як оптимальне. p-умови знаходяться в результаті теоретичних досліджень відповідного класу важковирішуваних задач комбінаторної оптимізації. Поліноміальна складова ПДС-алгоритму синтезується таким чином, щоб послідовна процедура конструювання допустимих рішень була найбільш ефективною з точки зору реалізації p-умов (достатніх ознак оптимальності допустимих рішень). Верхня оцінка складності поліноміальної складової відома. Іноді експоненціальна складова ПДС-алгоритму замінюється алгоритмом поліноміальної складності, що призводить до наближеного (субоптимального) рішення [1].

В [2] формулюється ряд нових одноетапних задач календарного планування, для кожної з яких знаходяться p-умови (ознаки оптимальності допустимого рішення). Отримані результати реалізують можливість побудови для цих класів комбінаторних задач оптимізації ефективних ПДС-алгоритмів.

Таким чином, розширюється клас задач, для яких можуть бути побудовані ПДС-алгоритми. Наведені в даному розділі задачі не досліджувалися на предмет того, до якого класу задач (P або не простіше, ніж NP-повні) вони відносяться. Однак це не є важливим по тій причині, що ПДС-алгоритми є ефективними і в наступних випадках:

* не знайдено точного поліноміального алгоритму вирішення задачі;
* точний поліноміальний алгоритм знайдений, але його обчислювальна складність є значно вищою за обчислювальну складність поліноміальної складової відповідного ПДС-алгоритму, при тому що поліноміальна складова ПДС-алгоритму є достатньо ефективною (реалізується достатньо часто для задоволення вимог конкретної предметної області).

Далі буде описано задачу складання оптимальних за векторним критерієм максимізації мінімального моменту початку роботи верстатів розкладів робіт з довільними директивними термінами на паралельних верстатах однакової продуктивності.

**Розклад** називається **допустимим**, якщо в ньому немає робіт із запізненням. **Робота** називається **допустимою**, якщо вона виконується без запізнення [3].

Нехай маємо *m* незалежних паралельних верстатів однакової продуктивності, які працюють без переривань і виконують *n* робіт ( – тривалість виконання -ї роботи, ). Роботи повинні бути виконані до директивних термінів . Моменти запуску верстатів довільні. Необхідно побудувати допустимий розклад, що максимізує наступний критерій:

|  |  |  |
| --- | --- | --- |
|  |  | (1.1) |

де – момент запуску верстата , – номер верстату, у якого момент запуску в оптимальному розкладі найбільш ранній (він є найбільш пізнім для всіх допустимих розкладів); , – номер верстату, у якого момент запуску наступний за величиною після верстатів , (він є найбільш пізнім для всіх допустимих розкладів з фіксованим ).

Допустимий розклад, у якого моментами запуску верстатів є  
, називається **оптимальним за векторним** (лексикографічним) **критерієм** максимізації мінімального моменту початку роботи верстатів [3].

### **Опис процесу діяльності**

Розглянемо дії, які виконує розробник під час основного процесу роботи в системі – аналізу нового ПДС-алгоритму.

Процес: «Аналіз алгоритму».

Передумови: користувач має бути авторизованим.

Послідовність дій розробника при тестуванні та аналізі алгоритму проілюстровано на UML діаграмі діяльності, яку наведено у частині графічного матеріалу дипломного проекту.

### **Опис функціональної моделі**

Акторами системи є:

* неавтентифікований користувач;
* розробник алгоритмів;
* адміністратор.

**Неавтентифікований користувач** має можливість створити обліковий запис та увійти в систему.

**Розробнику алгоритмів** доступні основні функції системи: ті, що стосуються його облікового запису, його алгоритмів, а також функції перегляду та аналізу публічних алгоритмів інших користувачів.

**Адміністратор** – актор, що має найбільші права і відповідальність у системі. Крім функцій розробника алгоритмів, адміністратор має безпосередній доступ до бази даних системи. Найважливіша функція адміністратора – ведення даних, що передаються на вхід алгоритмам користувачів при їх тестуванні та аналізі.

Визначимо дії, які можуть виконувати актори системи. Актори системи з їх діями зображені на схемі структурній варіантів використання у частині графічного матеріалу.

