regular fragments (ГГ= gli) are

глн	глу	гли
лей	вал	ала
лей	иле	тре
Осtre	лиз	арг

(the third letter is A)

фен	иле	тре	сер
тир	асн	сер	цис
гис	асп	гли	

(the third letter is any pyrimidine)

Several methods can be used to place all 20 amino acids of the standard collection into Table 1 in such a way that each acid would obtain its own cell. Among such systems the most regular one seems to be the following:

	мет			
	лиз	арг	три	
глн	глу	гли	арг	гис
лей	вал	ала	про	лей
лей	иле	тре	сер	фен
	асн	сер	цис	тир
	асп			

By a "cross" we have delineated cells invariant with respect to the third letter of the codon. To the left and above the diagonal lie the cells with purine as the third letter; to the right and below, with pyrimidine.

A NEW PROOF OF THE THEOREM ON EXPONENTIAL DIOPHANTINE REPRESENTATION OF ENUMERABLE SETS*

Yu. V. Matiyasevich

UDC 51.01:518.5

A new proof is given for the well-known theorem of Putnam, Davis, and Robinson on exponential diophantine representation of recursively enumerable sets. Starting from the usual definition of r.e. sets via Turing machines, a new method of arithmetization is given. This new method leads directly to a purely existential exponential formula. The new proof may be more suitable for a course on the theory of algorithms because it requires less knowledge of number theory.

1. The aim of the present paper is to give a simple proof of the well-known theorem of M. Davis, H. Putnam, and J. Robinson [1] that each recursively enumerable predicate $\mathcal{P}$ has an exponential diophantine expression, i.e., there exists for it a valid formula of the type

$$\mathcal{P}(a_1,\ldots,a_\mu)\Longleftrightarrow\exists x_1\ldots x_\nu\,[R=S], \tag{1}$$

where R and S are terms constructed from natural numbers and the variables $a_1,\ldots,a_\mu,x_1,\ldots,x_\nu$ by the operations of addition, multiplication, and raising to a power. (By natural numbers

*Translated by J. P. Jones and L. Guy.

Steklov Mathematical Institute, Academy of Sciences of the USSR, Leningrad. Translated from Zapiski Nauchnykh Seminarov Leningradskogo Otdeleniya Matematicheskogo Instituta im. V. A. Steklova Akad. Nauk SSSR, Vol. 60, pp. 75-92, 1976. Main results announced June 19, 1975.

0090-4104/80/1405-1475$07.50

we mean nonnegative integers; lower case Roman letters with or without subscripts are used as variables for natural numbers).

M. Davis and H. Putnam [2] originally proved this theorem in a conditional way, that is, they started from the unproved (up to now) assumption of the existence of arbitrarily long arithmetic progressions of primes. J. Robinson [3] modified their proof and rendered it independent of this conjecture. After considerable simplification this proof was published in the joint paper [1].

The Davis–Putnam–Robinson theorem is one of the steps in the negative solution of Hilbert's tenth problem (see [4], [5], [6]). After the diophantine nature of enumerable predicates was established, several modifications of the Davis–Putnam–Robinson proof were found, e.g., [5], [8], [10], which had various additional properties. All these modifications, as well as the original proof, took as their starting point the arithmetical representation of the predicate $\mathcal{P}$ with one bounded universal quantifier — a formula of the type

$$\mathcal{P}(a_1,\ldots,a_\mu) \Leftrightarrow \exists x_1 \ldots x_\nu \forall y_{\leqslant x_1} \exists z_1 \ldots z_\sigma [T = Q], \tag{2}$$

where T and Q are polynomials with integer coefficients in the variables $a_1, \ldots a_\mu, x_1, \ldots, x_\nu, y, z_1, \ldots, z_\sigma$. (Davis [7] established the existence of such an expression for each enumerable predicate.) In particular, in the search for diophantine expressions with the least possible number of variables, J. Robinson and the author (see [8], [5]) found a proof which, in place of the argument of prime divisors, used one involving greatest common divisors and an application of the multiplicative analog of the well-known Dirichlet principle. This proof did not require as a subsidiary result the exponential diophantine expressibility of the factorial, and permitted a substantial reduction in the number of variables.

