(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0072216
(43) 공개일자 2025년05월23일

(51) 국제특허분류(Int. Cl.)
G06F 16/903 (2019.01) *G06F 16/901* (2019.01)
G06F 7/76 (2006.01)
(52) CPC특허분류
G06F 16/90335 (2019.01)
G06F 16/9024 (2019.01)
(21) 출원번호 10-2023-0159154
(22) 출원일자 2023년11월16일
심사청구일자 2023년11월16일

(71) 출원인
인하대학교 산학협력단
인천광역시 미추홀구 인하로 100(용현동, 인하대학교)
(72) 발명자
심정섭
인천광역시 남동구 남동대로 860, 120동 601호 (간석동, 간석래미안자이아파트)
박경빈
인천광역시 미추홀구 인하로91번길 130, 401호 (용현동)
(74) 대리인
양성보

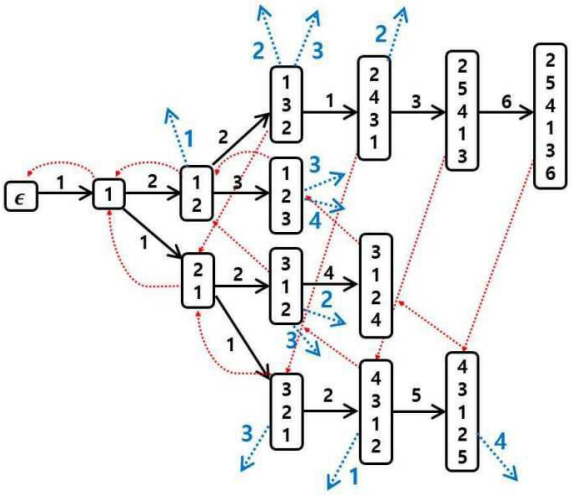
전체 청구항 수 : 총 7 항

(54) 발명의 명칭 **극소불포함순위문자열을 찾는 알고리즘**

(57) 요 약

극소불포함순위문자열을 찾는 알고리즘이 개시된다. 일 실시예에 따른 문자열 탐색 시스템에 의해 수행되는 극소불포함순위문자열 탐색 방법은, 문자열의 순위그래프를 생성하는 단계; 및 상기 생성된 문자열의 순위그래프를 이용하여 문자열의 극소불포함순위문자열의 집합을 탐색하는 단계를 포함할 수 있다.

대 표 도 - 도6



(52) CPC특허분류
G06F 7/76 (2021.08)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711179347
과제번호	00155915
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	인공지능융합혁신인재양성
연구과제명	[Ezbaro][정부+민간] 인공지능융합혁신인재양성(2차년도)
기 여 율	1/1
과제수행기관명	인하대학교 산학협력단
연구기간	2023.01.01 ~ 2023.12.31

명 세 서

청구범위

청구항 1

문자열 탐색 시스템에 의해 수행되는 극소불포함순위문자열 탐색 방법에 있어서,

문자열의 순위그래프를 생성하는 단계; 및

상기 생성된 문자열의 순위그래프를 이용하여 문자열의 극소불포함순위문자열의 집합을 탐색하는 단계

를 포함하는 극소불포함순위문자열 탐색 방법.

청구항 2

제1항에 있어서,

상기 생성하는 단계는,

순열의 마지막 원소로 순위를 추가한 순열을 획득하기 위한 제1함수를 정의하는 단계

를 포함하고,

상기 제1 함수는 $S_n \times [n+1] \rightarrow S_{n+1}$을 의미하는 것을 특징으로 하는 극소불포함순위문자열 탐색 방법.

청구항 3

제1항에 있어서,

상기 생성하는 단계는,

순열의 첫 번째 원소를 삭제한 순열을 획득하기 위한 제2함수를 정의하는 단계

를 포함하고,

상기 제2 함수는, $S_n \rightarrow S_{n-1}$을 의미하는 것을 특징으로 하는 극소불포함순위문자열 탐색 방법.

청구항 4

제1항에 있어서,

상기 생성하는 단계는,

제2 함수에 대한 제1 함수의 순열과, 제 1함수에 대한 2 함수와 제3함수의 값을 이용한 순열이 동치인 순열을 획득하기 위한 제3함수를 정의하는 단계

를 포함하고,

제3 함수는, $S_n \times [n+1] \rightarrow [n]$ 을 의미하는 것을 특징으로 하는 극소불포함순위문자열 탐색 방법.

청구항 5

제1항에 있어서,

상기 생성하는 단계는,

문자열 X에 대해, 문자열 X의 모든 부분문자열의 서로 다른 자연순위 집합을 N_X라고 할 때, 순열과 대응되는 정점과 정점에 대응되는 순열을 구성하고, 상기 구성된 정점에 제1 함수, 제2 함수 및 제3 함수를 이용하여 생성된 간선을 연결하여 문자열의 순위그래프를 생성하는 단계

를 포함하는 극소불포함순위문자열 탐색 방법.