Таблиця 1.1 *–*­ Опис варіантів використання

|  |  |  |  |
| --- | --- | --- | --- |
| **Актор** | **Назва варіанту** | **Опис** | **Пріоритет** |
| Неавтентифікований користувач | Вхід у систему | Неавтентифікований користувач входить у систему | Високий |
| Створення облікового запису | Користувач без облікового запису створює обліковий запис у системі | Високий |
| Розробник алгоритмів | Ведення власних алгоритмів | Розробник алгоритмів може додавати, редагувати та видаляти власні алгоритми | Високий |
| Публікація алгоритму | Користувач може опублікувати власний алгоритм, після чого алгоритм стане доступним для перегляду іншим користувачам | Низький |
|  |  |  |  |
| Адміністратор | Видалити тренування | Адміністратор видаляє тренування що не задовольняє правилам | Високий |

## **Огляд наявних аналогів**

При розробці будь-якого програмного забезпечення, одним з етапів його проектування є пошук вже існуючих рішень. Для оцінки доцільності створення нового програмного забезпечення необхідно виявити всі переваги та недоліки існуючих програмних рішень.

Програмний продукт, що розроблюється, базується на ряді наукових робіт в області важкорозв’язуваних задач побудови розкладів. Фактично, був створений новий метод розв’язання такого роду задач – ПДС-алгоритми, аналогів якому не існує в світі.

Хоча дана предметна область є вузькою, швидкий розвиток теорії розкладів створює потребу у такого роду системі. І хоча на сьогоднішній день не існує системи, яка б повністю задовольнила всі вимоги, висунуті в лабораторній роботі, дана система буде включати та поєднувати функції, що доступні в окремих системах, такі як системи контролю версій.

Наприклад, новий програмний продукт буде включати базові функції системи контролю версій. Ілюстрацію використання однієї з таких систем – github, наведено на наступному рисунку:

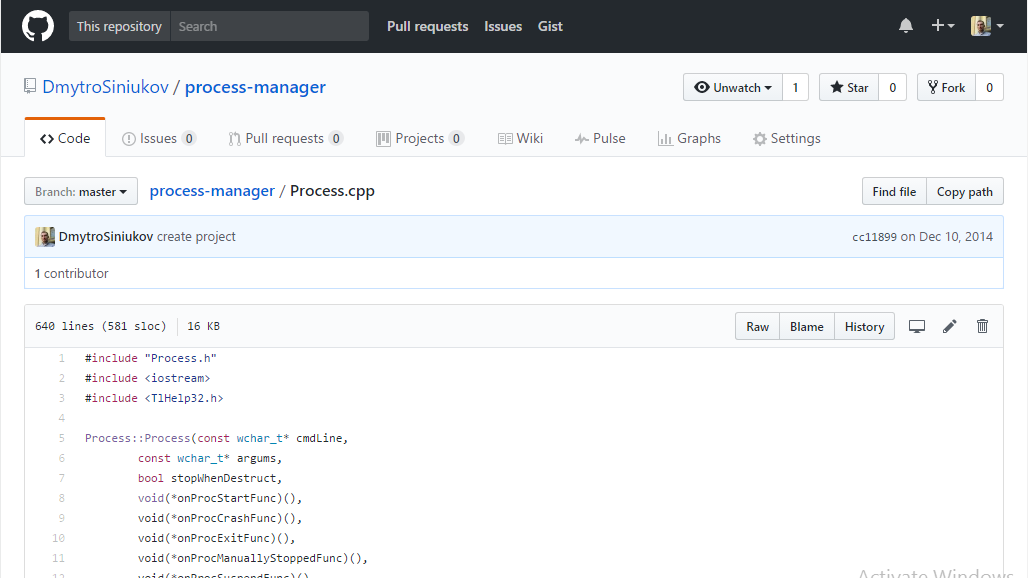


Рисунок 1.3 – Форма редагування коду у системі github

Крім того, система також включає функції онлайн інтерпретаторів коду, що дозволяють ввести і запустити користувацьку програму в режимі реального часу, не перезапускаючи весь проект.

Але, дана система має і унікальні функції, що потребуються для розробки алгоритмів саме ПДС-типу. Наприклад, аналіз ефективності поліноміальної та декомпозиційної складових ПДС-алгоритму.

Беручи до уваги складність побудови алгоритмів вирішення задач складання розкладів а також те, що сьогодні є високий попит в ефективному вирішенні таких задач (наприклад для планування робіт на виробництві), є потреба в інформаційній системі, яка б допомагала науковцям у спільній розробці ПДС-алгоритмів.

Зважаючи на те, що новий програмний продукт буде задовольняти попит, а також, що на сьогодні не існує системи, яка б одночасно включала всі необхідні функції, розробка даної системи є актуальною. Система буде ефективно поєднувати в собі комплекс функцій, що доступні у розрізнених системах, а також задовольняти вимоги, притаманні саме для розробки ПДС-алгоритмів.

