

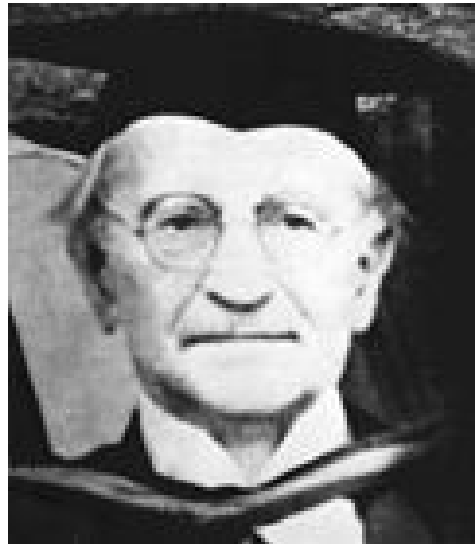
# Ký pháp Ba lan

Bởi:

Wiki Pedia

Ký pháp Ba lan (tiếng Anh: Polish notation), còn gọi là ký pháp tiền tố (tiếng Anh: prefix notation), là một cách viết một biểu thức đại số rất thuận lợi cho việc thực hiện các phép toán. Đặc điểm cơ bản của cách viết này là không cần dùng đến các dấu ngoặc và luôn thực hiện từ trái sang phải.

## Tác giả



*Jan Łukasiewicz*

Kí pháp Ba Lan do nhà logic toán Jan Łukasiewicz đề xuất khoảng năm 1920. Jan Łukasiewicz là một nhà toán học người Ba Lan. Ông sinh ra ở Lwów, Galicia (nay là Lviv, Ukraina). Lĩnh vực nghiên cứu chính của ông là logic toán.

## Các phương pháp biểu diễn phép toán hai ngôi

Một phép toán hai ngôi trên tập hợp  $X$  là một ánh xạ  $f: X \times X \rightarrow X$  cho  $(a, b) \in X \times X$ . Ánh xạ  $f$  khi đó thường được ký hiệu bởi  $*$ , được gọi là toán tử, các phần tử  $a, b$  được gọi là các hạng tử (còn gọi là toán hạng).

Khi viết biểu thức biểu diễn phép toán đó ta có thể đặt ký hiệu toán tử ở trước (ký pháp tiền tố), sau (ký pháp hậu tố) hoặc giữa (ký pháp trung tố) các toán hạng. Thông thường trong các biểu thức đại và số học, ta viết ký hiệu phép toán giữa hai hạng tử, đó là ký pháp trung tố. Ví dụ:  $a + b$ ,  $a * b$ ,... Khi một biểu thức có nhiều phép toán, ta dùng các cặp dấu ngoặc "(", ")" và thứ tự ưu tiên các phép toán để chỉ rõ thứ tự thực hiện các phép toán. (Các phép toán đều quy về phép toán 2 ngôi.)

Ta cũng có thể viết hai hạng tử trước và ký hiệu toán tử sau. Chẳng hạn:

$a + b$  viết là  $a b +$ ,  $a * b$  viết là  $a b *$

Cũng có thể viết toán tử trước, hai toán hạng sau. Chẳng hạn:

$a + b$  viết là  $+ a b$ ,  $a * b$  viết là  $* a b$

Về lý thuyết, ký pháp tiền tố và ký pháp hậu tố còn có thể được mở rộng cho các phép toán ba ngôi hoặc nhiều hơn mà vẫn không phải dùng tới dấu ngoặc để thể hiện độ ưu tiên các phép toán, tương tự với hàm số đa biến, còn ký pháp trung tố thì không thể. Tuy nhiên, trong thực tế không có nhiều phép toán đa ngôi và ký pháp trung tố vẫn được dùng rộng rãi vì thói quen. Ví dụ:  $+ a b c$  có thể được hiểu là tổng của 3 số  $a$ ,  $b$  và  $c$  trong ký pháp tiền tố. Tương tự,  $f a b c$  có thể được hiểu là hàm  $f$  của 3 biến  $a$ ,  $b$  và  $c$  trong ký pháp tiền tố.