청구항 6

제1항에 있어서,

상기 탐색하는 단계는,

상기 생성된 문자열의 순위그래프에서, 정점에서 나가는 간선들에 대해 간선의 정수 문자들의 집합과, 루트 정점에서 정점까지 간선들로 연결된 경로의 길이를 이용하여 문자열의 모든 극소불포함순위문자열의 집합을 탐색하는 단계

를 포함하는 극소불포함순위문자열 탐색 방법.

청구항 7

문자열 탐색 시스템에 있어서,

문자열의 순위그래프를 생성하는 그래프 생성부; 및

상기 생성된 문자열의 순위그래프를 이용하여 문자열의 극소불포함순위문자열의 집합을 탐색하는 문자열 탐색부

를 포함하는 문자열 탐색 시스템.

발명의 설명

기 술 분 야

[0001] 아래의 설명은 극소불포함순위문자열을 탐색하는 기술에 관한 것이다.

배 경 기 술

[0003] 두 문자열 x와 w에 대해, w가 x의 부분문자열이 아니면 w는 x의 불포함문자열(absent word)이라 한다. 불포함문자열 w에 대해, w의 가장 긴 진접두사와 진접미사가 모두 x의 부분문자열일 경우 w는 x의 극소불포함문자열(minimal absent word)이라고 한다. 문자열 X가 주어졌을 때, X의 모든 극소불포함문자열의 집합을 찾는 문제는 데이터 압축, 생물정보학 등 여러 분야에서 널리 응용되고 있다. 극소불포함문자열문제(minimal absent word problem)란, Σ로 이루어진 문자열 X가 주어졌을 때, X의 모든 극소불포함문자열의 집합을 찾는 문제이다. 일례로, X의 DAWG(directed acyclic word graph)를 이용하여 각각 $O(|\Sigma||X|)$, $O(|X|)$시간에 문제를 해결하는 알고리즘이 제시되 바 있다.

[0004] 두 문자열 x와 y의 길이가 같고, 두 문자열의 상대적 순위가 모두 동일하다면 x와 y 는 순위동형(order-iosmorphic)이라 한다. 순위패턴매칭문제(order-preserving pattern matching problem)란, 긴 문자열 T와 짧은 문자열 P에 대해, P와 순위동형인 T의 모든 부분문자열을 찾는 문제이다. 이 문제는 주가지수 분석, 멜로디 분석 등 시계열 데이터 분석에 사용될 수 있다. 이러한 순위패턴매칭문제를 해결하기 위해 종래에는 KMP 알고리즘의 실패함수를 이용하여 $O(|T|+sort(|P|))$ 시간에 해결하는 알고리즘, Boyer-Moore-Horspool 알고리즘과 q-그램을 이용하여 $O(mn+nq\log q+q!)$ 시간에 문제를 해결하는 알고리즘이 제안된 바 있다.

발명의 내용

해결하려는 과제

[0006] 문자열에 대한 극소불포함순위문자열을 정의하고, 문자열의 모든 극소불포함순위문자열의 집합 $MAOP_X$를 탐색할 수 있다.

과제의 해결 수단

[0008] 문자열 탐색 시스템에 의해 수행되는 극소불포함순위문자열 탐색 방법은, 문자열의 순위그래프를 생성하는 단계; 및 상기 생성된 문자열의 순위그래프를 이용하여 문자열의 극소불포함순위문자열의 집합을 탐색하는 단계를 포함할 수 있다.

[0009] 상기 생성하는 단계는, 순열의 마지막 원소로 순위를 추가한 순열을 획득하기 위한 제1함수를 정의하는 단계를 포함하고, 상기 제1 함수는 $S_n \times [n+1] \rightarrow S_{n+1}$을 의미하는 것일 수 있다.

[0010] 상기 생성하는 단계는, 순열의 첫 번째 원소를 삭제한 순열을 획득하기 위한 제2함수를 정의하는 단계를 포함하고, 상기 제2 함수는, $S_n \rightarrow S_{n-1}$을 의미하는 것일 수 있다.

[0011] 상기 생성하는 단계는, 제2 함수에 대한 제1 함수의 순열과, 제 1함수에 대한 2 함수와 제3함수의 값을 이용한 순열이 동치인 순열을 획득하기 위한 제3함수를 정의하는 단계를 포함하고, 제3 함수는, $S_n \times [n+1] \rightarrow [n]$ 을 의미하는 것일 수 있다.

