BULLETIN DE L'ACADÉMIE
POLONAISE DES SCIENCES
Série des sciences math., astr.
et phys. — Vol. XVIII, No. 8, 1970

MATHEMATICS
(COMPUTING MACHINES)

On the Notion of the Description of the Program

by

A. GÓRAJ, M. MIRKOWSKA and A. PALUSZKIEWICZ

Presented by A. MOSTOWSKI on March 6, 1970

The purpose of this paper is to systematize some ideas introduced by R. W. Floyd and to formulate and to prove the corresponding theorems. The notation used in this paper differs a little from the one used in [3] (e.g. a program is defined approximately as in Engeler [2] because we have found this definition more useful than one given by Floyd). The notion of description of a program introduced here is similar to Floyd's interpretation of program. We consider the following problems:

(1) semantical and syntactical characterization of the sets of descriptions (feasible and acceptable descriptions); We give a proof of a theorem (Theorem 5) about their equivalence,

(2) a method of building feasible descriptions with the aid of a transformation which is similar to the strongest consequent operation (defined by Floyd); this method is given by Theorem 1 but only for programs without loops;

(3) methods of achieving new feasible (acceptable) descriptions from different feasible (acceptable) descriptions — Theorem 4.

1. Let $\mathcal{L}=\langle A, T, F\rangle$ be a formalized language of the first order [see 4, V, § 3]. Let us additionally assume that the language $\mathcal{L}$ contains a binary predicate$=$(sign of equality).

Let $A^P=A \cup L \cup U$, where A denotes an alphabet of the first order language $\mathcal{L}$, L is an enumerable set disjoint with A and U. The set L will be called a set of labels. The elements of L will be denoted by l (with indices if necessary). U is a set of auxiliary signs $U=\{$**if, then, else, do, next, goto,:**$\}$

By a language of programs we shall understand any system

$$\mathcal{L}^P=\langle A^P, T, F, S, P\rangle,$$

where A^P denotes an alphabet defined as above, T is the set of terms defined as in formalized language of the first order, F is the set of open formulae defined as in formalized language of the first order [see 4, V, § 3], S is the set of substitutions (see below), P is the set of programs which will be defined below (after some auxiliary definitions).

If a denotes an expression of the form $a_1\, xa_2\, x \ldots xa_{n+1}$ (where a_i, $i=1, \ldots, n+1$, may be empty), then the expression

$$a_1\, \tau a_2\, \tau \ldots \tau a_{n+1}$$

is said to be obtained from a by simultaneous replacement of all occurrences of the sign x by the expression τ and will be denoted by $a\,(x/\tau)$ or sa, provided s denotes the expression of the form $[x/\tau]$. Let τ be a term and x an individual variable, any expression of the form $[x/\tau]$ will be called a substitution of the term τ for the variable x. A set of all such substitutions will be denoted by S.

Let l, l', l'' be elements of L. Let s be a substitution, and a a quantifier-free formula. By an *instruction* we shall understand any expression of the form

$$l\text{: } \mathbf{do}\ s\ \mathbf{next\ if}\ a\ \mathbf{then\ goto}\ l'\ \mathbf{else\ goto}\ l''$$

when $l' \neq l''$. Label l will be called a label of the instruction.

By a *program* Π we shall understand any finite sequence of instructions with distinguished one instruction (called the initial instruction) and such that every two instructions preceded by the same label are identical.

A label is said to be an *entrance* to a program if it is a label of its initial instruction.

By the *exit* of a program we shall understand every label which appears after the symbol **goto**, such that none of instructions of this program is preceded by this label.

Let R denote a realization of the languare $\mathcal{L}^P$ in a non-empty set J and a two--element Boolean algebra $B=\{0, 1\}$ [see 4, VI, § 6] and let v be a valuation in J. Similarly as in [2], for any term $\tau \in T$ (formula $a \in F$) the expression $\tau_R\,(v)$ $(a_R\,(v))$ will denote a value of the term τ (of the formula a) in the realization R at the point $v \in J$.

If $s \in S$ then by $s_R\,(v)$ we shall understand a valuation obtained from v by the change of value of exactly one variable pointed out by substitution s.

We shall consider any consistent and complete theory $\mathcal{T}=\{\mathcal{L}^P, C, \mathcal{A}\}$ [see 4, VII, § 1] and only such its realizations R that:

1) each realization R is a model for $\mathcal{T}$,

2) sign of equality is realized as the identity.