By a slight modification of the original Davis–Putnam–Robinson proof, the author [9] strengthened their theorem, proving the existence of a singlefold exponential diophantine representation, that is, a representation of the type (1) in which for each set $a_1, \ldots, a_\mu$, satisfying $\mathcal{P}$, there exists only one ν-tuple $x_1, \ldots, x_\nu$, for which $R = S$. This required an analogous sharpening of (2) (namely, the uniqueness of $z_1, \ldots, z_\sigma$). In [9] the required strengthening of (2) was easily achieved by using a diophantine expression for $\mathcal{P}$; however it is also possible to give a direct proof of the necessary strengthening.

Hirose and Lida proposed [10] a variant of the removal from (2) of the bounded universal ($\forall$) quantifier on condition that the variable y is taken not over all numbers, but over the members of a special recurrent sequence. Such a variant of the representation is easily obtained from an arbitrary representation of this type under the condition that the recurrent sequence under consideration is established as diophantine.

The modification furthest removed from the original proof was proposed by J. Robinson and the author [11] for finding diophantine representations with 13 variables. However this modification is applicable only in the case where in (2) $\sigma = 1$, $Q = z_1$, and T does not contain z_1. In [11] the construction of the representation (2) with such properties essentially uses the fact that $\mathcal{P}$ already has a diophantine representation.

We shall give here a proof of the theorem on exponential diophantine representation of enumerable sets which is based on new ideas. Our basic apparatus for the representation of

enumerable sets will be the Turing machine. We shall show how a suitable arithmetization of the work of this machine will give at once a purely existential formula so that a representation of the Davis type (2) is not needed as a preliminary. This proof is also more suitable than the original for inclusion in a course in theory of algorithms because it presupposes less knowledge of number theory. It can also be used as the main link in a proof of the fact that every Turing computable function is partially recursive. An essential part of the proof is a new method of numbering words in a k -lettered alphabet by means of a set of $k+1$ natural numbers. This method may also be of interest in the solving of other problems.

In Appendix 1 we introduce a system of diophantine equations with three parameters a, b and c, solvable with respect to the remaining variables if and only if the parameters are in the relation $a = b^c$. This purely number-theoretic result could be included without proof in a course on the theory of algorithms; it allows the transformation of the exponential diophantine representation into a diophantine one and thus the establishment of the algorithmic unsolvability of Hilbert's 10th problem.

In Appendix 2 we show that a small modification of the proof given below allows us to obtain a singlefold exponential diophantine representation (in [9] it required a more fundamental and less obvious modification of the proof in [1] to obtain this result; some specialists even conjectured that such a modification was impossible).

In Appendix 3 we consider the question of the possibility of using, instead of Turing machines, some extensions of word equations.

2. To simplify the notation we develop the proof for the case $\mu = 2$ (i.e., the two-place predicate). It will consist of two stages. In the first stage we will construct an existential representation of the predicate $\mathcal{P}$ in which the existential quantifiers refer to words of some alphabet. In the second stage this representation will be transformed by arithmetization into the required exponential diophantine representation.

For the first step we need a definition of the predicate $\mathcal{P}$ by means of the Turing machine. At the present time in the literature the Turing machine is taken to mean various types of abstract computers; we shall choose, more or less arbitrarily, the variant used in [8].

The machine memory is represented by a tape, potentially infinite on the right. The tape is divided into squares, on which letters of the exterior alphabet $A = \{a_0, \ldots, a_m\}$ can be printed: the squares containing a_0 are considered empty (all except a finite number of the squares of the potentially infinite tape are considered to contain the letter a_0 initially upon starting). The square to the extreme left is marked with an asterisk (*), which cannot be erased or printed in another square. The machine can be in one of a finite number of states $\mathfrak{E}_0, \ldots, \mathfrak{E}_n$; the state $\mathfrak{E}_1$ is the starting state and the state $\mathfrak{E}_0$ is the final stop-state.

A machine command has the form

$$\mathfrak{E}_i \varphi \Rightarrow \varphi \mathfrak{E}_j \beta \tag{3}$$

or

$$\varepsilon_i \chi \Rightarrow \rho \varepsilon_j \Pi ,$$

where φ, ψ are letters of the alphabet A and χ, ρ either are also letters of A or $\chi \equiv \rho \equiv *$; Λ denotes movement of the head to the left, and Π to the right. A machine program is an arbitrary finite set of commands with mutually distinct left hand sides.