[0012] 상기 생성하는 단계는, 문자열 X에 대해, 문자열 X의 모든 부분문자열의 서로 다른 자연순위 집합을 N_X라고 할 때, 순열과 대응되는 정점과 정점에 대응되는 순열을 구성하고, 상기 구성된 정점에 제1 함수, 제2 함수 및 제3 함수를 이용하여 생성된 간선을 연결하여 문자열의 순위그래프를 생성하는 단계를 포함할 수 있다.

[0013] 상기 탐색하는 단계는, 상기 생성된 문자열의 순위그래프에서, 정점에서 나가는 간선들에 대해 간선의 정수 문자들의 집합과, 루트 정점에서 정점까지 간선들로 연결된 경로의 길이를 이용하여 문자열의 모든 극소불포함순위문자열의 집합을 탐색하는 단계를 포함할 수 있다.

[0014] 문자열 탐색 시스템은, 문자열의 순위그래프를 생성하는 그래프 생성부; 및 상기 생성된 문자열의 순위그래프를 이용하여 문자열의 극소불포함순위문자열의 집합을 탐색하는 문자열 탐색부를 포함할 수 있다.

발명의 효과

[0016] 극소불포함순위문자열문제를 해결하기 위해 문자열의 순위그래프를 정의하고, 정의된 순위그래프에 기초하여 극소불포함순위문자열을 계산할 수 있다.

도면의 간단한 설명

[0018] 도 1은 일 실시예에 있어서, 극소불포함순위문자열을 설명하기 위한 예이다.

도 2는 일 실시예에 있어서, 문자열 탐색 시스템의 구성을 설명하기 위한 블록도이다.

도 3은 일 실시예에 있어서, 극소불포함순위문자열 탐색 방법을 설명하기 위한 흐름도이다.

도 4 및 도 5는 일 실시예에 있어서, 순위그래프를 설명하기 위한 예이다.

도 6은 일 실시예에 있어서, 극소불포함순위문자열의 집합을 탐색하는 동작을 설명하기 위한 예이다.

발명을 실시하기 위한 구체적인 내용

[0019] 이하, 실시예를 첨부한 도면을 참조하여 상세히 설명한다.

[0021] 도 1은 일 실시예에 있어서, 극소불포함순위문자열을 설명하기 위한 예이다.

[0022] 극소불포함순위문자열을 탐색하는 동작의 설명의 이해를 돕기 위해 불포함문자열과 극소불포함문자열, 순위동형과 순위표현법에 대하여 우선적으로 설명하기로 한다.

[0023] 정수 문자로 이루어진 집합을 Σ, Σ 로 구성된 문자열들의 집합을 Σ^* 라 한다. $x \in \Sigma^*$에 대해, $|x|$는 x의 크기를, $x_i(1 \le i \le |x|)$는 x의 i번째 문자를 의미한다. $x[i..j](1 \le i \le j \le |x|)$는 x의 i번째부터 j번째 문자로 구성된 부분문자열을 의미한다. j>i인 $x[i..j]$는 빈 문자열로, 빈 문자열은 모든 문자열의 부분문자열로 정의한다. 1부터 n까지의 정수들로 이루어진 집합을 [n], [n]으로 이루어진 모든 순열들의 집합을 S_n이라 한다. 편의상, 빈 문자열의 순열은 ϵ으로 정의한다.

[0024] 문자열 $x \in \Sigma^*$에 대해, 어떤 문자열 $w \in \Sigma^*(|w| \ge 1)$가 x의 부분문자열이 아니면, w는 x의 불포함문자열이라고 한다. w가 x의 불포함문자열이며, $w[1..|w|-1]$과 $w[2..|w|]$가 x의 부분문자열이면, w는 x의 극소불포함문자열이라 한다. 예를 들어, $\Sigma = \{1,2\}$, $x = (1,2,1,2,2,2,1)$에 대해, x의 극소불포함문자열은 (1, 1), (1, 2, 2, 1), (2, 1, 2, 1), (2, 2, 1, 2), (2, 2, 2, 2)로, 총 5개의 극소불포함문자열이 존재한다

[0025] 문자열 $x \in \Sigma^*$의 순위는 여러 표현법으로 나타낼 수 있다. x의 각 문자 x_i의 상대적 순위를 나열한 순열을 자연순위라 하고, $\pi(x)$라 표기하기로 한다. 정의는 다음과 같다.

[0026] $$\pi(x)_i = \left|\{x_j \mid x_j < x_i, 1 \le j \le |x|\}\right| \quad (1 \le i \le |x|)$$

[0027] x의 순위는 접두사표현법(prefix representation)으로 나타낼 수 있다. x의 접두사표현 $\mu(x)$는 다음과 같이 정의된다.