## **Постановка задачі**

### **Призначення розробки**

АС «Система підтримки розробки та аналізу ПДС-алгоритмів побудови розкладів робіт на паралельних верстатах однакової продуктивності» призначена для інформаційно-аналітичного забезпечення процесів розробки ПДС-алгоритмів складання розкладів переважно науковими колективами. Процеси, що повинна забезпечити система:

* Ведення інформації про ПДС-алгоритми задач складання розкладів а також їх вихідного коду
* Ведення наборів вхідних даних для тестування та аналізу
* Автоматичне тестування та аналіз алгоритмів
* Побудова розкладу обраним алгоритмом для довільних введених вхідних даних

### **Цілі та задачі розробки**

Основними цілями розробки АС «Система підтримки розробки та аналізу ПДС-алгоритмів для однієї з задач побудови розкладів» є:

* Створення комфортних умов для командної розробки ПДС-алгоритмів
* Забезпечення накопичення та ведення інформації, пов’язаної з розробкою ПДС-алгоритмів, включаючи вхідні дані для аналізу та тестування, в одній системі
* Підвищення ефективності тестування та аналізу ПДС-алгоритмів

Для досягнення поставлених цілей мають бути вирішені такі задачі:

* Інтерпретація та виконання вихідного коду алгоритмів
* Представлення статистичної інформації аналізу алгоритмів у наглядному (графічному) вигляді
* Ефективне розділення прав користувачів різних груп: адміністратори, пересічні користувачі-розробники

## **Висновок до розділу**

*Дати висновок до цього розділу.*

# Інформаційне забезпечення

## **Вхідні дані**

При побудові розкладів робіт і аналізі ефективності алгоритмів система використовує наступні вхідні дані:

* Опис наборів вхідних даних алгоритмів
* Власне вхідні дані, що будуть передаватись на вхід алгоритмів
* Опис алгоритмів з їх вихідним кодом

## **Вихідні дані**

Основними вихідними даними є графіки аналізу ефективності алгоритмів і розклади робіт, побудовані обраним алгоритмом на довільних вхідних даних.