Let Π denote a program in language $\mathcal{L}^P$ and let $L\,(\Pi)$ be a set of labels appearing in Π.

By a *graph* $\Gamma\,(\Pi)$ of the program we shall understand a system $\Gamma\,(\Pi)=\langle L\,(\Pi), K(\Pi)\rangle$ [see 1], where $K(\Pi)$ is a subset of cartesian product $L\,(\Pi) \times L\,(\Pi)$ such, that pair (w_1, w_2) belongs to $K\,(\Pi)$ if and only if instruction labelled by w_1 exists in program Π, and w_2 appears in this instruction after symbol **goto**. $L\,(\Pi)$ is called a set of vertices of graph $\Gamma\,(\Pi)$, and $K\,(\Pi)$ foims set of its edges.

2. Let $\Gamma\,(\Pi)=\langle L\,(\Pi), K\,(\Pi)\rangle$ be the graph of the program Π, and F—the set of formulae of the language $\mathcal{L}^P$. A mapping I_Π of the set $K\,(\Pi)$ of edges of the graph $\Gamma\,(\Pi)$ into the set F will be called the *description of the program* Π

$$I_\Pi : K\,(\Pi) \to F.$$

Let c be a fixed instruction of program labelled by l. Let us consider a set K of edges of the graph $\Gamma(\Pi)$ such that $(a, b) \in K$ if and only if $(a, b) \in K(\Pi)$ and either a or b is identical with l. A restriction of the mapping I_Π to the set K $(I_\Pi | K)$ defines the *description of the instruction c*. Formulae assigned (by the mapping I_Π) to the edges (w, l) $w \in L(\Pi)$ will be called *antecedents* of the instruction c. $I_\Pi(l, l')$ will be called the *right consequent* of the instruction c at the description I_Π, and $I_\Pi(l, l'')$ the *left consequent* of c. Let v be a fixed valuation then Nvc will be the formula defined as follows

$$Nvc = \begin{cases} I_\Pi(l, l') & \text{if} \quad a_R(s_R\, v) = 1, \\ I_\Pi(l, l'') & \text{if} \quad a_R(s_R\, v) = 0. \end{cases}$$

The formula Nvc will be called the consequent determinated by the valuation v and the instruction c.

Let c be an instruction, β an antecedent of c in a description I_Π and v a valuation.

We shall say, that the *description I_Π of instruction c is feasible in the realization R*, if the fact that $\beta_R(v) = 1$ implies that $Nvc_R(s_R\, v) = 1$.

The given *description of a program is feasible in the realization R* if descriptions of all instructions are feasible in this realization.

Description of program is feasible if it is feasible in every realization.

Let s be a substitution of the form $[x/\tau]$ and a a quantifier-free formula. By T_{sa} we understand the mapping of the' set F of formulae into the set F

$$T_{sa} : F \to F$$

which for every formula P assigns a formula $T_{sa}^+(P)$ or formula $T_{sa}^-(P)$ defined as follows:

$$T_{sa}^+(P) = \big(a \wedge \exists_\xi \big(P(x/\zeta) \wedge x = \tau(x/\zeta)\big)\big),$$
$$T_{sa}^-(P) = \big(\ominus\, a \wedge \exists_\xi \big(P(x/\zeta) \wedge x = \tau(x/\zeta)\big)\big).$$

¬

Let Π be a program, I_Π a descritpion of program Π and c an instruction of the following form

l: **do** s **next if** a **then goto** l' **else goto** l'',

where $s = [x/\tau]$, $x \in V$, $\tau \in T$, and let $P_1, \ldots, P_n$ be all antecedents of this instruction.

Using the above notation we have the following

THEOREM 1. *If formula $T_{sa}^+(\bigcup_{i=1}^{n} P_i)$ is a right consequent of the instruction c and formula $T_{sa}^-(\bigcup_{i=1}^{n} P_i)$ is a left consequent of c then the description of instruction c is feasible.*

Proof. Let for every fixed natural number k $(1 \leqslant k \leqslant n)$, valuation v and realization R,

(1) $$P_{\underline{\underline{K}}_R}(v) = 1. \qquad \leftarrow k$$

Observe that:

A) if β is the right consequent of instruction c, then Nvc is identical with β if and only if $a_R(s_R\, v) = 1$,

B) if γ is the left consequent of instruction c, then Nvc is identical with γ if and only if $a_R(s_R\, v)=0$.