[0028] $$\mu(x)_i = \pi(x[1..i])_i \quad (1 \le i \le |x|)$$

[0029] 즉, $\mu(x)$의 i번째 원소는 $x[1..i]$에서 x_i의 자연순위를 의미한다. 예를 들어, $x = (20,50,40,10,30,60)$이면 x의 자연표현과 접두사 표현은 각각 $\pi(x) = (2,5,4,1,3,6)$, $\mu(x) = (1,2,2,1,3,6)$이다.

[0030] 두 문자열 x와 y가 $|x| = |y|$이고, 모든 1≤i, j≤$|x|$에 대해 $x_i \le x_j \Leftrightarrow y_i \le y_j$를 만족할 경우, 즉, 문자들 간의 상대적 순위가 동일할 경우 두 문자 x와 y는 순위동형이라 하고 x y라 표기한다. 예를 들어, 두 문자열 $x = (10,2,15,7)$과 $y = (8,3,11,4)$는 길이가 같고, 순위가 (3, 1, 4, 2)로 동일하므로 순위동형이다.

[0031] 길이가 1 이상인 두 문자열 x와 w에 대해, w와 순위동형인 x의 부분문자열이 존재하지 않고, $w[1..|w|]$, $w[2..|w|]$와 순위동형인 x의 부분문자열이 존재하면, w의 자연순위 $\pi(w)$를 x의 극소불포함순위문자열로 정의한다. 예를 들어, 도 1과 같이 $x = (20,50,40,10,30,60)$, $w = (2,10,4,7)$라고 하자. $w[1..3] \approx x[1..3]$이고, $w[2..4] \approx x[3..5]$이지만 w와 순위동형인 x의 부분문자열은 존재하지 않는다. 따라서, $\pi(w) = (1,4,2,3)$은 x의 극소불포함순위문자열이다.

[0032] 극소불포함순위문자열문제는 문자열 $x \in \Sigma^*$가 주어졌을 때, X의 모든 극소불포함순위문자열의 집합 $MAOP_X$를

찾는 문제이다. 위의 예시와 같이 $x=(20,50,40,10,30,60)$ 라면, $MAOP_X$={(2, 1, 3), (2, 3, 1), (1, 2, 3, 4), (1, 2, 4, 3), (1, 4, 2, 3), (1, 4, 3, 2), (4, 1, 2, 3), (4, 1, 3, 2), (4, 3, 2, 1), (3, 5, 4, 1, 2), (5, 3, 1, 2, 4)}이다.

[0033] 도 2는 일 실시예에 있어서, 문자열 탐색 시스템의 구성을 설명하기 위한 블록도이고, 도 3은 일 실시예에 있어서, 극소불포함순위문자열 탐색 방법을 설명하기 위한 흐름도이다.

[0034] 문자열 탐색 시스템(100)의 프로세서는 그래프 생성부(210) 및 문자열 탐색부(220)를 포함할 수 있다. 이러한 프로세서의 구성요소들은 문자열 탐색 시스템에 저장된 프로그램 코드가 제공하는 제어 명령에 따라 프로세서에 의해 수행되는 서로 다른 기능들(different functions)의 표현들일 수 있다. 프로세서 및 프로세서의 구성요소들은 도 3의 극소불포함순위문자열 탐색 방법이 포함하는 단계들(310 내지 320)을 수행하는 문자열 탐색 시스템을 제어할 수 있다. 이때, 프로세서 및 프로세서의 구성요소들은 메모리가 포함하는 운영체제의 코드와 적어도 하나의 프로그램의 코드에 따른 명령(instruction)을 실행하도록 구현될 수 있다.

[0035] 프로세서는 극소불포함순위문자열 탐색 방법을 위한 프로그램의 파일에 저장된 프로그램 코드를 메모리에 로딩할 수 있다. 예를 들면, 문자열 탐색 시스템에서 프로그램이 실행되면, 프로세서는 운영체제의 제어에 따라 프로그램의 파일로부터 프로그램 코드를 메모리에 로딩하도록 문자열 탐색 시스템을 제어할 수 있다. 이때 프로세서는 그래프 생성부(210) 및 문자열 탐색부(220) 각각에 메모리에 로딩된 프로그램 코드 중 대응하는 부분의 명령을 실행하여 이후 단계들(310 내지 320)을 실행하기 위한 프로세서의 서로 다른 기능적 표현들일 수 있다.

