PRIMITIV-REKURSIVE WORTBEZIEHUNGEN IN DER PROGRAMMIERUNGSSPRACHE »ALGOL 60«

von

Rózsa Péter

1. In der Zeitschrift »Numerische Mathematik« ist in diesem Jahr ein Referat erschienen über die algorithmische Sprache »Algol 60«, welche zum Zweck der Programmierung von Rechenautomaten konstruiert wurde. Darin werden Definitionen von gewissen Prädikaten angegeben, die für den ersten Augenblick zirkelhaft aussehen; es wird aber die Bemerkung gemacht, dass es sich dabei um gewisse Rekursionen handelt. Tatsächlich ergeben sich bei exakterer Fassung die genannten Prädikate als primitiv-rekursive Beziehungen auf einer Wortemenge.

2. Die sogenannten »Worte« einer Wortemenge entstehen ähnlich wie die natürlichen Zahlen, nur statt aus 0 ausgehend eine einzige Nachfolgerfunktion zu benutzen, geht man hier aus dem üblich mit Λ bezeichneten leeren Wort aus, und benutzt zur Bildung der Worte das Anknüpfen der »Buchstaben« eines »Alphabetes«. In einer im Acta Hungarica erscheinenden Arbeit, worüber ich schon am Warschauer Symposium über infinitistische Methoden der Grundlagenforschung in 1959 berichtet habe, habe ich allgemein die »zahlenartig aufbaubaren Mengen«, darunter als einen Spezialfall auch die Wortemengen, und die auf diesen definierbaren rekursiven Funktionen untersucht. Es werden hier die Ergebnisse dieser Arbeit verwendet.

Das Schema der primitiven Rekursion auf einer Wortemenge $\mathfrak{M}$ mit dem Alphabet $\mathfrak{A} = \{\ldots, a_i, \ldots\}$ lautet:

$$f(\Lambda) = k\,,$$

dann, falls $a_i \in \mathfrak{A}$

(D) $$f(a_i) = g_{a_i},$$

endlich, wenn $a_i,\ a_j \in \mathfrak{A}$,

$$f(a_i x a_j) = g'_{a_j}(a_i x, x a_j, f(a_i x), f(x a_j))\,,$$

wobei k konstant, g_{a_i} bei jedem a_i konstant, und g'_{a_j} für jedes a_j eine bereits definierte Funktion ist (natürlich können dabei auch Parameter auftreten, diese schreibe ich der Kürze halber nicht aus).

$a_i x$, $x a_j$ sind die »unmittelbaren Vorgänger« von $a_i x a_j$, d. h. solche, die um 1 weniger Buchstaben enthalten. Als Vorgänger eines Wortes gelten seine zusammenhängende Bestandteile; für diese wird auch eine zweckmässige

Anordnung angegeben. In dieser Anordnung sind die Vorgänger des Wortes $x = a_1 a_2 \ldots a_n$, wobei $a_i \in \mathfrak{A}$ für $i = 1, 2, \ldots, n$ ist:

$$\Lambda, a_1, a_2, a_1 a_2, \ldots, a_{n-1}, \ldots, a_1 a_2 \ldots a_{n-1}, a_n, \ldots, a_2 a_3 \ldots a_n, a_1 a_2 \ldots a_n .$$

Die unmittelbaren Vorgänger von x sind für $n > 1$

$$a_1 a_2 \ldots a_{n-1} \quad \text{und} \quad a_2 a_3 \ldots a_n ;$$

jedes $a_i \in \mathfrak{A}$ hat den einzigen echten Vorgänger Λ. Man sieht, dass alle echten Vorgänger von x Vorgänger seiner unmittelbaren Vorgänger sind. Als Zeichen dafür, dass y ein Vorgänger bzw. echter Vorgänger von x ist, wird

$$y \leqq x \quad \text{bzw.} \quad y \prec x$$

verwendet.