For technical reasons it is more convenient to deal only with words and to eliminate the head moving along the tape. For this reason we introduce the alphabet $E = \{\vec{\varepsilon}_0, \ldots, \vec{\varepsilon}_n, \overleftarrow{\varepsilon}_0, \ldots, \overleftarrow{\varepsilon}_n\}$. A configuration (instantaneous description) is a word of the form

$$*\Delta \vec{\varepsilon}_i \Xi \tag{5}$$

or of the form

$$*\Delta \overleftarrow{\varepsilon}_i \Xi , \tag{6}$$

where Δ and Ξ are words in the alphabet A . Both configurations correspond to the case where the word $\Delta\Xi$ is written on the tape and all the remaining squares are empty. In case (5), the head sees the square containing the left-most letter of the word Ξ (or, if Ξ is an empty word, then the first empty square after the word Δ). In case (6), the head sees the square containing the right-more letterof the word Δ (or, if Δ is an empty word, then the square marked with the asterisk *). Command (3) will now have two corresponding commands

$$\vec{\varepsilon}_i \varphi \Rightarrow \vec{\varepsilon}_j \psi , \tag{7}$$

$$\varphi \vec{\varepsilon}_i \Rightarrow \vec{\varepsilon}_j \psi , \tag{8}$$

and command (4) either two commands

$$\vec{\varepsilon}_i \chi \Rightarrow \rho \vec{\varepsilon}_j , \tag{9}$$

$$\chi \vec{\varepsilon}_i \Rightarrow \rho \vec{\varepsilon}_j , \tag{10}$$

or, if $\chi \equiv \rho \equiv *$, then the command (10). We can regard commands (7)-(10) as rules for transforming configurations, and consider the letters of the alphabets A and E to be equivalent. In fact for uniformity of notation, we will consider the machine program to be the list of the commands rewritten in the form of productions

$$\lambda_i \mu_i \Rightarrow \sigma_i \tau_i \quad (i = 1, \ldots, l), \tag{11}$$

where λ_i , μ_i , σ_i , τ_i are letters of the alphabet $A \cup E \cup \{*\}$.

We shall consider a two-place recursively enumerable predicate $\mathcal{P}$ to be defined by a Turing machine $\mathfrak{P}$ if the following holds: having started work in the state corresponding to the configuration $*\vec{\varepsilon}_1 a_1^{\langle a_1 \rangle} a_2^{\langle a_2 \rangle}$, the machine $\mathfrak{P}$ stops if and only if the numbers a_1 and a_2 satisfy the predicate $\mathcal{P}$ (if $\mathfrak{a}$ is a letter and c is a natural number, then $\mathfrak{a}^{\langle c \rangle}$ denotes the word $\mathfrak{a} \ldots \mathfrak{a}$, in which the letter $\mathfrak{a}$ is repeated c times).

Now suppose that the machine $\mathfrak{P}$, having started work in the condition corresponding to some configuration K , has terminated its computation after s steps. Clearly in this

case we can find words Z and W and a list of configurations $K_0, \ldots, K_\delta$ such that

I. Z contains only the letter α_0,

II. $K_0 \equiv KZ$, K_i is obtained from K_{i-1} by one of the rules (11) for $i = 1, \ldots, \delta$, $K_\delta \equiv W$,

III. W contains either $\vec{\varepsilon}_0$ or $\overleftarrow{\varepsilon}_0$.

On the other hand, if there exist words Z and W and a list of configurations $K_0, \ldots, K_\delta$ which have the three stated properties, then the machine $\mathfrak{P}$, having started in a condition corresponding to the configuration K, will stop after δ steps.

To deal with lists of arbitrary length is always awkward for arithmetization. So we shall replace the list $K_1, \ldots, K_{\delta-1}$ by one word, $L \equiv K_1 \ldots K_{\delta-1}$. The analog of property II is formulated in terms of the word L as follows.

II*. LW is obtained from KZL as a result of simultaneously replacing all occurrences of words of type $\lambda_i \mu_i$ by the words $\sigma_i \tau_i$.

(It is easy to see that any occurrence of a word of type $\lambda_i \mu_i$ in the word KZL is induced by some occurrence of this word in one of the words $K_0, \ldots, K_{\delta-1}$. A surreptitious entry could occur only as a result of some word K_j finishing in λ_i and the word K_{j+1} beginning with μ_i, which is impossible, since K_{j+1} begins with the letter *.)†

II* deals with replacements of some two-letter words by others. It is more convenient to work with replacement of letters by letters; therefore we will introduce a new alphabet $B = \{\beta_1, \ldots, \beta_\ell, \gamma_1, \ldots, \gamma_\ell\}$ and replace each rule (11) by two rules

$$\lambda_i \mu_i \Longrightarrow \beta_i \gamma_i, \tag{12}$$

$$\beta_i \gamma_i \Longrightarrow \sigma_i \tau_i \tag{13}$$

(we recall that λ_i, μ_i, σ_i, τ_i are letters of the alphabet $A \cup E \cup \{*\}$).

