

알고리즘

문자열

이영석

yslee@cs-cnu.org

오늘 학습 내용

- 정규식
- Anagram
- Palindrome
- ...

예제

- <https://tech.kakao.com/2017/09/27/kakao-blind-recruitment-round-1/>

입출력 예제

2. 다트 게임(난이도: 하)

카카오톡에 뜬 네 번째 별! 심심할 땐? 카카오톡 게임별~



카카오톡 게임별의 하반기 신규 서비스로 다트 게임을 출시하기로 했다. 다트 게임은 다트판에 다트를 세 차례 던져 그 점수의 합계로 실력을 겨루는 게임으로, 모두가 간단히 즐길 수 있다.

갓 입사한 무지는 코딩 실력을 인정받아 게임의 핵심 부분인 점수 계산 로직을 맡게 되었다. 다트 게임의 점수 계산 로직은 아래와 같다.

예제	dartResult	answer	설명
1	1S2D*3T	37	$1^1 * 2 + 2^2 * 2 + 3^3$
2	1D2S#10S	9	$1^2 + 2^1 * (-1) + 10^1$
3	1D2S0T	3	$1^2 + 2^1 + 0^3$
4	1S*2T*3S	23	$1^1 * 2 * 2 + 2^3 * 2 + 3^1$
5	1D#2S*3S	5	$1^2 * (-1) * 2 + 2^1 * 2 + 3^1$
6	1T2D3D#	-4	$1^3 + 2^2 + 3^2 * (-1)$
7	1D2S3T*	59	$1^2 + 2^1 * 2 + 3^3 * 2$

정규식(Regular Expression)

정규식	문자열	Match 여부	설명
<code>ca*t</code>	ct	Yes	"a"가 0번 반복되어 매치
<code>ca*t</code>	cat	Yes	"a"가 0번 이상 반복되어 매치 (1번 반복)
<code>ca*t</code>	caaat	Yes	"a"가 0번 이상 반복되어 매치 (3번 반복)

정규식	문자열	Match 여부	설명
<code>ca+t</code>	ct	No	"a"가 0번 반복되어 매치되지 않음
<code>ca+t</code>	cat	Yes	"a"가 1번 이상 반복되어 매치 (1번 반복)
<code>ca+t</code>	caaat	Yes	"a"가 1번 이상 반복되어 매치 (3번 반복)

정규식	문자열	Match 여부	설명
<code>ab?c</code>	abc	Yes	"b"가 1번 사용되어 매치
<code>ab?c</code>	ac	Yes	"b"가 0번 사용되어 매치

```
>>> import re
>>> p = re.compile('ab*')
```

Method	목적
<code>match()</code>	문자열의 처음부터 정규식과 매치되는지 조사한다.
<code>search()</code>	문자열 전체를 검색하여 정규식과 매치되는지 조사한다.
<code>findall()</code>	정규식과 매치되는 모든 문자열(substring)을 리스트로 돌려준다.
<code>finditer()</code>	정규식과 매치되는 모든 문자열(substring)을 반복 가능한 객체로 돌려준다.

문제1: Anagram 찾기

- 주어진 2개 문자열의 알파벳 수가 같은가?
 - "abcd", "dcba": OK
 - "heart", "earth"
 - "python", "typhon"

Anagram 해결방법1

- 문자열을 정렬해서 비교하자!
 - "abcd", "dcba": "abcd", "abcd"
- 복잡도는?

Anagram 해결방법2

- 알파벳을 카운트하자
 - "abca", "acba":
 - a: 2, b:1, c: 1
- 복잡도는?

문제2: Palindrome

- 앞으로 뒤로 똑같은 문자열
- 해결방법
 - String 앞에서 뒤에서 비교?
 - 뒤집기?
 - 재귀?
- 복잡도?
 - $O(n)$

문제3: Longest Palindromic Substring(LPS)

- 주어진 문자열에서 가장 긴 Palindrome 부분 문자열을 찾으시오 (길이를 출력하시오.)
- 예제
 - babad -> bab or aba (3)
 - cbabd -> bb (2)
 - banana -> anana (5)

