김일성종합대학학보
(자연과학)
주체105(2016)년 제62권 제12호

JOURNAL OF **KIM IL SUNG** UNIVERSITY
(NATURAL SCIENCE)
Vol. 62 No. 12 JUCHE105(2016).

정규표현식과 패턴정합을 리용한 프로그람 평가의 한가지 방법

김순실, 김예화, 리청한

경애하는 **김정은**동지께서는 다음과 같이 말씀하시였다.

《정보기술, 나노기술, 생물공학을 비롯한 핵심기초기술과 새 재료기술, 새 에네르기기술, 우주기술, 핵기술과 같은 중심적이고 견인력이 강한 과학기술분야를 주타격방향으로 정하고 힘을 집중하여야 합니다.》(《조선로동당 제7차대회에서 한 중앙위원회사업총화보고》 단행본 39페지)

현재 프로그람문제의 자동채점에서 일반적으로 리용되고있는 방법은 시험자의 프로그람의 실행결과를 조사하여 실행결과가 옳으면 5점이고 틀리면 0점으로 채점하는 방법이다. 이러한 방법은 사람에 의한 채점과 배치되고 불공평하며 심지어 학생에게 투기적인 기회를 주게 된다. 일부 점수채점체계에서는 학생의 결과코드를 입력하고 열쇠단어의 비교를 진행하는 구조를 갖추고있지만 간단한 단어의 비교를 진행하는데 머무르고있으며 게다가 열쇠단어를 어떻게 선택하겠는가 하는 문제가 존재하고있다.[2, 3]

이러한 부족점을 극복하기 위해 선행연구[1]에서는 학생들이 편집한 원천코드에 대하여 검사를 진행하는 방법이 제안되였다. 또한 학생의 원천프로그람과 미리 작성된 가능한 전체를 대표할수 있는 표준답안과의 비교를 진행하고 비교결과에 따라 코드믿음도라고 부르는 자료를 얻고 이 자료를 점수평가표준양식에 넣는다. 코드믿음도가 높을수록 높은 점수로 평가한다. 이러한 방법은 이전의 방법들에 비해 많은 개선을 가져왔지만 이 방법역시 본문에 대한 간단한 비교로서 효과는 물론 만족스럽지 못하다. 그러므로 실현방법에 있어서 연구를 심화시켜야 한다.

론문에서는 정규표현식을 리용하여 학습자들이 작성한 프로그람들을 표준화하고 무게벡토르에 의한 패턴정합방법으로 프로그람을 자동채점하기 위한 한가지 방법을 론의한다.

1. 정규표현식을 리용한 프로그람의 표준화

프로그람작성자들은 서로 다른 변수와 명령문들을 리용하여 프로그람을 작성하므로 하나의 문제에 대한 프로그람은 각이하게 주어지게 된다.

이러한 프로그람에 대하여 정답과의 정합을 진행하려면 각이한 변수와 명령문들로 작성된 프로그람을 정규표현식을 리용하여 표준화하는것이 중요하다.

정규표현식에 리용되는 기호들은 다음과 같다.

우선 문자들과 략자들은 다음과 같다.

c, \c, \a, \f, \n, \r, \t, \v, \xhhh, \0ooo, \d, \D, \s, \S, \w, \W

다음 문자모임들은 다음과 같이 표시한다.

^(제외문자), -(범위문자)

한편 표현식은 자동적으로 {1, 1}에 의하여 제한된다. 이때 리용되는 제한자들은 다음과 같다.

E?, E^+, E^*, E{n}, E{n,}, E{, m}, E{n, m}, ^, $, \b, \B

여기서 매 제한자들의 의미는 다음과 같다.

E?: E의 0번 또는 한번의 출현을 정합한다.

E^+: E의 한번 또는 그 이상의 출현을 정합한다.

E^*: E의 0번 또는 그 이상의 출현을 정합한다.

E{n}: 이것은 표현식을 n번 반복하는것과 같다.

E{n,}: 표현식을 적어도 n번 반복하는것과 같다.