Let M be a word in the alphabet $A \cup B \cup \{*\}$, which is obtained from KZL by rule (12) and from which the word LW is obtained by rule (13). The latter rule (13) can be replaced by the rule or rules

$$\beta_i \Longrightarrow \sigma_i, \tag{14}$$

$$\gamma_i \Longrightarrow \tau_i. \tag{15}$$

Analogously the word KZL is obtained from M by the rules

$$\beta_i \Longrightarrow \lambda_i, \tag{16}$$

$$\gamma_i \Longrightarrow \mu_i. \tag{17}$$

We would like to deal only with rules (14)-(17) and not with rules (12)-(13). To this end we introduce the following concept. We shall say that in the word N the letter δ is the shadow of the letter $\mathfrak{x}$ if each occurrence of the letter $\mathfrak{x}$ is followed by an occurrence of the letter δ and each occurrence of the letter δ is preceded by an occurrence of the letter $\mathfrak{x}$; this relation will be denoted by $Sh(N, \mathfrak{x}, \delta)$.

†As the letter * is not permitted to occur in (7), (8), or (9), $\mu = *$ cannot occur in (11) — Translators.

We have: the word M is obtained from the word KZL by rule (12) if and only if μ_i is the shadow of λ_i in the word M, and the word KZL is obtained from M by rules (16)-(17).

We shall denote by $Sub(N, \mathfrak{x}, \delta)$ the word obtained from $\mathfrak{x}$ by replacing all occurrences of the word N by the letter δ.

Finally we have the following. The numbers a_1 and a_2 satisfy the predicate $\mathcal{P}$ if and only if there exist words K, Z, L, W, M in the alphabet $A \cup B \cup E \cup \{*\}$ such that:

a) only the letter α_0 occurs in Z,

b) $K \equiv * \bar{\varepsilon}_1 \alpha_1^{\langle a_1 \rangle} \alpha_2^{\langle a_2 \rangle}$,

c) M does not contain $\bar{\varepsilon}_0, \ldots, \bar{\varepsilon}_n, \tilde{\varepsilon}_0, \ldots, \tilde{\varepsilon}_n$,

d) $Sh(M, \beta_i, \gamma_i)$, $i = 1, \ldots, l$,

e) $KZL \equiv Sub(\ldots Sub(M, \beta_1, \lambda_1) \ldots, \gamma_l, \mu_l)$,

f) $LW \equiv Sub(\ldots Sub(M, \beta_1, \sigma_1) \ldots, \gamma_l, \tau_l)$,

g) W contains either $\bar{\varepsilon}_0$ or $\tilde{\varepsilon}_0$.

3. In order to obtain the required exponential diophantine representation of the predicate $\mathcal{P}$, we must arithmetize the predicates and functions which have been defined on words and which appear in conditions a-g. We shall consider arithmetization in a general form, independently of the preceding investigation of a Turing machine.

Let $\Gamma = \{\gamma_1, \ldots, \gamma_k\}$ be some alphabet. For each word R in the alphabet Γ we shall define its code to be the $k+1$-tuple of natural numbers $\langle r_0, \ldots, r_k \rangle$, where r_0 is the length of the word R and r_i is the number whose binary notation is obtained from the word R by replacing each occurrence of the letter γ_i by the digit 1 and all occurrences of the remaining letters by the digit 0; the code of the empty word is $\langle 0, \ldots, 0 \rangle$. Clearly a word is uniquely determined by its code.