## Cây biểu diễn biểu thức

Dùng cây biểu diễn biểu thức có thể thấy rõ trình tự tính toán biểu thức.

Ví dụ:

$$a * (b + c) - d^5$$

Được thực hiện theo sơ đồ biểu diễn bởi cây nhị phân sau

-

/\

\* ^

/\ /\

$a + d^5$

/\

b c

Ta mô tả quá trình đọc, ghi nhận giá trị và thực hiện phép tính, giống như quá trình duyệt cây biểu thức theo thứ tự giữa như sau:

1. Đọc và ghi nhận giá trị biến a (con trái)
2. Đọc và ghi nhận loại phép toán \* (con trái)
3. Đọc giá trị biến b (con trái)
4. Đọc và ghi nhận loại phép toán + (con phải)
5. Đọc giá trị biến c (con phải) thực hiện phép cộng  $b + c = x$  và ghi nhận kết quả x, thực hiện phép nhân  $a * x = y$  và ghi nhận kết quả y
6. Đọc và ghi nhận phép toán -
7. Đọc giá trị biến d
8. Đọc và ghi nhận phép toán ^
9. Đọc giá trị 5 và thực hiện phép lũy thừa  $z = d ^ 5$ , thực hiện phép trừ  $k = y - z$

Để đơn giản ta giả sử các phép toán đều là hai ngôi. Khi đó cây biểu thức là cây nhị phân đầy đủ. Quy tắc thực hiện phép toán trên cây như sau:

Hoặc duyệt cây theo thứ tự giữa mỗi lần duyệt con phải nếu đã tính được giá trị tại con đó thì thực hiện phép tính quy định bởi toán tử ghi tại đỉnh cha.

Viết tuần tự các đỉnh duyệt theo thứ tự giữa thì ta có dãy

--

/\\/\

\* ^ + ^

/\\/\

a + d 5 \* c d 5

/\\/\

b c a b

a \* b + c - 4 \* d ^ 5 (1)

Theo cách này, mỗi khi duyệt một đỉnh

1. Nếu là lá bên trái (biến hoặc hằng) thì ghi nhận giá trị của biến
2. Nếu là toán tử thì lưu dạng của toán tử

3. Nếu là con phải và lá thực hiện phép tính theo toán tử đã lưu ở đỉnh cha giữa các giá trị đã lưu tại con phải và giá trị mới đọc tại con trái, ghi kết quả vào đỉnh cha.

Hoặc duyệt cây biểu thức trên theo thứ tự sau: Mỗi đỉnh được duyệt sau khi cả hai đỉnh con đã được duyệt. Khi đó mỗi khi duyệt một đỉnh trong ta thực hiện toán tử ghi tại đỉnh này với giá trị của hai con của đỉnh ấy.

Thứ tự duyệt các đỉnh khi đó sẽ là:

$$a \ b \ c \ + \ * \ d \ 5 \ ^ \ - \ (2)$$

Khi duyệt theo thứ tự sau, việc thực hiện phép tính tiến hành theo quy tắc:

Mỗi khi thăm một đỉnh thì:

- Nếu là lá thì ghi nhận giá trị của biến
- Nếu là đỉnh trong thì thực hiện toán tử ghi ở đỉnh này vào các giá trị ghi ở hai đỉnh con, ghi kết quả vào chính đỉnh này.

Với dãy (1) nếu không dùng đến dấu ngoặc, có thể có hai cây biểu thức khác nhau cho cùng một dãy đỉnh khi duyệt thứ tự giữa. Còn với dãy (2) cùng với lưu ý rằng các biến và hằng luôn được biểu diễn bằng các lá, các toán tử luôn biểu diễn bởi các nút trong, từ mỗi dãy dạng (2) ta luôn dựng lại được cây biểu thức duy nhất theo giải thuật sau: Khi độ dài dãy lớn hơn 1, duyệt từ bên trái sang, nếu gặp một phần tử là toán tử, thì lấy hai phần tử đứng trước nó ra khỏi dãy chuyển thành hai con của phần tử ấy (theo đúng thứ tự). Vì thế biểu thức viết bởi dãy (2) là hoàn toàn xác định.