Eine Wortfunktion ist in $\mathfrak{M}$ primitiv-rekursiv, wenn sie von dem leeren Wort Λ, von der charakteristischen Funktion der Gleichheit und von den Anknüpfungsfunktionen xa_i ausgehend, wo $a_i \in \mathfrak{A}$, durch endlichviele Substitutionen und primitive Rekursionen aufgebaut werden kann.

Eine Wortbeziehung $B(x_1, \ldots, x_n)$ ist primitiv-rekursiv in $\mathfrak{M}$, wenn ihre charakteristische Funktion primitiv-rekursiv ist, welche etwa wie folgt definiert werden kann:

$$b(x_1, \ldots, x_n) = \begin{cases} \Lambda, & \text{wenn } B(x_1, \ldots, x_n) \\ a_0 & \text{sonst,} \end{cases}$$

wobei a_0 ein festes Element des Alphabets ist.

Ich habe bewiesen, dass jeder Wortefolge $x_0, x_1, \ldots, x_n$ ein Wort $c_n(x_0, x_1, \ldots, x_n)$ als für festes n in $\mathfrak{M}$ primitiv-rekursive Funktion derart zugeordnet werden kann, dass sich daraus die Glieder $x_0, x_1, \ldots, x_n$ bzw. die »Länge« n der Folge mit Hilfe von in $\mathfrak{M}$ primitiv-rekursiven Funktionen $k_i(x)$ bzw. long (x) folgendermassen ergeben: für $i = 0, 1, \ldots, n$

$$x_i = k_i(c_n(x_0, \ldots, x_n))$$

und

$$n = \operatorname{long}(c_n(x_0, \ldots, x_n)) .$$

Mit diesen lässt sich eine Wertverlaufsfunktion $\varphi(x)$ einer Funktion $f(x)$ folgenderweise definieren: wenn

$$\bar{x}_0, \bar{x}_1, \ldots, \bar{x}_s = x$$

sämtliche Vorgänger von x sind, in der genannten Reihenfolge, dann sei

$$\varphi(x) = c_s(f(\bar{x}_0), f(\bar{x}_1), \ldots, f(\bar{x}_s)) .$$

Daraus ergibt sich $f(\bar{x}_i)$ für $i \leq s$ als

$$f(\bar{x}_i) = k_i(\varphi(x)) ;$$

ferner gilt

$$s = \operatorname{long}(\varphi(x)) .$$

Wird in der Definition (D) $f(a_i x)$ und $f(ax_j)$ durch $\varphi(a_i x)$ bzw. $\varphi(xa_j)$ ersetzt, so wird aus (D) eine Wertverlaufsrekursion (D*); und es ist leicht zu beweisen, dass solche Definitionen nicht von der Klasse der primitiv-rekursiven Funktionen von $\mathfrak{M}$ hinausführen.

Dabei sieht man, dass bei primitiv-rekursivem $f(x)$ auch $\varphi(x)$ primitiv-rekursiv ist. Die Funktion $f(x) = x$ ist — wie man leicht zeigt — in jeder Wortemenge primitiv-rekursiv, daher ist ihre Wertverlaufsfunktion

$$\varphi_0(x) = c_s(\bar{x}_0, \bar{x}_1, \ldots, \bar{x}_s)$$

primitiv-rekursiv in $\mathfrak{M}$, und daraus ergeben sich die Vorgänger von x für $i \leq s = \text{long}\,(\varphi_0(x))$ als in $\mathfrak{M}$ primitiv-rekursive Funktionen:

$$\bar{x}_i = v_i(x) = k_i(\varphi_0(x)) \,.$$

Es haben sich eine Schar von Wortfunktionen und Wortbeziehungen als primitiv-rekursiv erwiesen; diese werde ich in den Folgenden benutzen. Insbesondere werde ich mich auf die folgenden Ergebnisse berufen:

1) Gehören a_i und a_j zum Alphabet einer Wortemenge $\mathfrak{M}$, so ist

$$a_i x a_j = d(a_i x, x a_j)\,,$$

mit in $\mathfrak{M}$ primitiv-rekursivem d.