Suppose that for a valuation v, $a_R(s_R\, v)=1$. We shall show that

$$T^+_{s\alpha}\,(\bigcup_{i=1}^{n} P_i)_R\,(s_R\, v)=1\,. \tag{2}$$

By definition of $T^+_{s\alpha}$ we have

$$T^+_{s\alpha}\,(\bigcup_{i=1}^{n} P_i)_R\,(s_R\, v)=\big(a\wedge\exists_\zeta\,(\bigcup_{i=1}^{n} P_i\,(x/\zeta)\wedge x=\tau\,(x/\zeta))\big)_R\,(s_R\, v)\,. \tag{3}$$

Let u be a variable that does not appear in the expression τ nor in any formula P_i ($i=1, \dots, n$). By the definition of realization R [see 4, VI, § 6] the right side of (3) is equal to

$$a_R\,(s_R\, v)\wedge\bigcup_{j\in J}\big(\bigcup_{i=1}^{n} P_i\,(x/u)\wedge\big(x=\tau\,(x/u)\big)\big)_R\,(w_j)\,,$$

where $w_j=\{w_{jz}\}_{z\in V}\in J^V$ is the valuation such that $w_{jz}=(sv)_z$ for $z\neq u$, $w_{ju}=j$.

It is enough to show that $\bigcup_{j\in J}\big(\bigcup_{i=1}^{n} P_i\,(x/u)\wedge\big(x=\tau\,(x/u)\big)\big)_R\,(w_j)=1$, since $a_R\,(s_R v)=1$.

Let j_0 an element of J such that $w_{j_0 u}=v_x$. For valuation w_{j_0} we have

$$\bigcup_{i=1}^{n} P_i\,(x/u)_R(w_{j_0})=\bigcup_{i=1}^{n} P_i\,(x)_R\,(v)=1,$$

and

$$\begin{aligned}\big(x=\tau\,(x/u)\big)_R\,(w_{j_0})&=\big(x_R\,(w_{j_0})=_R\,\tau\,(x/u)_R\,(w_{j_0})\big)=\big(x_R\,(sv)=_R\,\tau\,(x/u)_R\,(w_{j_0})\big)\\&=\big(\tau\,(x)_R\,(v)=_R\,\tau\,(x/u)_R\,(w_{j_0})\big)\,.\end{aligned}$$

Since the value of the variable u is equal to the value of x, we have

$$\big(x=\tau\,(x/u)\big)_R\,(w_{j_0})=1\,.$$

So, we have found $j_0\in J$ such that $\big(\bigcup_{i=1}^{n} P_i\,(x/u)\wedge\big(x=\tau\,(x/u)\big)\big)_R\,(w_{j_0})=1$ and consequently (2) holds. In a similar way we can obtain the proof for the case where $a_R\,(s_R\, v)=0$.

THEOREM 2. *Let $P'_1, P'_2, \dots, P'_n, P''_1, P''_2, \dots, P''_n$ be arbitrary formulae, then for any instruction every formula built according to one of following schemas is a theorem of predicate calculus.*

W1) $$\big(T_{s\alpha}\,(\bigcup_{i=1}^{n}(P'_i\vee P''_i))=T_{s\alpha}\,(\bigcup_{i=1}^{n} P'_i)\vee T_{s\alpha}\,(\bigcup_{i=1}^{n} P''_i)\big),$$

W2) $$\big(T_{s\alpha}\,(\bigcup_{i=1}^{n}(P'_i\wedge P''_i))=>T_{s\alpha}\,(\bigcup_{i=1}^{n} P'_i)\wedge T_{s\alpha}\,(\bigcup_{i=1}^{n} P''_i)\big),$$

W3) $$\big(T_{s\alpha}\,(\bigcup_{i=1}^{n}(\exists_\eta\, P'_i\,(\eta)))=\exists_\eta\, T_{s\alpha}\,(\bigcup_{i=1}^{n} P'_i\,(\eta))\big),$$

W4) $$\big((\bigcap_{i=1}^{n}(P'_i=>P''_i))=>\big(T_{s\alpha}\,(\bigcup_{i=1}^{n} P'_i)=>T_{s\alpha}\,(\bigcup_{i=1}^{n} P''_i)\big)\big),$$

W5) $$\big(T_{s\alpha}\,(\bigcup_{i=1}^{n}(\forall_\eta\, P'_i\,(\eta)))=>\forall_\eta\, T_{s\alpha}\,(\bigcup_{i=1}^{n} P'_i\,(\eta))\big),$$

The proof is omitted.