### **Tính giá trị của biểu thức viết dưới dạng phép toán sau**

$$(x+y)*z$$

### **Chuyển đổi từ biểu thức bình thường sang kí pháp Ba Lan**

Việc tính giá trị một biểu thức viết dưới dạng phép toán sau rất thuận tiện như trên, tuy nhiên, theo thói quen thông thường, việc nhập biểu thức đó vào lại không dễ, người ta thường nhập vào một công thức dưới dạng thông thường (phép toán giữa) rồi dùng chương trình chuyển đổi nó sang dạng phép toán sau. Chúng ta hãy xét biểu thức trong ví dụ trên

$$Q=a*(b+c)-d^5$$

Kí hiệu biểu thức ghi dưới dạng phép toán sau là P. Trong quá trình chuyển đổi ta dùng một stack S để lưu các phần tử trong P chưa sử dụng đến. Khi đọc từ trái sang phải biểu thức Q la lần lượt có:

1. Đọc và ghi nhận giá trị a, ghi giá trị a vào P. Vậy  $P = "a"$ .
2. Đọc toán tử  $"*"$ . Đưa toán tử này vào stack S:  $S = "*" "$
3. Đọc dấu ngoặc mở  $"(" "$ , đưa dấu ngoặc này vào stack:  $S = "*" "(" "$ .
4. Đọc hạng tử b, đưa b vào P:  $P = "a b"$
5. Đọc toán tử  $"+" "$ , đặt  $"+" "$  vào stack:  $S = "*" "(" "+" "$
6. Đọc hạng tử "c", đưa c vào cuối P:  $P = "a b c"$
7. Đọc dấu ngoặc đóng  $)"$ . Lần lượt lấy các toán tử ở cuối stack ra khỏi stack đặt vào cuối P cho đến khi gặp dấu ngoặc mở  $"(" "$  trong stack thì giải phóng nó:  $S = "*" "$ ;  $P = "a b c +"$
8. Đọc toán tử  $"-" "$ . Cuối stack S có toán tử  $"*"$  có mức ưu tiên lớn hơn toán tử  $"-" "$ , ta lấy toán tử  $"*"$  ra khỏi stack, đặt vào cuối P, đặt toán tử  $"-" "$  vào stack:  $S = "-" "$ ;  $P = "a b c + *"$
9. Đọc hạng tử d, đưa d vào cuối P.  $P = "a b c + * d"$
10. Đọc toán tử  $"^"$ , đưa toán tử  $"^"$  vào cuối stack:  $S = "-" "^"$
11. Đọc hằng số 5, đưa 5 vào cuối P:  $P = "a b c + * d 5"$
12. Đã đọc hết biểu thức Q, lần lượt lấy các phần tử cuối trong stack đặt vào P cho đến hết.  $P = "a b c + * d 5 ^ -"$ .

Thuật toán chuyển từ ký pháp trung tố sang ký pháp tiền tố hoặc hậu tố rất gần với cách xử lý các phép tính trong máy tính bấm tay (hay máy tính bỏ túi). Một biểu thức chỉ gồm các phép toán hai ngôi bất kỳ luôn có thể được tính bằng máy tính bấm tay mà không cần dùng dấu ngoặc. Các phép toán ở trước nếu có độ ưu tiên (ưu tiên bởi toán tử hoặc bởi dấu ngoặc) thấp hơn một phép toán ở sau được đẩy vào một hàng chờ (stack), chỉ khi nào các phép toán ưu tiên hơn ở sau được tính xong, các phép toán ở trước mới được xử lý.