Let $\langle r_0, \ldots, r_k \rangle$ be the code of the word R, and $\langle s_0, \ldots, s_k \rangle$ be the code of the word S. It is easy to see that:

$R \equiv S$ if and only if $r_i = s_i$, $i = 0, \ldots, k$;

the code of the word RS is the $k+1$-tuple $\langle r_0 + s_0, r_1 2^{s_0} + s_1, \ldots, r_k 2^{s_0} + s_k \rangle$;

the code of the word $Sub(R, \gamma_i, \gamma_j)$, where $i \neq j$, is the sequence $\langle r'_0, \ldots, r'_k \rangle$, where $r'_0 = r_0$, $r_i = 0$, $r'_j = r_j + r_i$, and $r'_l = r_l$ when $l \neq i \& l \neq j$;

the code of the word $\gamma_i^{\langle c \rangle}$ is the sequence $\langle c, t_1, \ldots, t_k \rangle$, where $t_i = 2^c - 1$; $t_l = 0$ for $l \neq i$;

the letter γ_j is the shadow of the letter γ_i in R if and only if $r_j = 2 r_i$;

the word R consists only of the letter γ_i if and only if $r_l = 0$ for $l \neq i \& l \neq 0$;

the word R contains either γ_i or γ_j if and only if $r_i + r_j > 0$;

the word R does not contain γ_i if and only if $r_i = 0$.

It remains for us to express the predicate "the $k+1$-tuple $\langle r_0, \ldots, r_k \rangle$ is the code of some word."

It is easy to see that this property of the sequence $\langle x_0, \ldots, x_k \rangle$ is expressed by the formula

$$\left[\bigwedge_{i=1}^{k-1} \bigwedge_{j=i+1}^{k} x_i T x_j\right] \wedge \left[x_1 + \ldots + x_k = 2^{x_0} - 1\right], \tag{18}$$

where $x T z$ is the predicate "at no binary place do x and z both have the digit 1."

From an old result of E. Kummer [12] (see also [13, Vol. 1, p. 270; 14]) it follows that

$$x T z \Longleftrightarrow \binom{x+z}{x} \equiv 1 \pmod 2. \tag{19}$$

This formula can be proved in different ways; for completeness we produce one proof based on the fact that binomial coefficients form a so-called Pascal triangle:

$$\binom{s+1}{t+1} = \binom{s}{t} + \binom{s}{t+1}. \tag{20}$$

The proof is by induction on the sum $x+z$. The cases $x=0$ and $z=0$ are trivial, since then $x T z$ and

$$\binom{z}{0} = \binom{x}{x} = 1. \tag{21}$$

So let $x>0$, $z>0$, $\bar{x} = x-1$, $\bar{z} = z-1$.

From (20) and the equality

$$\binom{s+t}{s} = \binom{s+t}{t} \tag{22}$$

it follows that

$$a = b + c, \tag{23}$$

where

$$a = \binom{x+z}{x}, \quad b = \binom{\bar{x}+z}{\bar{x}}, \quad c = \binom{x+\bar{z}}{\bar{z}}. \tag{24}$$

By induction

$$\bar{x} T z \Longleftrightarrow b \equiv 1 \pmod 2, \tag{25}$$

$$x T \bar{z} \Longleftrightarrow c \equiv 1 \pmod 2. \tag{26}$$

Let the number $\bar{x}$ end in exactly k ones, and the number $\bar{z}$ in exactly l ones ($k=0$ or $l=0$ is possible). Since $x T z \Longleftrightarrow z T x$, then without loss of generality we can suppose that $l \leq k$.

Case $l < k$.

$$\begin{array}{ll} \bar{x} & * \; * \ldots * 0 \overbrace{1 \ldots 1}^{k \text{ times}} 1 \ldots 1 \\ x & * \; * \ldots * 1 \; 0 \ldots 0 \; 0 \; 0 \ldots 0 \\ z & \Delta \; \Delta \ldots \Delta \Delta \Delta \ldots \Delta 1 \; 0 \ldots . 0 \\ \bar{z} & \Delta \; \Delta \ldots \Delta \Delta \Delta \ldots \Delta 0 \underbrace{1 \ldots 1}_{l \text{ times}} \end{array} \tag{27}$$

Here the symbols $*$ and Δ represent the unknown binary digits of the numbers $\bar{x}$, x, z, $\bar{z}$. Clearly $\neg(\bar{x} T z)$ and $x T \bar{z} \Longleftrightarrow x T z$. Thus according to (23), (25), and (26)

$$b \equiv 0 \pmod 2, \tag{28}$$

$$x T z \iff x T \bar{z} \iff c \equiv 1 \pmod 2 \iff a \equiv 1 \pmod 2. \tag{29}$$

Case $l = k$.

$$\begin{array}{ll} \bar{x} & **\ldots*\overbrace{01\ldots1}^{k \text{ times}} \\ x & **\ldots*10\ldots0 \\ z & \Delta\Delta\ldots\Delta10\ldots0 \\ \bar{z} & \Delta\Delta\ldots\Delta\underbrace{01\ldots1}_{l \text{ times}} \end{array} \tag{30}$$

Clearly $\neg(x T z)$ and $\bar{x} T z \iff x T \bar{z}$. Thus according to (25), (26), and (23)

$$b \equiv c \pmod 2, \tag{31}$$

$$a \equiv 0 \pmod 2. \tag{32}$$

Proposition 19 is proved.