2) Aussagenlogische Verknüpfungen überführen primitiv-rekursive Beziehungen wieder in solche.

3) Samt $f(x)$ und $g(x)$ ist auch die Beziehung

$$(Ey)\,[y \leqq f(x) \;\&\; g(y) = \Lambda]$$

primitiv-rekursiv in einer Wortemenge.

4) Falls in einer Wortemenge für jede Stelle $x_1, \ldots, x_n$ eine und nur eine der primitiv-rekursiven Beziehungen $B_i\,(x_1, \ldots, x_n)$ $(i = 1, 2, \ldots, r)$ gilt, dann ist auch die aus primitiv-rekursiven Funktionen $f_i(x_1, \ldots, x_n)$ »zusammengeflickte« Funktion

$$f(x_1, \ldots, x_n) = \begin{cases} f_1(x_1, \ldots, x_n)\,, & \text{falls} \quad B_1(x_1, \ldots, x_n) \\ \ldots\ldots\ldots\ldots\ldots\ldots & \\ f_r(x_1, \ldots, x_n)\,, & \text{falls} \quad B_r(x_1, \ldots, x_n) \end{cases}$$

primitiv-rekursiv.

3. Nun enthält das in der Sprache »Algol 60« in Betracht kommende Alphabet die Grundsymbole dieser Sprache: Buchstaben, Ziffern, Zeichen für logische Werte, Operationszeichen, Klammern, Trennungszeichen; auch einige fett gedruckten Worte und Wortefolgen, die als einzige »Buchstaben« des Alphabets zu betrachten sind, z. B. **else** oder **go to.** Die natürlichen Zahlen 1, 2, 3, . . . können in der so entstehenden Wortemenge $\mathfrak{M}$ etwa mit 0, 00, 000, . . . identifiziert werden; dann ist $m < n$, wobei m, n natürliche Zahlen sind, mit $m \prec n$ gleichbedeutend.

In der Sprache »Algol 60« werden Begriffe auf folgende Weise definiert: Z. B. »Ein Faktor ist entweder ein Primausdruck, oder ein Faktor hoch ein Primausdruck«. Eine solche Definition — worin auch der definierte Begriff benutzt wird — ist nur dann nicht zirkelhaft, wenn sie nur ein kurzer Ausdruck folgender Aussage ist: »Jeder Primausdruck gilt als Faktor, und wenn

F bereits ein Faktor und P ein Primausdruck ist, so ist F hoch P, mit der linearen Bezeichnung im Algol

$$F \uparrow P,$$

auch ein Faktor. Wird das Prädikat »x ist ein Faktor« mit $F(x)$ bezeichnet, und »x ist Primausdruck« mit $P(x)$, so kann diese Definition exakt wie folgt aufgeschrieben werden:

$$F(x) \equiv P(x) \vee (Ey)\,(Ez)\,[F(y)\,\&\,P(z)\,\&\,x = y \uparrow z]\,.$$

Dabei sind hier die Quantoren beschränkt, und zwar ist sowohl y als auch z ein echter Vorgänger von x. Dies benutzend könnte man, falls $P(x)$ bereits eine primitiv-rekursive Beziehung in $\mathfrak{M}$ wäre, auch $F(x)$ als eine solche nachweisen.

Es tritt aber auch eine weitere Komplikation auf: in der Definition von $P(x)$ wird auch das Prädikat »arithmetischer Ausdruck zu sein« verwendet, und in der Definition dieses Prädikats $A(x)$ wird wieder $F(x)$ benutzt. Das scheint nun wieder zirkelhaft zu sein.

In der exakten Fassung kommt aber daraus wieder kein Zirkel, sondern eine simultane Definition für $F(x)$, $P(x)$, $A(x)$ und für weitere, in der Definition auftretende Prädikate heraus, worin z.B. $P(x)$ wiederum mit Benutzung solcher $A(y)$ definiert wird, wobei y echter Vorgänger von x ist.