[0036] 단계(310)에서 그래프 생성부(210)는 문자열의 순위그래프를 생성할 수 있다. 그래프 생성부(210)는 순열의 마지막 원소로 순위를 추가한 순열을 획득하기 위한 제1함수를 정의할 수 있다. 그래프 생성부(210)는 순열의 첫 번째 원소를 삭제한 순열을 획득하기 위한 제2함수를 정의할 수 있다. 그래프 생성부(210)는 제2 함수에 대한 제1 함수의 순열과, 제 1함수에 대한 2 함수와 제3함수의 값을 이용한 순열이 동치인 순열을 획득하기 위한 제3 함수를 정의할 수 있다. 그래프 생성부(210)는 문자열 X에 대해, 문자열 X의 모든 부분문자열의 서로 다른 자연순위 집합을 N_X라고 할 때, 순열과 대응되는 정점과 정점에 대응되는 순열을 구성하고, 구성된 정점에 제1 함수, 제2 함수 및 제3 함수를 이용하여 생성된 간선을 연결하여 문자열의 순위그래프를 생성할 수 있다.

[0037] 단계(320)에서 문자열 탐색부(220)는 생성된 문자열의 순위그래프를 이용하여 문자열의 모든 극소불포함순위문자열의 집합을 탐색할 수 있다. 문자열 탐색부(220)는 생성된 문자열의 순위그래프에서, 정점에서 나가는 간선들에 대해 간선의 정수 문자들의 집합과, 루트 정점에서 정점까지 간선들로 연결된 경로의 길이를 이용하여 문자열의 모든 극소불포함순위문자열의 집합을 탐색할 수 있다.

[0038] 도 4 및 도 5는 일 실시예에 있어서, 순위그래프를 설명하기 위한 예이다.

[0039] 문자열 X의 극소불포함순위문자열의 집합 $MAOP_X$를 찾기 위한 순위그래프 G_X 를 생성하는 동작에 대하여 설명하기로 한다.

[0040] 정의 1. g 함수(g : $S_n \times [n+1] \rightarrow S_{n+1}$)

[0041] 모든 $\sigma \in S_n$, $a \in [n+1]$에 대해, $g(\sigma, a) = (\tau_1, \tau_2, ..., \tau_n, a)$는 다음과 같다.

[0042]
$$\tau_i = \begin{cases} \sigma_i & \text{if } \sigma_i < a \\ \sigma_i + 1 & \text{otherwise} \end{cases}$$

[0043] 즉, $g(\sigma, a)$는 순열 σ의 마지막 원소로 순위 a를 추가한 순열을 의미한다. 예를 들어, $\sigma = (2,1,3)$과 $a = 2$에 대해 $g(\sigma, a) = (3,1,4,2)$이다.

[0044] 정의 2. f 함수(f : $S_n \rightarrow S_{n-1}$)

[0045] 모든 $\sigma \in S_n$에 대해, $f(\sigma)=(v_1,\ldots,v_{n-1})$은 다음과 같다.

[0046]
$$v_i=\begin{cases}\sigma_{i+1} & \text{if } \sigma_{i+1}<\sigma_1\\ \sigma_{i+1}-1 & \text{otherwise}\end{cases}$$

[0047] $f(\sigma)$는 순열 σ의 첫 번째 원소를 삭제한 순열을 의미한다. 예를 들어, $\sigma=(3,1,4,2)$에 대해, $f(\sigma)=(1,3,2)$이다.

[0048] 정의 3. h 함수(h : $S_n\times[n+1]\rightarrow[n]$)

[0049] 모든 $\sigma \in S_n$, $a \in [n+1]$, $b \in [n]$에 대해, $h(\sigma,a)=b$는 $f(g(\sigma,a))=g(f(\sigma),b)$와 동치이다.

[0050] 증명. $\tau=g(\sigma,a)$, $v=f(\tau)$라고 하자. 즉, $a=\tau_{n+1}$, $b=v_n$이다. (1) $\sigma_1<a$의 경우, g에 의해 $\tau_1=\sigma_1$이다. 즉, $\sigma_1<a$는 $\tau_1<\tau_{n+1}$임을 의미한다. 따라서, f의 정의에 의해 $b=v_n=\tau_{n+1}-1=a-1$이다. (2) $\sigma_1\geq a$의 경우도 유사하게 구할 수 있다. 즉, g에 의해 $\tau_1=\sigma_1+1>a$이고, 이는, $\tau_{n+1}<\tau_1$이므로 $b=v_n=\tau_{n+1}=a$이다. 이는 순열 $\sigma=(\sigma_1,\sigma_2,\ldots,\sigma_n)$와 순위 a를 연결한 순열 $g(\sigma,a)$를 $(\sigma'_1,\sigma'_2\ldots,\sigma'_n,a)$, $g(\sigma,a)$의 첫 문자를 제거한 순열 $f(g(\sigma,a))$를 $(\sigma''_2,\ldots,\sigma''_n,a')$라 한다면, $g(\sigma,a)$와 $f(g(\sigma,a))$를 계산하지 않고, 순위 a'를 구할 수 있음을 의미한다. 예를 들어, $\sigma=(2,1,3)$, $a=2$일 때, $f(g(\sigma,a))$의 마지막 원소가 2임을 h를 이용해 계산할 수 있다.