It remains for us to demonstrate the exponential diophantine representation of binomial coefficients. J. Robinson first gave such a representation in [15]. Here we set forth a slightly simpler variant of this proof.

Binomial coefficients are defined as numbers for which the following equation is satisfield:

$$(u+1)^n = \sum_{i=0}^{n} \binom{n}{i} u^i \tag{33}$$

for any value of u; the variable u will be existentially quantified. We note that if

$$u > 2^n, \tag{34}$$

then

$$u > 2^n = (1+1)^n = \sum_{i=0}^{n} \binom{n}{i} \geqslant \binom{n}{i}. \tag{35}$$

Thus the binomial coefficients are defined uniquely as natural numbers satisfying (33) for at least one value of u which satisfies (34) and (35) — these are precisely the digits in the u-ary expansion of the number $(u+1)^n$. We obtain the result that $c = \binom{n}{k}$ if and only if there exist natural numbers u, w, v such that

$$u = 2^n + 1, \tag{36}$$

$$(u+1)^n = w u^{k+1} + c u^k + v, \tag{37}$$

$$v < u^k, \tag{38}$$

$$c < u. \tag{39}$$

APPENDIX 1

By introducing new variables, an arbitrary exponential diophantine equation can be transformed into an equivalent system consisting of equations of the type $a = b + c$, $a = bc$,

$\alpha = \beta^{\gamma}$, where α, β, γ are concrete natural numbers, variables of the original equation, or new additional variables. To obtain the equivalent diophantine system it is sufficient to replace each equation of the third type by a copy of the diophantine equation

$$D(a,b,c,z_1,\ldots,z_x)=0, \tag{40}$$

solvable with respect to $z_1,\ldots,z_x$ if and only if the parameters a, b, c are in the relation $a = b^c$ (for a, b, and c we substitute the corresponding natural numbers, variables from the original diophantine equation or additional variables, and for $z_1,\ldots,z_x$ we substitute eachtime new variables which have not been used earlier).

J. Robinson [15] established that to construct a diophantine equation with the required properties it is sufficient to find a diophantine predicate with exponential growth, and in [16] the author constructed an example of such a predicate. Later authors [17]-[21] constructed relativelysimple diophantine equations for the predicate $a = b^c$. The example given below is actually taken from [11]; the difference is that here we are not aiming at minimizing the number of variables. ([11] contains a method of constructing Eq. (40) with $x = 5$. This is achieved at the expense of increasing the degree of the polynomial D.)

THEOREM. The natural numbers a,b,c satisfy the relation $a=b^c$ only inthe casethere exist numbers d, f, g, h, i, j, k, l such that $Q = 0$, where Q is a polynomial definable by the following system of reductions:

$$A \rightleftharpoons 4c(a+1)+b+2, \tag{41}$$

$$B \rightleftharpoons Ab, \tag{42}$$

$$C \rightleftharpoons c+k+1, \tag{43}$$

$$D \rightleftharpoons (B^2-1)C^2+1, \tag{44}$$

$$E \rightleftharpoons 2(i+1)DC^2, \tag{45}$$

$$F \rightleftharpoons (B^2-1)E^2+1, \tag{46}$$

$$G \rightleftharpoons B+F(F-B), \tag{47}$$

$$H \rightleftharpoons 2jC+c+1, \tag{48}$$

$$I \rightleftharpoons (G^2-1)H^2+1, \tag{49}$$

$$J \rightleftharpoons DFI-d^2, \tag{50}$$

$$K \rightleftharpoons Ff-H+C, \tag{51}$$

$$L \rightleftharpoons l(A-1)+c+1, \tag{52}$$

$$M \rightleftharpoons (A^2-1)L^2-g^2+1, \tag{53}$$

$$N \rightleftharpoons (L^2-4(C-La)^2)abc-h-1, \tag{54}$$

$$P \rightleftharpoons ((a-1)^2+c)(a+b+(c-1-k)^2), \tag{55}$$

$$Q \rightleftharpoons P(J^2+K^2+M^2+N^2). \tag{56}$$

For the proof see the fourth section of [11]. We note only a change of the notation: the symbols A, B, a, b, c used in this theorem correspond to the symbols M, A, y, x, n in the paper [11].