Statt dieser sehr langwierigen Definition möchte ich hier eine einfachere simultane Definition nur zweier Prädikate behandeln; aber der dabei benutzte Gedankengang lässt sich ganz ähnlich auf alle im »Algol 60« vorkommenden Definitionen anwenden.

4. Sei die simultane Definition der Beziehungen $F_0(x)$ und $F_1(x)$ (dementsprechend, dass das leere Wort z. B. auch als ein spezieller arithmetischer Ausdruck gelten kann): für $r = 0,1$

$$F_r(\Lambda) = \textbf{wahr},$$

und für $x \neq \Lambda$

$$F_r(x) \equiv (Ey)\,(Ez)\,[y, z \prec x\,\&\,F_0(y)\,\&\,F_1(z)\,\&\,x = g_r(y, z)]\,,$$

wobei $g_0(y, z)$ und $g_1(y, z)$ in $\mathfrak{M}$ primitiv-rekursive Funktionen sind. Ich werde beweisen, dass $F_0\,(x)$ und $F_1\,(x)$ primitiv-rekursive Beziehungen in $\mathfrak{M}$ sind, d. h. dass ihre charakteristische Funktionen primitiv-rekursiv sind, welche etwa derart definiert werden können: für $r = 0,1$

$$f_r(x) = \begin{cases} \Lambda, & \text{falls } F_r(x) \\ 0 & \text{sonst.} \end{cases}$$

Ausführlich ausgeschrieben erhält man für $r = 0,1$

$$f_r(\Lambda) = \Lambda$$

und für $x \neq \Lambda$

$$f_r(x) = \begin{cases} \Lambda, & \text{falls } (Ey)\,(Ez)\,[y, z \prec x\,\&\,f_0(y) = f_1(z) = \Lambda\,\&\,x = g_r(y, z)] \\ 0 & \text{sonst.} \end{cases}$$

Diese simultane Definition für $f_0(x)$ und $f_1(x)$ kann mit Hilfe der in Nr. 2 eingeführten Hilfsfunktionen wie üblich auf die Definition einer einzigen Funktion $f(x)$ aufgelöst werden. Dabei ist

$$f(x) = c_1(f_0(x), f_1(x)),$$

woraus sich

$$f_0(x) = k_0(f(x)), \qquad f_1(x) = k_1(f(x))$$

ergibt; so werden in $\mathfrak{M}$ samt $f(x)$ auch $f_0(x)$ und $f_1(x)$ primitiv-rekursiv ausfallen. Wir haben also nur $f(x)$ zu untersuchen.

5. Erstens gilt nach den Definitionen von $f(x)$, $f_0(x)$ und $f_1(x)$

$$f(\Lambda) = c_1(f_0(\Lambda), f_1(\Lambda)) = c_1(\Lambda, \Lambda). \tag{1}$$

Dann gilt für ein beliebiges $a_i \in \mathfrak{A}$ (da echter Vorgänger von a_i nur Λ, und $f_0(\Lambda) = f_1(\Lambda) = \Lambda$ ist):

$$f(a_i) = c_1(f_0(a_i), f_1(a_i)) = \begin{cases} c_1(\Lambda, \Lambda), \text{ falls } g_0(\Lambda, \Lambda) = a_i \mathbin{\&} g_1(\Lambda, \Lambda) = a_i \\ c_1(\Lambda, 0), \text{ falls } g_0(\Lambda, \Lambda) = a_i \mathbin{\&} g_1(\Lambda, \Lambda) \neq a_i \\ c_1(0, \Lambda), \text{ falls } g_0(\Lambda, \Lambda) \neq a_i \mathbin{\&} g_1(\Lambda, \Lambda) = a_i \\ c_1(0, 0), \text{ falls } g_0(\Lambda, \Lambda) \neq a_i \mathbin{\&} g_1(\Lambda, \Lambda) \neq a_i. \end{cases}$$