[0051] 문자열 X에 대해, X의 모든 부분문자열의 서로 다른 자연순위 집합을 N_X라 하자. X의 극소불포함순위문자열 집합 $MAOP_X$는 다음과 같이 정의된다.

[0052]
$$MAOP_X=\left\{\tau=g(\sigma,a)\,|\,\sigma\in N_X \text{ and } f(g(\sigma,a))\in N_X \text{ and } \tau\notin N_X\right\}$$

[0053] X에 대한 순위그래프 $G_X=(V,E_g\cup E_f)$는 다음과 같이 정의되는 유향그래프이다.

[0054] 정점 집합 V는 N_X와 일대일 대응인 정점들로 구성된다. $\sigma\in N_X$와 대응되는 정점은 $v(\sigma)$로, $u\in V$에 대응되는 순열은 $\sigma(u)$로 표기하여 사용한다.

[0055]
$$E_g=\left\{<u,a,v>\,|\,u,v\in V \text{ and } \exists a \text{ s.t. } g(\sigma(u),a)=\sigma(v)\right\}$$

[0056]
$$E_f=\{<u,v>\,|\,u,v\in V \text{ and } f(\sigma(u))=\sigma(v)\}$$

[0057] $u\in V$에 대해, $A_u=\{a\,|\,<u,a,v>\in E_g\}$이다. 즉, 집합 A_u는 u에서 나가는 간선 $e\in E_g$에 대해, e의 정수 문자들의 집합을 의미한다.

[0058] d_u는 $v(\epsilon)$에서 u까지 $e\in E_g$들로 연결된 경로의 길이를 의미한다.

[0059] 관찰 1. 그래프(V, E_g)는 루트 $v(\epsilon)$인 트라이(trie)이다.

[0060] 관찰 2. $u\in V$에 대해, $v(\epsilon)$에서 u까지 E_g의 원소로 이루어진 경로의 정수 문자를 모두 연결한 문자열은 $\sigma(u)$

의 접두사표현 $\mu(\sigma(u))$이다. 이는 μ와 g의 정의에 의해 자명하다.

[0061] G_X는 $u \in V$에 대응하는 자연순위 $\sigma(u)$를 실제로 저장하지 않는다. 도 4를 참고하면, X=(20, 50, 40, 10, 30, 60)의 순위그래프를 나타낸 것으로, 즉, 도 4와 같이 실제 자연순위를 저장하지 않는다. 대신, A_u, d_u를 이용하여 $MAOP_X$를 계산한다. G_X의 각 정점 u에 대해 세 집합 A_u, $A_{v(f(\sigma(u)))}$, $A_u' = [d_u + 1] - A_u$를 고려하기로 한다. $a' \in A_u'$은 u에서 나가는 간선 $e = < u, a', v >$가 E_g에 존재하지 않음을 의미한다. 예를 들어, 도 5의 정점 u에 대해, $A_u = \{1\}$, $A_{v(f(\sigma(u)))} = \{1, 2\}$, $A'_u = \{2, 3, 4\}$이다.

[0062] 정리 1. 각 원소 $a' \in A'$에 대해, $h(\sigma(u), a') \in A_{v(f(\sigma(u)))}$라면 $g(u, a') \in MAOP_X$이다. $h(\sigma(u), a') \notin A_{v(f(\sigma(u)))}$라면 $g(u, a') \notin MAOP_X$이다.

[0063] 증명. $\sigma = \sigma(u)$, $\tau = f(\sigma)$라 한다. 문제의 가정에 의해 $\sigma, \tau \in N_X$이고, $g(\sigma, a') \in N_X$이다. $b = h(\sigma(u), a')$이라 한다. b가 $A_{v(f(\sigma(u)))}$에 존재하는 것(즉, 그래프 상에 존재하는 것)은, $g(f(\sigma), b) \in N_X$임을 의미한다. h의 정의에 의해 $g(f(\sigma), b) = f(g(\sigma, a'))$이고, 이는 $f(g(\sigma, a')) \in N_X$를 의미한다.

[0064] 따라서, $g(u, a') \in MAOP_X$이다. 마찬가지 논리로 $b \notin A_{v(f(\sigma(u)))}$에 대해서도 생각할 수 있다. 이 경우, $f(g(\sigma, a')) \notin N_X$이므로, $g(u, a') \notin MAOP_X$이다.