APPENDIX 2

The exponential diophantine representation of the predicate $\mathcal{P}$ obtained by the method described above is not singlefold. However, all the arbitrariness in the choice of the values of the variables $x_1, \ldots, x_v$ in (1) is engendered by the arbitrariness in the choice of the word Z, containing only the letter a_0. The length of the word Z is the number of empty squares on the tape of the Turing machine, used for its work. After the choice of the word Z the words L, W, M are uniquely defined (the word K is already uniquely defined by the values a_1, a_2). Similarly the values of the numbers forming the codes of all the words are uniquely defined and consequently, so are the required arguments of the predicate T and, it follows, the arguments of the binomial coefficients, whichin their turn uniquely define the numbers u, w, v, appearing in (36)-(39). The values of the variables which are produced in replacing the inequalities by equalities are obviously also uniquely defined by the values of the compared numbers.

In order to eliminate the arbitrariness in the choice of Z (and thus to obtain a unique representation) we can choose for Z the shortest of all the possible words. In order to be able to write out this condition in the form of an equation, we must first modify the original Turing machine; namely, the machine must never write the symbol a_0 on the tape. For this we introduce, as its double, the symbol a_{m+1}. We replace a_0 by a_m in the right hand side of commands (3) and (4), and in addition for each command containing a_0 in the left hand side we add a command which replaces a_0 by a_{m+1}. Now the condition for minimality of Z may be written in the form of a condition on W — this word must not contain a_0.

APPENDIX 3

As a possible approach to Hilbert's tenth problem, A. A. Markov suggested the possibility of proving the undecidability of solvability of equations over a free semigroup (these equations, also called word equations, are easily reduced to diophantine equations — see for example [22]). But the question of decidability of word equations is still open and moreover, specialists working in this area conjecture the existence of a decision procedure.† However, instead of word equations it is possible to take also more complicated equations on condition that there is a suitable method of arithmetizing these equations in such a way as to get purely existential formulas, i.e., leading again to equations. One variant of such a generalization of word equations was proposed in [22].

In the present paper in the second section we actually established the algorithmic undecidability of a system of word equations with the additional operation Sub and the predicates Sh and "the word contains one (does not contain any) of the letters $p_1, \ldots, p_l$." The arithmetic representation of words described in the third section allows a further extension

†This problem was recently shown to be decidable. See G. S. Makinan, "The problem of solvability of equations in a free semigroup," Mat. Sb. (N.S.), 103(145) (1977), No. 2, 147-236, 319 M. R. 57, 9874 — Translators.

of the list of additional operations and predicates, which in general eases the problem of establishing undecidability.

In particular, V. G. Durnev [23] established the undecidability of equations in a free semigroup with additional two-place predicates of the form "the letter γ_i occurs in the words R and S an equal number of times." In order to be able to arithmetize this predicate too, it is sufficient to replace, in the definition of the code of a word, the binary system by a p-ary number system where the base p is a variable ranging over prime numbers. (Since the set of primes is diophantine, this condition may be written in the form of an existential formula.) It is also necessary to impose a supplementary condition that $r_0 < p-1$. (Thus, for a fixed p we could deal only with words of length not greater than $p-2$, but this limit is immaterial since in the final formula p will occur in an existential quantifier so that it will always be possible to choose for it a sufficiently large value.)

The analog of formula (18) will now be the formula

$$\left[\bigoplus_{i=1}^{k} r_i T_p (p-2)(p^{r_0}-1)/(p-1)\right] \& \left[\bigoplus_{i=1}^{k-1}\bigoplus_{j=i+1}^{k} (p-1) r_i T_p (p-1) r_j\right] \& \left[(p-1)(r_1+\ldots+r_k) = p^{r_0}-1\right], \tag{57}$$

where $x T_p z$ is the predicate "at no place in the p-ary expansion of the numbers x and z do there appear digits whose sum exceeds $p-1$." As before, the theorem of Kummer allows us to also express this predicate in terms of binomial coefficients:

$$x T_p z \Leftrightarrow \binom{x+z}{x} \not\equiv 0 \pmod{p}. \tag{58}$$

The letter γ_i enters the words R and S the same number of times if and only if $r_i \equiv s_i \pmod{p-1}$.