Nach 2) und 4) der Nr. 2 ist $f(a_i)$ als »zusammengeflickte« Funktion primitiv-rekursiv in $\mathfrak{M}$; ihr Wert ist daher für jedes feste a_i eine bestimmte Konstante g_{a_i}. Es gilt damit für $a_i \in \mathfrak{A}$

$$f(a_i) = g_{a_i}. \tag{2}$$

6. Endlich ergibt sich bei $a_i, a_j \in \mathfrak{A}$ für

$$f(a_i x a_j) = c_1(f_0(a_i x a_j), f_1(a_i x a_j))$$

auch eine »zusammengeflickte« Definition, da $f_0(x)$ und $f_1(x)$ nur die Werte Λ und 0 annehmen, und daher der Wert $f(a_i x a_j)$ nur einer der vier Werte

$$c_1(\Lambda, \Lambda),\ c_1(\Lambda, 0),\ c_1(0, \Lambda),\ c_1(0, 0)$$

sein kann, je nachdem von den beiden Beziehungen

I. $\quad (Ey)\,(Ez)\,[y, z \prec a_i x a_j \mathbin{\&} k_0(f(y)) = k_1(f(z)) = \Lambda \mathbin{\&} a_i x a_j = g_0(y, z)]$

und

II. $\quad (Ey)\,(Ez)\,[y, z \prec a_i x a_j \mathbin{\&} k_0(f(y)) = k_1(f(z)) = \Lambda \mathbin{\&} a_i x a_j = g_1(y, z)]$

beide wahr sind (also auch ihre Konjunktion), oder die Konjuktion von I. mit der Negation von II., oder die Konjuktion der Negation von I. mit II., oder endlich die Konjuktion der Negationen von I. und von II. wahr ist. Es genügt hier I. zu untersuchen; II. und die genannten Konjuktionen lassen sich genau so erledigen.

7. Nun sieht man, dass in I. der Wert von f an der Stelle $a_i x a_j$ mit Hilfe von Werten $f(y)$, $f(z)$ definiert wird, wobei y und z echte Vorgänger der Stelle $a_i x a_j$ sind. So ist naheliegend, dass es sich hier um eine Wertverlaufsrekursion handelt. Eine Schwierigkeit bedeutet nur, dass y und z gebundene Variablen

sind. Das lässt sich ausschalten, wenn wir die Wertverlaufsfunktion $\varphi(x)$ unserer Funktion $f(x)$ verwenden.

Wie ich bereits erwähnt habe, sind die echten Vorgänger von $a_i x a_j$ Vorgänger von einem der unmittelbaren Vorgänger $a_i x$ oder $x a_j$. Seien sämtliche Vorgänger von $a_i x$ bzw. $x a_j$ in der genannten Reihenfolge:

$$\bar{x}_0', \bar{x}_1', \ldots, \bar{x}_{s'}' \quad \text{bzw.} \quad \bar{x}_0'', \bar{x}_1'', \ldots, \bar{x}_{s''}'' .$$

Es ist klar, da $a_i x$ von gleichvielen Buchstaben besteht wie $x a_j$, dass sie beide gleichviele Vorgänger haben, also dass $s' = s''$ ist und wir wissen bereits dass

$$s' = \operatorname{long}\bigl(\varphi(a_i\, x)\bigr)$$

ist. So gilt für die echten Vorgänger y und z von $a_i x a_j$ mit gewissen $t,\ u \leq s'$

$$y = \bar{x}_t' \text{ oder } \bar{x}_t'', \qquad z = \bar{x}_u' \text{ oder } \bar{x}_u'',$$

d. h. mit den Bezçihungen der Nr. 2:

$$y = v_t(a_i\, x) \text{ oder } v_t(x a_j), \quad z = v_u(a_i\, x) \text{ oder } v_u(x a_j),$$

und daher ist $f(y)$ das Glied mit Index t, und $f(z)$ das Glied mit Index u der Folge, welcher $\varphi(a_i x)$ oder $\varphi(x a_j)$ zugeordnet wird, d. h.

