Зміст

6 ПОЛІЗ, регулярні вирази, і автомати

6.1 Польський інверсний запис для регулярних виразів

Польський інверсний запис (ПОЛІЗ) для регулярних виразів будується на основі початкового регулярного виразу на основі наступних правил:

1. Порядок операндів в початковому виразі і в перетвореному виразі співпадають.

2. Операції в перетвореному виразу йдуть з урахуванням пріоритету безпосередньо за операндами.

Наприклад, ПОЛІЗ для виразу $(a^\star + b)^\star c$ має такий вигляд:

$$a, \star, b, +, \star, c, \cdot \tag{6.1}$$

В цьому прикладі в стандартному записі регулярного виразу бінарна операція конкатенація $\cdot$ природньо опущена, але в ПОЛІЗ потрібно завжди цю операцію явно вказувати.

Важливою характеристикою ПОЛІЗ є відсутність дужок в запису виразу, тобто його можна опрацьовувати лінійно.

6.1.1 Алгоритм

Для перетворення виразу в ПОЛІЗ необхідно з кожною операцією зв'язати деяке число, яке будемо називати "пріоритет" (0 — найвищий пріоритет присвоїмо дужці `'('`). Наведемо псевдокод алгоритму:

```
while lexem <- прочитати поточну лексему:
    if lexem is операнд:
        занести її в поле результату
```

```
    if lexem = '(':
        занести її в стек

    if lexem is код операції:
        while (пріоритет операції на вершині стека >= \
         пріоритет поточної операції):
            елемент з вершини стека перенести в поле результату
        поточну лексему занести в стек

    if lexem = ')':
        while код операції на вершині стеку != '(':
            елемент з вершини стека перенести в поле результату
        дужку '(' зняти з вершини стека

всі елементи із стека перенести в поле результату
```

6.2 Інтерпретація ПОЛІЗ регулярного виразу

Результат інтерпретації ПОЗІЗ — це скінченний автомат M, який розпізнає (сприймає) множину ланцюжків, котрі позначає регулярний вираз.

6.2.1 Алгоритм

Наведемо псевдокод алгоритму:

```
while lexem <- прочитати поточну лексему:
    if lexem is операнд ai:
        M: L(M) = {ai\}

    if lexem = '+':
        M1, M2 <- автомати з вершини стеку
        M: L(M) = L(M1) \cup L(M2)

    if lexem = '\times':
        M1, M2 <- автомати з вершини стеку
        M: L(M) = L(M2) \times L(M1)

    if lexem = '\star':
        M1 <- автомат з вершини стеку
        M: L(M)= L(M1)^\star
```

```
    M занести в стек

вершину стека перенести в поле результату
```

Якщо досягли кінця регулярного виразу, то на вершині стека знаходиться автомат M, який розпізнає множину слів (ланцюжків), які позначає регулярний вираз.

6.3 Контрольні запитання

1. Що таке ПОЛІЗ?
2. Чи можна в ПОЛІЗ опускати операції які природнім чином опускаються у класичному записі?
3. Яка основна характеристика ПОЛІЗ і яку обчислювальну перевагу вона пропонує?
4. Сформулюйте алгоритм перетворення регулярного виразу у ПОЛІЗ та оцініть його складність.
5. Що є результатом інтерпретації ПОЛІЗ регулярного виразу?
6. Сформулюйте алгоритм інтерпретації ПОЛІЗ регулярного виразу.
7. Оцініть складність попереднього алгоритму через складності операцій побудови автоматів.
8. Для регулярного виразу $(a^\star + b)^\star \cdot c$ побудуйте скінчений автомат, який розпізнає множину ланцюжків, що позначаються цим виразом.