

«Системне програмування»

КІ-32, Дмитрик Валерій

Варіанти запитань

N	Пункти		
	Теорія		Практика
1	11	37	8
2	14	23	27
3	7	20	1
4	4	3	31
5	25	11	34
6	39	13	25
7	5	19	21
8	40	1	18
9	16	34	28
10	38	14	10
11	33	22	36
12	17	8	29
13	19	32	11
14	2	31	35

Варіант 3

Теорія 1 (7)

Розширення регулярних виразів

Регулярні вирази - це шаблони символів, які використовуються для пошуку та вилучення певних частин тексту. Вони дозволяють визначати складні шаблони, такі як послідовності символів, діапазони, повторення та багато іншого.

Регулярні вирази будуються з більш простих регулярних виразів з використанням набору правил. Кожний регулярний вираз r задає мову $L(r)$. Правила визначають, яким чином з мов, заданих підвиразами r , формуються $L(r)$.

Розширення регулярних виразів включає в себе використання різних конструкцій та скорочень для зручності визначення мов. Давайте розглянемо детальніше розширення регулярних виразів:

1. Символи та Класи символів:

- У регулярних виразах можна використовувати символи з абетки, наприклад, якщо a - символ з абетки Σ , то " a " є регулярним виразом, що позначає множину $\{a\}$.

- Також можна використовувати класи символів, наприклад, $[abc]$ позначає множину символів, які можуть бути " a ", " b " або " c ", а $[0-9]$ позначає множину цифр від 0 до 9.

2. Операції над регулярними виразами:

- Об'єднання ($|$): При використанні оператора $|$ можна об'єднати два регулярних вирази, що позначають мови $L(r)$ та $L(s)$, і отримати новий вираз, який позначає об'єднання мов $L(r) \cup L(s)$.

- Зчеплення (конкатенація) (\cdot): Операція зчеплення дозволяє об'єднати два регулярних вирази, що позначають мови $L(r)$ та $L(s)$, і отримати новий вираз, який позначає зчеплення мов $L(r) \cdot L(s)$.

- Ітерація ($*$): Операція ітерації дозволяє створити регулярний вираз, який позначає 0 або більше повторень мови, що відповідає початковому виразу.

3. Порожня стрічка та Спеціальні символи:

- Порожня стрічка (ϵ): Порожня стрічка позначається як ϵ і задає стрічку нульової довжини.

- Спеціальні символи: У регулярних виразах можна використовувати спеціальні символи для позначення певних шаблонів, наприклад, $*$ для ітерації, $?$ для нульового або одного входження тощо.

4. Визначення та Іменовані вирази:

- Для зручності можна давати імена та визначати регулярні визначення.

Це дозволяє створювати складніші вирази шляхом поєднання вже визначених виразів.

Розширення регулярних виразів дозволяє зручно виражати складні шаблони та умови для пошуку та обробки текстової інформації. Використання цих конструкцій полегшує роботу з лексичним аналізом та обробкою стрічок у програмуванні.

В програмному засобі LEX для створення лексичних аналізаторів використовується розширений набір скорочень для регулярних виразів.

Теорія 2 (20)

Детермінований скінченний автомат

Детермінований скінчений автомат (ДСА) - це математична модель, яка використовується для автоматичного розпізнавання рядків символів. Він складається зі станів, переходів між станами та вхідних символів. Основна відмінність ДСА від недетермінованого скінченного автомата (НСА) полягає в тому, що для кожного стану і символу вхідного алфавіту в ДСА існує лише один можливий перехід.

У ДСА кожен стан має чітко визначений перехід для кожного можливого вхідного символу, що робить його простим у реалізації та розумінні. Коли автомат отримує вхідну послідовність символів, він переходить від одного стану до іншого відповідно до правил переходів, поки не досягне кінцевого стану. Якщо автомат опиняється у кінцевому стані після обробки всіх вхідних символів, то він приймає вхідну послідовність, інакше відхиляє.

ДСА є швидшими у роботі порівняно з НСА, оскільки вони не потребують вибору між можливими переходами. Однак ДСА можуть бути більшими за розміром, оскільки кожен стан повинен мати перехід для кожного символу вхідного алфавіту. Тому важливо вміло побудувати та мінімізувати ДСА для ефективної роботи з рядками символів.

Детермінований скінчений автомат (ДСА) має кілька особливостей, які визначають його унікальність та ефективність у роботі з рядками символів:

1. Однозначність переходів: ДСА має чітко визначені переходи для кожного стану і символу вхідного алфавіту. Це означає, що для кожного стану і символу існує лише один можливий перехід, що спрощує роботу автомата.
2. Стартовий та кінцеві стани: ДСА має один початковий стан, з якого починається обробка вхідних символів, а також множину кінцевих станів, які позначають успішне завершення обробки вхідної послідовності.
3. Детермінованість: Оскільки ДСА має однозначні переходи, він є детермінованим, тобто для кожного стану і символу вхідного алфавіту завжди існує один єдиний наступний стан.

4. Простота реалізації: Благодаря однозначним переходам та чіткій структурі, ДСА легше реалізувати в програмному коді, що робить їх популярними для автоматизації обробки рядків символів.

5. Ефективність: ДСА є швидшими у роботі порівняно з недетермінованими скінченими автоматами, оскільки не потребують вибору між можливими переходами.

Загалом, детермінований скінчений автомат є потужним інструментом для роботи з рядками символів, завдяки своїм особливостям, які спрощують розпізнавання та обробку вхідних послідовностей.

Практика (1)

Завдання 1.

Описати мови, які відповідають регулярним виразам:

1. $a(a|b)^*a$;

2. $((\epsilon|a)b^*)^*$;

3. $(a|b)^*a(a|b)(a|b)$;

1. Регулярний вираз $a(a|b)^*a$:

- Цей вираз починається з символу 'a', потім може містити будь-яку кількість 'a' або 'b' (включаючи порожню стрічку) і завершується на 'a'.

- Таким чином, цей вираз відповідає мові, яка складається зі стрічок, що починаються та закінчуються на 'a', де середина може містити будь-яку кількість 'a' та 'b'.

2. Регулярний вираз $((\epsilon|a)b^*)^*$:

- Цей вираз описує мову, яка складається з будь-якої кількості груп, де кожна група може бути порожньою або починатися з 'a', а потім містити будь-яку кількість 'b'.

- Оскільки кожна група може бути порожньою або містити 'a' та 'b', то цей вираз відповідає мові, яка містить будь-які комбінації 'a' та 'b', включаючи порожню стрічку.

3. Регулярний вираз $(a|b)^*a(a|b)(a|b)$:

- Цей вираз описує мову, яка починається з будь-якої кількості 'a' або 'b', потім містить 'a', далі може бути 'a' або 'b', і завершується на 'a' або 'b'.

- Цей вираз відповідає мові, яка містить стрічки, що починаються з будь-якої кількості 'a' або 'b', містять 'a' у середині, а потім можуть мати будь-яку комбінацію 'a' та 'b' на кінці.