$$f(y) = k_t\bigl(\varphi(a_i\, x)\bigr) \text{ oder } k_t\bigl(\varphi(x a_j)\bigr),$$

$$f(z) = k_u\bigl(\varphi(a_i\, x)\bigr) \text{ oder } k_u\bigl(\varphi(x a_j)\bigr).$$

Nach 1) der Nr. 2 gilt noch mit in $\mathfrak{M}$ primitiv-rekursivem d

$$a_i x a_j = d(a_i\, x,\ x a_j).$$

Endlich ist die Existenz von y und z der gewünschten Art gleichbedeutend mit der Existenz von entsprechenden Indizes t und u.

Wird das alles in I. eingesetzt, so erhält man zufolge der Fallunterscheidungen eine 4-gliedrige Disjunktion. Das erste Disjunktionsglied ist:

$$\text{III.} \quad (Et)\,(Eu)\,\Bigl[t, u \leqq \operatorname{long}\bigl(\varphi(a_i\, x)\bigr) \mathbin{\&} k_0\Bigl(k_t\bigl(\varphi(a_i\, x)\bigr)\Bigr) = k_1\Bigl(k_u\bigl(\varphi(a_i\, x)\bigr)\Bigr) = \Lambda \mathbin{\&}$$

$$\mathbin{\&}\, d(a_i\, x,\ x a_j) = g_0\bigl(v_t(a_i\, x),\ v_u(a_i\, x)\bigr)\Bigr],$$

und die drei anderen Disjunktionsglieder unterscheiden sich davon nur darin, dass in ihnen der Reihe nach für das 3-te, 6-te, dann für das 2-te, 5-te, endlich für das 2-te, 3-te, 5-te, 6-te Auftreten von $a_i x$ immer $x a_j$ gesetzt wird. Setzen wir

$$B'(x_1, x_2, x_3, x_4) \equiv (Et)\,(Eu)\,\bigl[t, u \leq \operatorname{long}(x_3) \mathbin{\&} k_t\bigl(k_0(x_4)\bigr) = k_1\bigl(k_u(x_4)\bigr) = \Lambda \mathbin{\&}$$

$$\mathbin{\&}\, d(x_1, x_2) = g_0\bigl(v_t(x_1),\ v_u(x_1)\bigr)\bigr],$$

dann ergibt sich diese Beziehung mit zweimaliger Anwendung von 3) der Nr. 2 als primitiv-rekursiv in $\mathfrak{M}$, und III. ist äquivalent mit

$$B'\bigl(a_i\, x,\ x a_j,\ \varphi(a_i\, x),\ \varphi(a_i\, x)\bigr).$$

Ganz ähnlich ergeben sich die drei anderen Disjunktionsglieder als auf $a_i x$, $x a_j$, $\varphi(a_i x)$ und $\varphi(x a_j)$ angewandte, in $\mathfrak{M}$ primitiv-rekursive Beziehungen, und nach 2) der Nr. 2 gilt dasselbe auch für ihre Disjunktion.

8. Daher ist I. der Nr. 6 mit in $\mathfrak{M}$ primitiv-rekursivem $B^{(1)}$ der Beziehung

$$B^{(1)}(a_i x, xa_j, \ \varphi(a_i x), \varphi(xa_j))$$

äquivalent. Genau so erhält man eine mit II. der Nr. 6 äquivalente Beziehung

$$B^{(2)}(a_i x, xa_j, \varphi(a_i x), \varphi(xa_j)),$$

und aus diesen die in Nr. 6 aufgezählten Konjuktionen als

$$B_w(a_i x, xa_j, \varphi(a_i x), \varphi(xa_j)) \qquad (w = 1, 2, 3, 4),$$

wobei B_1, B_2, B_3, B_4 in $\mathfrak{M}$ primitiv-rekursive Beziehungen sind. So ergibt sich für a_i, $a_j \in \mathfrak{A}$

$$f(a_i x a_j) = \begin{cases} c_1(\Lambda, \Lambda), \text{ falls } B_1(a_i x, xa_j, \varphi(a_i x), \varphi(xa_j)) \\ c_1(\Lambda, 0), \text{ falls } B_2(a_i x, xa_j, \varphi(a_i x), \varphi(xa_j)) \\ c_1(0, \Lambda), \text{ falls } B_3(a_i x, xa_j, \varphi(a_i x), \varphi(xa_j)) \\ c_1(0, 0), \text{ falls } B_4(a_i x, xa_j, \varphi(a_i x), \varphi(xa_j)). \end{cases}$$