Thus, the theorem on the algorithmic undecidability of word equations with the additional predicate of the equality of the number of entries of a given letter could be the starting point of a proof of unsolvability of Hilbert's tenth problem, on condition, of course, that we have a proof of this theorem which, in contrast to the proof in [23], does not use as a starting point this unsolvability of Hilbert's tenth problem itself.

LITERATURE CITED

1. M. Davis, H. Putnam, and J. Robinson, "The decision problem for exponential diophantine equations," Ann. Math., 74, No. 3, 425-436 (1961).
2. M. Davis and H. Putnam, "On Hilbert's tenth problem," U.S. Air Force O.S.R. Report, AFOSR TR 59-124, Part III (1959).
3. J. Robinson, "The undecidability of exponential diophantine equations," Notices Am. Math. Soc., 7, 75 (1960).
4. Yu. I. Manin, "Hilbert's tenth problem," Sovrem. Prob. Mat., 1, 5-37 (1973).
5. Yu. V. Matiyasevich, "Diophantine sets," Usp. Mat. Nauk, 27, 185-222 (1972).
6. M. Davis "Hilbert's tenth problem is unsolvable," Am. Math. Monthly, 80, No. 3, 233-269 (1973).
7. M. Davis, "Arithmetical problems and recursively enumerable predicates," J. Symbol. Logic, 18, No. 1, p. 33-41 (1953).
8. Yu. Matiyasevich, "On recursive unsolvability of Hilbert's tenth problem," Proc. IV. Int. Congress Logic, Method. and Phil. of Sci. (Bucharest, 1971), North-Holland (1973), pp. 89-110.

†If we wish, formula (57) may be replaced by $\frac{(r_1+r_2+\ldots+r_k)!}{r_1!r_2!\ldots r_k!} \not\equiv 0 \pmod{p}$ and $(p-1)(r_1+r_2+\ldots+r_k) = p^{r_0}-1$ – Translators.

9. Yu. V. Matiyasevich, "The existence of noneffectizable estimates in the theory of exponential diophantine equations," Zap. Nauch. Sem. Leningr. Otd. Mat. Inst. Akad. Nauk SSSR, 40, 77-93 (1974).
10. K. Hirose and Sh. Iida, "A proof of negative answer to Hilbert's tenth problem," Proc. Jpn. Acad., 49, 10-12 (1973).
11. Yu. Matiyasevich and J. Robinson, "Reduction of an arbitrary diophantine equation to one in 13 unknowns," Acta Arithm., 27, 521-553 (1975).
12. E. E. Kummer, "Uber die Erganzungsreihe zu den algemeinen Reciprotatsgesetzen," J. Reine Angew. Math., 44, 93-146 (1852).
13. L. E. Dickson, History of the Theory of Numbers, New York (1952).
14. D. Singmaster, "Notes on binomial coefficients, I, II, III," J. London Math. Soc., 8, No. 3, 545-548, 549-554, 555-560 (1974).
15. J. Robinson, "Existential representability in arithmatic," Trans. Am. Math. Soc., 72, No. 3, 437-449 (1952).
16. Yu. V. Matiyasevich, "Enumerable sets are diophantine," Dokl. Akad. Nauk SSSR, 191, No. 2, 279-282 (1970).
17. N. K. Kosovskii, "On diophantine representation of the sequence of solutions of Pell's equation," Zap. Nauch. Sem. Leningr. Otd. Mat. Inst. Akad. Nauk SSSR, 20, 49-59 (1971).
18. G. V. Chudnovskii, "Certain arithmetical problems," Inst. Mat. Akad. Nauk Ukrains. SSR, Preprint IM-71-3.
19. Yu. V. Matiyasevich, "Diophantine representation of the set of prime numbers," Dokl. Akad. Nauk SSSR, 196, 770-773 (1971).
20. Yu. V. Matiyasevich, "Diophantine representation of enumerable predicates," Izv. Akad. Nauk SSSR, 35, Ser. Mat., 3-30 (1971).
21. M. Davis, "An explicit diophantine definition of the exponential function," Commun. Pure Appl. Math., 24, No. 2, 137-145 (1971).
22. Yu. V. Matiyasevich, "The connection between systems of equations in words and their lengths and Hilbert's tenth problem," Zap. Nauch. Sem. Leningr. Otd. Mat. Inst. Akad. Nauk SSSR, 8, 132-144 (1968).
23. V. G. Durnev, "On equations in free semigroups and groups," Mat. Zametki, 16, No. 5, 717-724 (1974).