LPS 해결 방법1

- 모든 문자열을 찾고, palindrome 검사?
 - Brute-force, exhaustive search
 - 복잡도는?
 - 길이 n 인 문자열에서 만들 수 있는 문자열은 몇 개? 2^n
 - OK
 - 문자열의 길이가 짧을 때에는 가능
 - 모든 문자열 찾는 복잡도 낮추기
 - $O(n^2)$
 - 팰린드롬 검사 복잡도
 - $O(n)$

LPS 해결 방법2

- 길이 n 인 문자열이 palindrome
 - $n-2$ 문자열이 palindrome 이고 앞, 뒤 문자가 똑같은 때!
 - $b \rightarrow aba \rightarrow cabac \rightarrow dcabacd$

LPS 해결 방법2

- $s = \text{"abaab"}$
- Substring 모든 케이스

i,j	0	1	2	3	4
0	$s[0:0] = a$	$s[0:1] = ab$	$s[0:2] = aba$	$s[0:3] = abaa$	$s[0:4] = abaab$
1		$s[1:1] = b$	$s[1:2] = ba$	$s[1:3] = baa$	$s[1:4] = baab$
2			$s[2:2] = a$	$s[2:3] = aa$	$s[2:4] = aab$
3				$s[3:3] = a$	$s[3:4] = ab$
4					$s[4:4] = b$

i,j	0	1	2	3	4
0	s[0:0] = a	s[0:1] = ab	s[0:2] = aba	s[0:3] = abaa	s[0:4] = abaab
1		s[1:1] = b	s[1:2] = ba	s[1:3] = baa	s[1:4] = baab
2			s[2:2] = a	s[2:3] = aa	s[2:4] = aab
3				s[3:3] = a	s[3:4] = ab
4					s[4:4] = b

i,j	0	1	2	3	4
0	s[0:0] = a T	s[0:1] = ab F	s[0:2] = aba T	s[0:3] = abaa F	s[0:4] = abaab F
1		s[1:1] = b T	s[1:2] = ba F	s[1:3] = baa F	s[1:4] = baab T
2			s[2:2] = a T	s[2:3] = aa T	s[2:4] = aab F
3				s[3:3] = a T	s[3:4] = ab F
4					s[4:4] = b T

길이 1 substring 검사

i,j	0	1	2	3	4
0	$s[0:0] = a$ T				
1		$s[1:1] = b$ T			
2			$s[2:2] = a$ T		
3				$s[3:3] = a$ T	
4					$s[4:4] = b$ T

길이 2 substring 검사

i,j	0	1	2	3	4
0	s[0:0] = a T	s[0:1] = ab F			
1		s[1:1] = b T	s[1:2] = ba F		
2			s[2:2] = a T	s[2:3] = aa T	
3				s[3:3] = a T	s[3:4] = ab F
4					s[4:4] = b T

길이 3 substring 검사

i,j	0	1	2	3	4
0	s[0:0] = a T	s[0:1] = ab F	s[0:2] = aba T		
1		s[1:1] = b T	s[1:2] = ba F	s[1:3] = baa F	
2			s[2:2] = a T	s[2:3] = aa T	s[2:4] = aab F
3				s[3:3] = a T	s[3:4] = ab F
4					s[4:4] = b T

길이 4 substring 검사

i,j	0	1	2	3	4
0	s[0:0] = a T	s[0:1] = ab F	s[0:2] = aba T	s[0:3] = abaa F	
1		s[1:1] = b T	s[1:2] = ba F	s[1:3] = baa F	s[1:4] = baab T
2			s[2:2] = a T	s[2:3] = aa T	s[2:4] = aab F
3				s[3:3] = a T	s[3:4] = ab F
4					s[4:4] = b T

길이 5 substring 검사

i,j	0	1	2	3	4
0	s[0:0] = a T	s[0:1] = ab F	s[0:2] = aba T	s[0:3] = abaa F	s[0:4] = abaab F
1		s[1:1] = b T	s[1:2] = ba F	s[1:3] = baa F	s[1:4] = baab T
2			s[2:2] = a T	s[2:3] = aa T	s[2:4] = aab F
3				s[3:3] = a T	s[3:4] = ab F
4					s[4:4] = b T

정리

- 문자열 기본 연산 익히기
 - 리스트 <-> 문자열 변환
 - 정규식
 - 기본문제