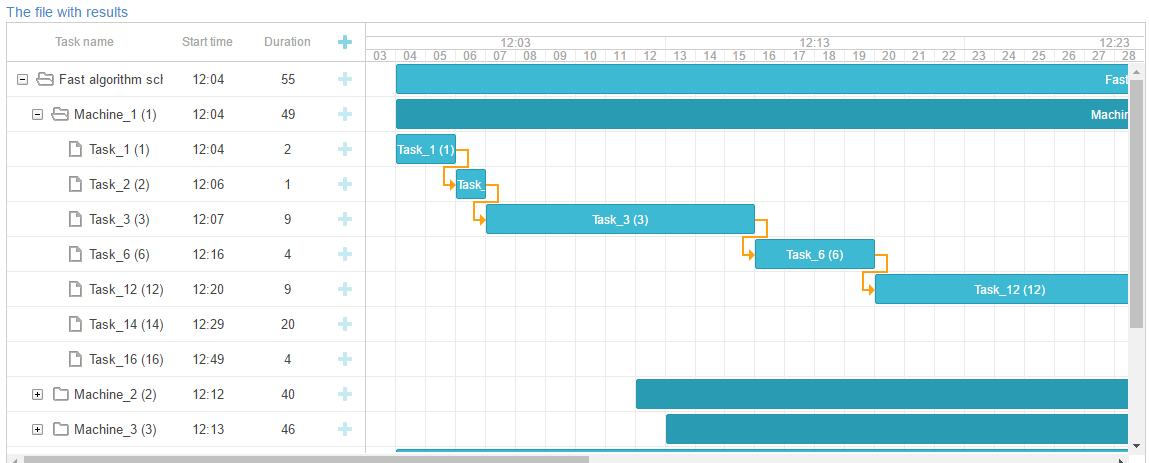


Рисунок 2.1 – Діаграма Ганта побудованого розкладу робіт

## **Опис структури бази даних**

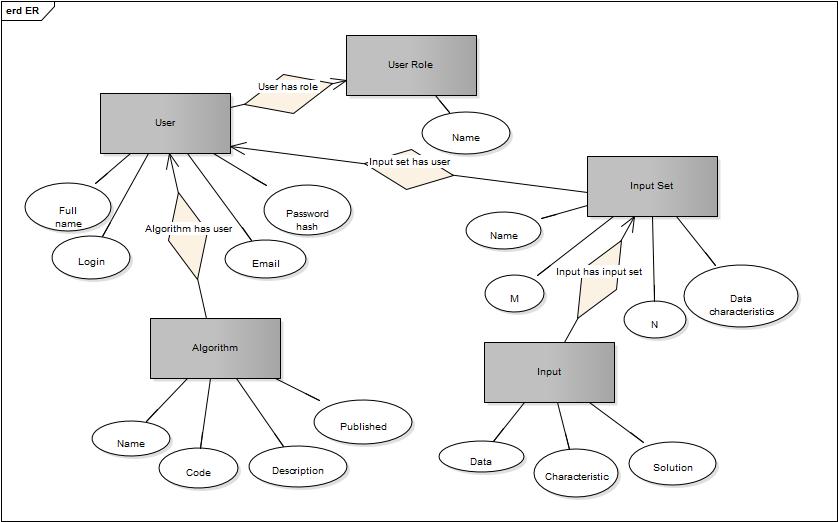


Рисунок 2.1 – ER-діаграма бази даних

Таблиця 2.1 – Опис сутностей

|  |  |
| --- | --- |
| **Назва** | **Призначення** |
| Користувач (User) | Представляє користувача системи |
| Роль користувача (User Role) | Описує роль користувача системи (розробник алгоритмів, адміністратор) |
| Алгоритм (Algorithm) | Містить інформацію про ПДС-алгоритм |
| Вхідні дані (Input ) | Представляє вхідні дані для одного запуска алгоритму |
| Набір вхідних даних (Input Set) | Містить інформацію про один набір вхідних даних зі спільними характеристиками, що будуть використані для побудови графіку |

## **Структура масивів інформації**

*Якщо БД не має, то відображаємо структуру даних (масивів інформації). Наприклад, у вигляді ХМL- розмітки.*

## **Висновок до розділу**

*Дати висновок до цього розділу.*

# Математичне забезпечення ???

## **Змістовна постановка задачі**

## **Математична постановка задачі**

## **Обґрунтування методу розв’язання**

## **Опис методів розв’язання**

## **Висновок до розділу**

# Програмне та технічне забезпечення

## **Засоби розробки**

## **Вимоги до технічного забезпечення**

### **Загальні вимоги**

### **Опис локальної обчислювальної мережі**

*Описувати тільки у випадку коли комплекс задач:*

* *має трирівневу архітектуру;*
* *потребує використання локальної мережі.*

## **Архітектура програмного забезпечення**

### **Діаграма класів**

### **Діаграма послідовності**

### **Діаграма компонентів**

*В підрозділах 4.3.1-4.3.3 наводиться ілюстрація побудованих діаграм та їх повний опис.*

### **Специфікація функцій**

*В даному підрозділі описуються функції класів програмного забезпечення.*

## **Опис звітів**

## **Висновок до розділу**

*Дати висновок до цього розділу.*

# Технологічний розділ

## **Керівництво користувача**

*В підрозділі описуємо основні функції, які були реалізовані відповідно постановки задачі та підтверджуємо їх зображеннями екранних форм.*

## **Випробування програмного продукту**

*В цьому підрозділі наведено опис тестів і порядок їх виконання для перевірки відповідності програмного забезпечення комплексу задач функціональним вимогам, представленим у технічному завданні на створення вибраного Вами комплексу задач.*

### **Мета випробувань**

Метою випробувань являється перевірка відповідності функцій комплексу задач підтримки діяльності фірми з прокату автомобілів вимогам технічного завдання.

### **Загальні положення**

Випробування проводяться на основі наступних документів:

* ГОСТ 34.603−92. Інформаційна технологія. Види випробувань автоматизованих систем;
* ГОСТ РД 50-34.698-90. Автоматизовані системи вимог до змісту документів.

### **Результати випробувань**

## **Висновок до розділу**

*Дати висновок до цього розділу.*

ЗАГАЛЬНІ ВИСНОВКИ

Перелік посилань

1. Згуровский М.З., Павлов А.А. Принятие решений в сетевых системах с ограниченными ресурсами: Монография.– К.: Наукова думка. – 2010. – 573 с.
2. Павлов А.А. Признаки оптимальности допустимых решений труднорешаемых задач комбина-торной оптимизации // Вісник НТУУ “КПІ”. Серія «Інформатика, управління та обчислю¬вальна техніка». – К.: “ВЕК+”, 2013. – №59 – С.4–11.
3. Згуровский М.З. Минимизация лексикографического критерия для допустимого расписания на независимых параллельных приборах с произвольными директивными сроками [Текст] / М.З. Згуровский, А.А. Павлов, Е.Б. Мисюра // Вісник НТУУ “КПІ”. Серія «Інформатика, управління та обчислювальна техніка». – К.: “ВЕК+”, 2014. – №61. – С.4–17.
4. афвафівафі

Додаток А

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

16

ДП ІС-0100.1393-с.ПЗ

(Найменування програми (документа))

***Тексти програмного коду***

***Система моніторингу стану працездатності банкоматної мережі***

(Вид носія даних)

*DVD-R*

(Обсяг програми (документа) , арк.,) Кб)

*73 арк, 244 Кб*

Київ – 2017 року