[0065] 도 5의 예시에서, $\sigma(u) = (1, 3, 2)$이므로, $A_u' = \{2, 3, 4\}$의 각 원소에 대한 $h(\sigma(u), a')$은 각각 $h(\sigma(u), 2) = 1$, $h(\sigma(u), 3) = 2$, $h(\sigma(u), 4) = 3$이다. $A_{v(f(\sigma(u)))} = \{1, 2\}$이므로, $g(\sigma(u), 2) = (1, 4, 3, 2)$, $g(\sigma(u), 3) = (1, 4, 2, 3)$은 X의 극소불포함순위문자열이 된다. 도 6을 참고하면, X=(20, 50, 40, 10, 30, 60)의 극소불포함순위문자열 집합의 예이다. $g(\sigma(u), 1)$은 N_X에 존재하고, $g(\sigma(u), 4)$는 $f(g(\sigma(u), 4))$가 N_X에 존재하지 않으므로 극소불포함순위문자열이 아니다. 이때, h를 계산하기 위해 $\sigma(u)$전체가 아닌 $\sigma(u)_1$만 필요하므로, 각 정점마다 이를 저장하여 계산할 수 있다. G_X의 모든 정점 u를 탐색하며, 위에서 서술한 두 집합 $A_{v(f(\sigma(u)))}$, A_u'과 $\sigma(u)_1$을 이용하여 X의 모든 극소불포함순위문자열을 찾을 수 있다. $g(\sigma, a)$가 X의 극소불포함순위문자열이라면, $v(\sigma) \in V$이므로, 모든 정점에 대해 탐색이 종료되면 X의 모든 극소불포함순위문자열, 즉 집합 $MAOP_X$를 찾을 수 있다.

[0066] 이상에서 설명된 장치는 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서

(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0067] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치에 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

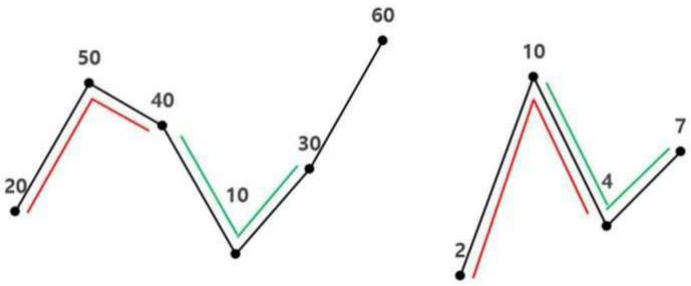
[0068] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다.

[0069] 이상과 같이 실시예들이 비록 한정된 실시예와 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

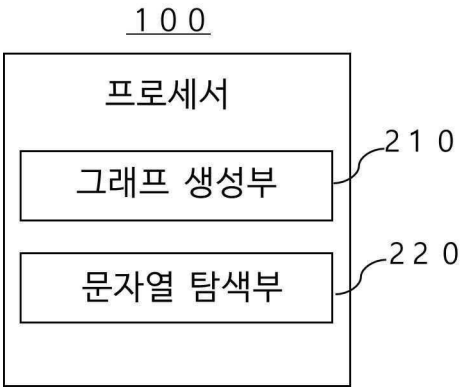
[0070] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