E{, m}: 기껏해서 표현식을 m번 반복하는것과 같다.

E{n, m}: 표현식을 적어도 n, 기껏해서 m번 반복한다.

^: 문자렬의 시작을 표현한다.

$: 문자렬의 끝을 표현한다.

\b: 단어경계를 표현한다.

\B: \b가 false일 때 true이다.

이러한 기호들에 의하여 문법을 서술하는 방법은 다음과 같다.

수자: 0|1|2|…|9 혹은 [0-9]

문자: A|B|C|…|Z|a|b|c|…|z

수값: ＜수자＞＜수값＞｜＜수자＞

문자렬: ＜문자＞｜＜수자＞｜＜문자＞＜문자렬＞｜＜수자＞＜문자렬＞

이름: ＜문자＞＜문자렬＞｜＜문자＞

변수: ＜문자＞＜문자렬＞

산수식: ＜산수식＞+＜행＞｜＜산수식＞=＜행＞｜＜행＞

행: ＜행＞*＜인자＞｜＜행＞／＜인자＞｜＜인자＞

비교: ＝＝｜＞｜＜｜＞＝｜＜＝｜!＝

식: ＜산수식＞＜비교＞＜산수식＞｜＜산수식＞

대입문: ＜변수＞=＜식＞

if문: if ＜식＞ then ＜실행문＞ else ＜실행문＞

｜if ＜식＞ then ＜실행문＞

while문: while ＜식＞＜실행문＞

복합문: {＜실행문＞}

실행문: ＜대입문＞｜＜if문＞｜＜while문＞｜＜복합문＞

여기에 기초하여 $\sqrt{\sqrt{\cdots\sqrt{2}}} < 1.065$를 만족시키는 뿌리개수를 구하는 프로그람을 while 명령문을 리용하여 표준화하여보자.

우선 뿌리개수를 구하는 PHP프로그람은 다음과 같다.

＜?php

```
$n=1;
$a=sqrt(2);
while($a＞1.065) {
    $a=sqrt($a);
    $n++;    }
echo $n-1;
?＞
```

이 프로그람을 우에서 정의한 정규표현식의 기호들과 프로그람예약어들을 리용하여 표준화하면 다음과 같다.

```
<?php | ＜? | ＜% | ＜%script ⇒ <?php
$([A-z]+[0-9]*\)=1; ⇒ $a=1;
$([A-z]+[0-9]*\)=sqrt(2); ⇒ $b=sqrt(2);
while($([A-z]+[0-9]*\)＞1.065){ ⇒ while($b＞1.065){
$([A-z]+[0-9]*\)=sqrt($([A-z]+[0-9]*\)); ⇒ $b=sqrt($b);
$([A-z]+[0-9]*\)(++; | = $([A-z]+[0-9]*\)+1;)} ⇒ $a++;}
(echo | print) $([A-z]+[0-9]*\)-1; ⇒ echo $a-1;
?＞ | %＞ | /script＞ ⇒ ?＞
```

결과 표준화된 정답프로그람은 다음과 같다.

```
<?php
$a=1;
$b=sqrt(2);
while($b＞1.065){
$b=sqrt($b);
$a++;}
echo $a-1;
?＞
```

2. 패턴정합에 의한 프로그람평가방법

프로그람언어는 련이은 문자 또는 기호, 명령문들의 렬이며 이러한것들이 서로 결합되여 프로그람의 행을 형성한다. 그러므로 패턴정합방법으로 프로그람을 평가하기 위해서는 먼저 띄여쓰기를 하지 않고있는 프로그람언어의 형태부해석을 해야 하는데 우리는 앞최대길이일치법으로 진행하였다.

앞최대길이일치해석알고리듬은 다음과 같다.

① 문장 S의 문자렬의 왼쪽 끝(l_1=0)에서부터 해석을 시작한다.