Let I_Π be a description of a program Π. Let c be an instruction of the program. We say that:

A description of the program Π is acceptable in the theory $\mathcal{T}$, iff for every pair of formulae β, γ, such that there exist $l, l', l'' \in L(\Pi)$ and $\beta = I_\Pi(l', l)$ and $\gamma = I_\Pi(l, l'')$ the following conditions hold: if l is a label of insfruction c then the corresponding formula is a theorem in theory $\mathcal{T}$

$$(T^+_{s\alpha}(\beta) => \gamma) \quad \text{when} \quad \gamma \quad \text{is the right consequent of} \quad c$$

or

$$(T^-_{s\alpha}(\beta) => \gamma) \quad \text{when} \quad \gamma \quad \text{is the left consequent of} \quad c.$$

THEOREM 3. *Let s be a substitution, a an open formula and $\beta, \gamma, \beta', \gamma'$ arbitrary formulae, then every formula built according to one of following schemas is a theorem in the theory $\mathcal{T}$.*

w1) $$\Big(\big((T_{s\alpha}(\beta) => \gamma) \wedge (T_{s\alpha}(\beta') => \gamma')\big) => \big(T_{s\alpha}(\beta \vee \beta') => (\gamma \vee \gamma')\big)\Big),$$

w2) $$\Big(\big((T_{s\alpha}(\beta) => \gamma) \wedge (T_{s\alpha}(\beta') => \gamma')\big) => \big(T_{s\alpha}(\beta \wedge \beta') => (\gamma \wedge \gamma')\big)\Big),$$

w3) $$\big((T_{s\alpha}(\beta) => \gamma) => (T_{s\alpha}(\exists_\zeta \beta) => \exists_\zeta \gamma)\big),$$

w4) $$\Big(\big((T_{s\alpha}(\beta) => \gamma) \wedge (\beta' => \beta) \wedge (\gamma => \gamma')\big) => \big(T_{s\alpha}(\beta') => \gamma'\big)\Big).$$

The proof is omitted.

Let $\circ$ denote one of the propositional connectives $\vee$, $\wedge$ and let I^1_Π, I^2_Π be the descriptions of a program Π acceptable in a theory $\mathcal{T}$. The equalities given below determinate a new description I_Π of the program Π:

$$I_\Pi(l_1, l) = (I^1_\Pi(l_1, l) \circ I^2_\Pi(l_1, l)),$$
$$I_\Pi(l, l_2) = (I^1_\Pi(l, l_2) \circ I^2_\Pi(l, l_2)),$$

for every triple of labels $l, l_1, l_2 \in L(\Pi)$. With the above assumptions the Theorem 3 implies the following:

THEOREM 4. *I_Π is an acceptable description of program Π.*

Let be given: a program Π, a description I_Π of the program Π, a theory $\mathcal{T} = \langle \mathcal{L}^P, C, \mathcal{A} \rangle$. The following theorem holds.

THEOREM 5. *A description I_Π is acceptable in theory $\mathcal{T}$ if and only if it is feasible in every model of theory $\mathcal{T}$.*

Proof. Suppose that the description I_Π of the program Π is not acceptable. Then there exist an insfruction $c \in \Pi$, its antecedent P and consequent Q, such that a formula $(T_{s\alpha}(P) => Q)$ is not a theorem in $\mathcal{T}$. So, by the completeness theorem for $\mathcal{T}$, the above formula is not satisfied in some model of considered theory $\mathcal{T}$, i.e. there exists a realization R and a valuation $\hat{v}$ such that

$$T_{s\alpha}(P)_R(\hat{v}) = 1 \quad \text{and} \quad Q_R(\hat{v}) = 0.$$

We shall consider only the case in which $a_R(\hat{v}) = 1$. By our assumption and the definition of $T_{s\alpha}(P)$ we have:

$$1 = T_{s\alpha}(P)_R(\hat{v}) = \Big(\alpha \wedge \big(\exists_\zeta \big(P(x/\zeta) \wedge (x = \tau(x/\zeta))\big)\big)\Big)_R(\hat{v}).$$