Die durch

$$h(x_1, x_2, x_3, x_4) = \begin{cases} c_1(\Lambda, \Lambda), \text{ falls } B_1(x_1, x_2, x_3, x_4) \\ c_1(\Lambda, 0), \text{ falls } B_2(x_1, x_2, x_3, x_4) \\ c_1(0, \Lambda), \text{ falls } B_3(x_1, x_2, x_3, x_4) \\ c_1(0, 0), \text{ falls } B_4(x_1, x_2, x_3, x_4) \end{cases}$$

definierte »zusammengeflickte« Funktion h ist in $\mathfrak{M}$ primitiv-rekursiv, und damit gilt für a_i, $a_j \in \mathfrak{A}$

$$f(a_i x a_j) = h(a_i x, xa_j, \varphi(a_i x), \varphi(xa_j)). \tag{3}$$

9. Wie wir an der in Nr. 2 angedeuteten Modifizierung (D*) von (D) sehen, ergibt (1), (2) und (3) für $f(x)$ eine Wertverlaufsrekursion in $\mathfrak{M}$, und somit ist $f(x)$, und damit auch $f_0(x)$ und $f_1(x)$ primitiv-rekursiv in $\mathfrak{M}$.

Da sie ferner die charakteristischen Funktionen der Beziehungen $F_0(x)$, $F_1(x)$ sind, gilt das auch für diese Beziehungen.

Ähnlich beweist man, dass alle in der Sprache »Algol 60« definierten Prädikate in der Algol-Wortemenge $\mathfrak{M}$ primitiv-rekursiv sind.

(Eingegangen: 8. Oktober, 1960.)

ПРИМИТИВНО-РЕКУРСИВНЫЕ СЛОВОСООТНОШЕНИЯ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ

R. PÉTER

Резюме

В журнале «Numerische Matematik» в этом году было опубликовано сообщение об алгорифмическом языке «ALGOL 60» сконструированном для программирования вычислительных машин. В нем даются такие определения некоторых предикатов, которые в первой момент кажутся пороч-

ным кругом; но делается замечания, что здесь речь идет о взятых в некотором смысле рекурсиях. При более точной формулировке эти предикаты действительно оказываются примитивно-рекурсивными реляциями на некотором множестве слов.

Так называемые «слова» некоторого множества слов получаются так же, как и натуральные числа, только вместо того, чтобы исходя из 0 применять единственную функцию образования следующего элемента, здесь обычно исходят из пустого слова Λ и образуют слова добавлением «букв» некоторого «алфавита». В одной работе, публикуемой в Acta Mathematica Academiae Scientiarum Hungaricae, о которой я уже доложила в 1959-ом году на варшавском симпозиуме о нефинитным методах исследовиная онсований математики, я в общем виде исследовала «множества, которые могут бть построенны как числа», — среди них в качестве специального случая и множества слов — и рекурсивные функции, которые на них можно определить. Здесь используются результаты этой работы.

«Алфавит» языка «ALGOL 60» содержит основные символы этого языка: буквы, числа, символы логических значений, знаки действий, скобки, знаки препинания, несколько крупно набранных слов и их последовательностей, также считающихся буквами «алфавита», например *else* или *go to* На основанном на нем множестве слов для характеристических функций определенных на языке «ALGOL 60» предикатов сначала могут быть получены рекурсии для совместного множества значений; которые могут затем привести к примитивным рекурсиям.