② 문자 l_1+1로부터 l_2(l_1<l_2)까지의 (l_2–l_1)문자가 형태부를 구성하는가를 탐색하고 사전에 올림말이 있으면 등록하고 l_2=l_1 +1로부터 순차적으로 l_2←l_1+1로 하여 l_2≤n까지 진행한다.

③ 문자 l_2를 마지막문자로 하는 단어 W_i와 l_1+1로부터 l_2까지의 문자로 구성되는 단어 W_{i+1}에 대하여 W_i와 W_{i+1}이 문법적으로 린접가능한가를 검사하고 가능하지 않으면 W_{i+1}의 등록을 제외한다.

④ ③에서 W_{i+1}의 등록이 취소된 경우에 $l_2 \rightarrow l_1$로 하고 ②로 이행한다.

패턴정합에 의한 프로그람평가방법은 다음과 같다.

① 표준화된 프로그람과 정답자료기지의 프로그람을 한행씩 비교하여 오유행을 찾아낸다.

② 오유행의 개수가 프로그람의 전체 행에서 차지하는 비중을 계산한다.

③ 오유가 검출된 행에 대하여 앞최대길이일치법에 의하여 형태부해석을 진행하고 틀린 부분이 그 행에서 차지하는 비중을 계산한다.

여기로부터 프로그람평가식은 다음과 같이 표시할수 있다.

$$M = 5 - \sum_{i=1}^{m} \frac{1}{N} \times e(s_i) \times 5$$

여기서 N은 프로그람행의 개수이고 m은 오유개수이며 s는 오유가 있는 행이며 $e(s_i)$는 오유크기로서 대응하는 행의 무게벡토르 $p(s)$와 평가벡토르 $q(s)$의 스칼라적으로 표시할수 있다. 즉

$$e(s) = \langle p(s),\ q(s) \rangle .$$

제안한 방법을 다음의 프로그람을 실례로 들어 평가해보자.

```
<?php
 $n=1;
 $a=sqrt(2);
 while($a>1.065){
$a=sqrt($a);
    $n++; }
 echo $n-1;
 ?>
```

평가를 위해 우의 프로그람에서 4행의 >기호를 <로 바꾸어썼다고 하자.

오유가 들어있는 4행에 대하여 형태부해석을 진행하면 다음과 같은 형태부렬

$$W(s) = (\ \text{while},\ (,\ \$a,\ >,\ 1.065,\ )\ )$$

를 얻는다.

이때 정답자료기지의 대응하는 행의 무게벡토르는

$$p(s) = (0.4,\ 0.1,\ 0.1,\ 0.2,\ 0.1,\ 0.1)$$

이고 평가벡토르는

$$q(s) = (0,\ 0,\ 0,\ 1,\ 0,\ 0)$$

이므로 오유행 s에 대한 오유의 크기는

$$e(s) = 0.2$$

로 계산된다.

따라서 이 프로그람에 대한 평가점수는

$$M = 5 - \sum_{i=1}^{m} \frac{1}{N} \times e(s_i) \times 5 = 5 - \frac{1}{8} \times 0.2 \times 5 = 4.875$$

로 된다.

맺 는 말

정규표현식을 리용하여 작성된 프로그람을 표준화하고 패턴정합을 리용하여 표준화된 프로그람을 자동채점하는 한가지 방법을 제안하였다.

참 고 문 헌

[1] **김일성**종합대학학보(자연과학), 58, 12, 24, 주체101(2012).

[2] Pang To Il; Distance Education System by Computer Network, Industrial Publishing House, 144～155, 2007.

[3] Nguyen Tuan Dang et al.; The 2nd International Conference on Information and Computing Science, 21, 2009.

주체105(2016)년 8월 5일 원고접수

A Method of Program Evaluation on using Regular Expression and Pattern Matching

Kim Sun Sil, Kim Ye Hwa and *Ri Chong Han*

We have standardized the programs by using the regular expression.

Also we have proposed a method for evaluating standardized programs with pattern matching by weight vector and verified the effectiveness through experimentation.

Key words: regular expression, pattern matching, evaluation