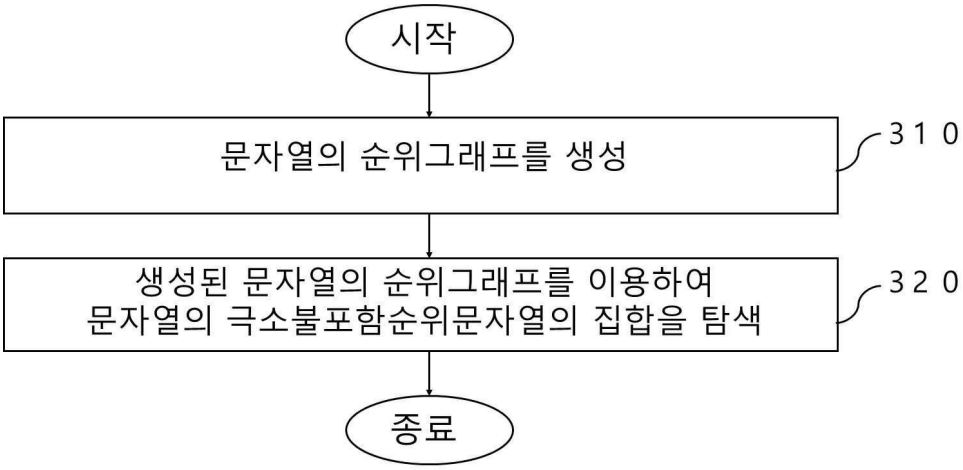
도면1



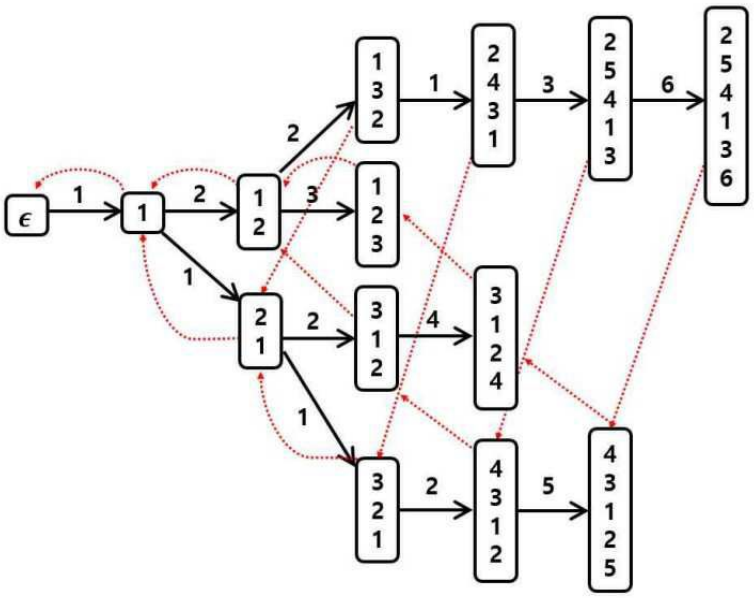
도면2


100
프로세서
그래프 생성부
210
문자열 탐색부
220

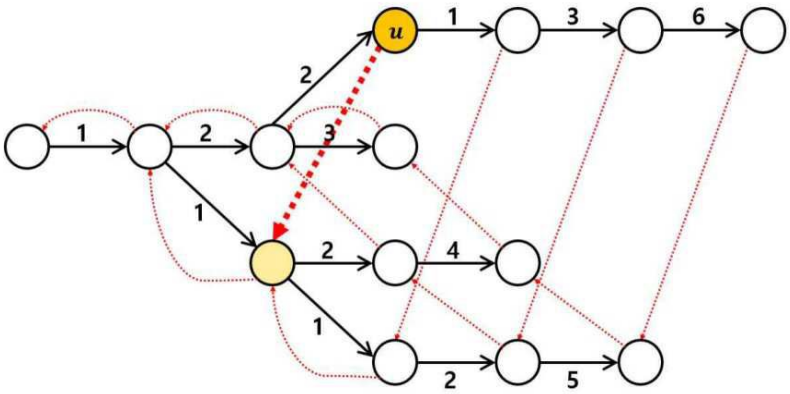
도면3


시작
문자열의 순위그래프를 생성
310
생성된 문자열의 순위그래프를 이용하여
문자열의 극소불포함순위문자열의 집합을 탐색
320
종료

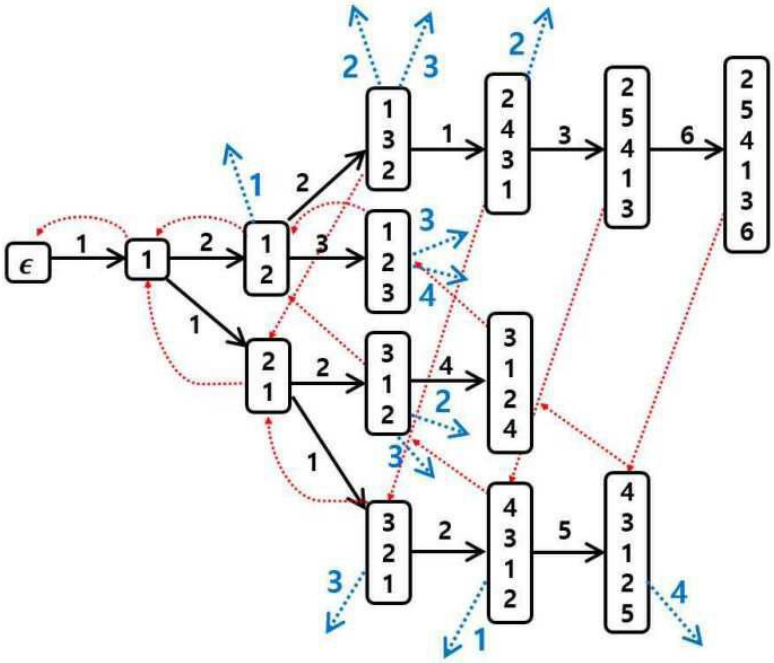
도면4


ϵ
1
2
12
3
123
1
21
2
312
4
3124
1
321
2
4312
5
43125
132
1
2431
3
25413
6
2541336

도면5


u
1
2
3
4
5
6

도면6


ϵ
1
12
21
132
123
312
321
2431
31124
43112
25413
43125
2541136