SETL Newsletter Number 65 February 2, 1972

Some notational suggestions Robert A. Bonic

I. Present Functional Notation.

Given sets X and Y let

$$\pi(x): X \times Y \to X \ , \quad \text{and}$$

$$\pi(y): X \times Y \to Y$$

be the projection maps.
Given a relation rela $\subset X \times Y$ and sets $A \subset X$, $B \subset Y$, SETL notation provides for the set

$$\pi(y)((\pi(x)^{-1}(A)) \cap \text{rela})$$

but not for the set

$$\pi(x)((\pi(y)^{-1}(B)) \cap \text{rela}) \ .$$

More exactly,

$$\pi(y)((\pi(x)^{-1}(A)) \cap \text{rela}) \quad \text{(or}$$

$$\{t \in Y \mid \exists a \in A, \langle a,t\rangle \in \text{rela}\})$$

has the SETL form

$$\text{rela } [A] \ ,$$

but there is no similar notation for

$$\{s \in X \mid \exists b \in B, \langle s,b\rangle \in \text{rela}\} \ .$$

II. A More Symmetric Notation.

Given $A \subset X$, $B \subset Y$, rela $\subset X \times Y$
a symmetric function notation is given by

$$[A]\ \text{rela} = \{y \in Y \mid \exists a \in A, \langle a,y\rangle \in \text{rela}\}$$

$$\text{rela}[B] = \{x \in X \mid \exists b \in B, \langle x,b\rangle \in \text{rela}\} \ ,$$

along with the related notations

{a} rela in place of [{a}] rela and
(a) rela in the single valued case (likewise for rela {b} and rela (b)).

III. Containment Notation.

In some occasions in place of the notations

$$x \in A \quad \text{and} \quad B \subset A$$

the notations

$$A\ x \quad \text{and} \quad A[B]$$

are more fluid.

IV. Example of the Use of the Above Notations.

We begin with a relation

before

on the set

nodes = hd[before] u tℓ [before]

```
definef derivative(before);
/* we first define an auxiliary relation 'outrank' */
outrank = nℓ;  (∀nodes x) hisinfs = {x};
(while ({y ∈ ([hisinfs] before minus hisinfs)|hisinfs[before(y)]
        is later) ne nℓ)
   (∀later y) y in hisinfs;  end ∀later;  end while;
        (x)outrank = hisinfs less x;  end ∀nodes;
        outranked = tℓ[outrank];
        heads = nodes minus outranked;
   (∀heads y) intervalof(y) = (y) outrank less y;  end ∀heads;
/*  will now create the nodes for the derivative  */
   over = nℓ; newnodes = nℓ;
   (∀heads x) y = newat;  y in newnodes;
        (y)over = x;  end ∀heads;
/*  finally  */  before2 = n ;
   (∀newnodes x)
        (x) before2 = over[[intervalof((x)over)]before];
   end ∀newnodes;
        return before2;  end derivative;
```