Let u be a variable not appearing in the term τ nor in any of the formulae Q, P, a. By the definition of a realization of a formula [see 4, VI, § 6] the right-hand side of the above equality is equal to

$$a_R(\hat{v}) \wedge \bigcup_{j \in J} \big(P(x/u) \wedge (x = \tau(x/u))\big)_R (w_j),$$

where $w_j = \{w_{jz}\}_{z \in V} \in J^V$ is the valuation such that $w_{jz} = \hat{v}_z$ for $z \neq u$, $w_{ju} = j$. Obviously $a_R(\hat{v}) = 1$. Let w_{j_0} be a valuation for which $\big(P(x/u) \wedge (x = \tau(x/u))\big)_R (w_{j_0}) = 1$ holds.

Now let v be a valuation defined as follows $v_x = w_{j_0 u}$, $v_z = w_{j_0 z}$ for $z \neq x$. From the above definitions we obtain $P(x)_R(v) = P(x/u)_R(w_{j_0})$. The valuation $s_R\, v$ differs from the valuation v only for value of x and $(s_R\, v)_x = \tau_R(v)$. Observe that $x_R(w_j) = \tau(x/u)_R(w_{j_0}) = \tau_R(v)$. Hence $s_R\, v = w_{j_0}$ that is $Q_R(s_R\, v) = Q_R(w_{j_0})$.

The valuation w_{j0} differs from the valuation $\hat{v}$ only for the value of variable u, by our assumption u does not appear in formula Q.

From the above we obtain $Q_R(s_R\, v) = Q_R(\hat{v})$ because the value of a formula Q does not depend on variables not appearing in it. So, we have proved that there exists a realization R and a valuation v such that for the instruction c (for which $(T_{sa}(P) => Q)$ is not a theorem in $\mathcal{T}$) $P_R(v) = 1$ and $Q_R(s_R\, v) = 0$. According to definition of the feasible description we obtain that I_Π is not feasible.

Now we shall prove that, if a description I_Π is acceptable in the theory $\mathcal{T}$ then I_Π is feasible in every model for this theory. Let I_Π be an acceptable description. We recal that a description I_Π is said to be acceptable if for every instruction c and its arbitrary antecedent P and consequent Q

1) $(T^+_{sa}(P) => Q)$ is a theorem in $\mathcal{T}$, where Q is a right consequent of c,

2) $(T^-_{sa}(P) => Q)$ is a theorem in $\mathcal{T}$, where Q is a left consequent of c.

By Theorem 1 we have for the case 1):

For every valuation v', if $P_R(v') = 1$ then $T^+_{sa}(P)_R(s_R\, v') = 1$.

Hence, by our assumption and by the completeness theorem: in every model R of the theory and for every valuation $(T_{sa}(P) => Q)_R(v) = 1$. From the above if $P_R(v') = 1$, then $Q_R(s_R\, v') = 1$, i.e. I_Π is a feasible description.

Similar proof can be repeated in the second case.

3. Finally, let us notice that a description of a program can be useful for proving some properties of the program. With the aid of a description we can find out, for instance, whether for a given initial valuation v the program will work endlessly in a cycle or not. Similarly, we can use a description to show that a program realizes an intended algorithm. Moreover, we can (with the instance of another feasible description) check whether the required accuracy was achieved.

In connection with the discussed problems the following questions arise:

1) how to build an acceptable description for a program which includes cycles?

2) how to extend a description of a part of a program (i.e. a subset of program instructions) to a feasible description of the whole program?

3) what are the operations on the descriptions which correspond to operations on the programs For instance, how to get a description of the program being a superposition of the given two programs?

The authors wish to thank Dr A. Salwicki for his interest in the problem, and for helpful suggestions. They are also grateful to the referee for valuable suggestions in the preparation of the manuscript.

INSTITUTE OF MATHEMATICAL MACHINES, UNIVERSITY, WARSAW
(INSTYTUT MASZYN MATEMATYCZNYCH, UNIWERSYTET, WARSZAWA)

REFERENCES

[1] C. Berge, *Théorie des graphes et ses applications*, Paris, 1958.

[2] E. Engeler, *Algorithmic properties of structures*, Math. Syst. Theory, 1967.

[3] R. W. Floyd, *Assigning meanings to programs*, Proceedings of Symposia in Applied Mathematics, 1967.

[4] H. Rasiowa, R. Sikorski, *Mathematics of metamathematics*, Warszawa, 1963.