Article

Not peer-reviewed version

There Exists a Subset of N Which Is Not Recursively Enumerable and Has a Short Description in Terms of Arithmetic

Apoloniusz Tyszka *

Posted Date: 27 February 2026

doi: 10.20944/preprints202508.0363.v8

Keywords: arithmetic of N; computable function; eventual domination; Hilbert's 10th problem; limit-computable function; recursively enumerable set; undecidable decision problem

Article

There Exists a Subset of ℕ Which Is Not Recursively Enumerable and Has a Short Description in Terms of Arithmetic

Apoloniusz Tyszka

Hugo Kołłątaj University, Balicka 116B, 30-149 Kraków, Poland; rttyszka@cyf-kr.edu.pl

Abstract

We prove that the set

$$\begin{gathered}\{n \in \mathbb{N}:\ \exists p,q \in \mathbb{N}\ ((n = 2^p \cdot 3^q) \wedge \\ \forall (x_0,\ldots,x_p) \in \mathbb{N}^{p+1}\ \exists (y_0,\ldots,y_p) \in \{0,\ldots,q\}^{p+1} \\ ((\forall k \in \{0,\ldots,p\}\ (1 = x_k \Rightarrow 1 = y_k)) \wedge \\ (\forall i,j,k \in \{0,\ldots,p\}\ (x_i + x_j = x_k \Rightarrow y_i + y_j = y_k)) \wedge \\ (\forall i,j,k \in \{0,\ldots,p\}\ (x_i \cdot x_j = x_k \Rightarrow y_i \cdot y_j = y_k))))\}\end{gathered}$$

is not recursively enumerable.

Key words and phrases: arithmetic of ℕ; computable function; eventual domination; Hilbert's 10th problem; limit-computable function; recursively enumerable set; undecidable decision problem

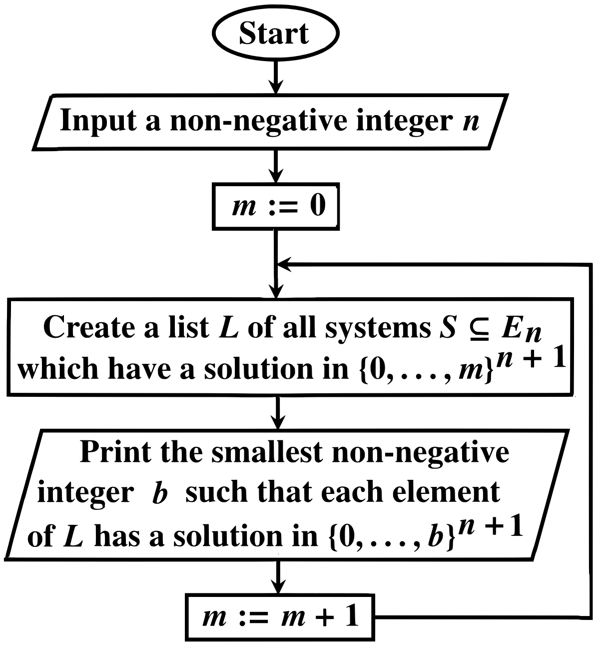
2020 Mathematics Subject Classification: 03D25

This article is a shortened version of the article [3]. For $n \in \mathbb{N}$, let

$$E_n = \{1 = x_k,\ x_i + x_j = x_k,\ x_i \cdot x_j = x_k :\ i,j,k \in \{0,\ldots,n\}\}$$

Theorem 1 ([1], p. 118)**.** *There exists a limit-computable function $f : \mathbb{N} \to \mathbb{N}$ which eventually dominates every computable function $g : \mathbb{N} \to \mathbb{N}$.*

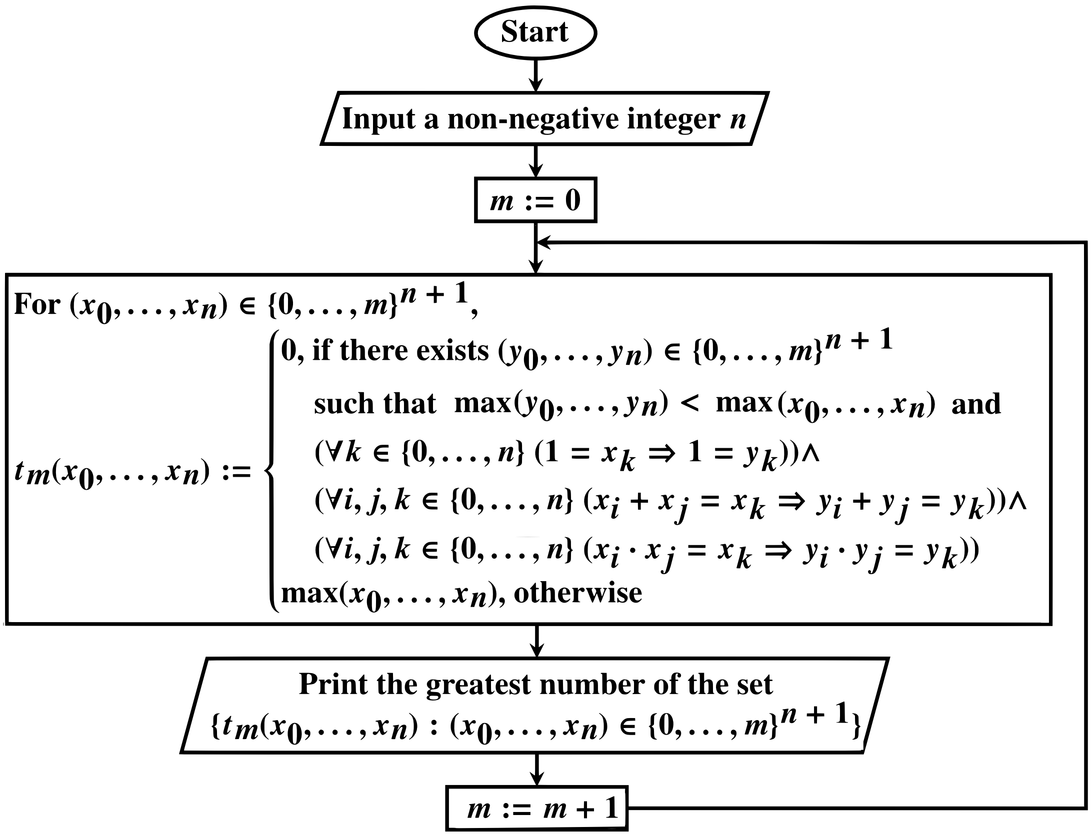
We present an alternative proof of Theorem 1. For $n \in \mathbb{N}$, $f(n)$ denotes the smallest $b \in \mathbb{N}$ such that if a system of equations $S \subseteq E_n$ has a solution in $\mathbb{N}^{n+1}$, then S has a solution in $\{0,\ldots,b\}^{n+1}$. The function $f : \mathbb{N} \to \mathbb{N}$ is computable in the limit and eventually dominates every computable function $g : \mathbb{N} \to \mathbb{N}$, see [2]. The term *"dominated"* in the title of [2] means *"eventually dominated"*. Flowchart 1 shows a semi-algorithm which computes $f(n)$ in the limit, see [2].



Flowchart 1

A semi-algorithm which computes $f(n)$ in the limit

Flowchart 2 shows a simpler semi-algorithm which computes $f(n)$ in the limit.



Flowchart 2

A simpler semi-algorithm which computes $f(n)$ in the limit

Lemma 1. *For every $n, m \in \mathbb{N}$, the number printed by Flowchart 2 does not exceed the number printed by Flowchart 1.*

Proof. For every $(a_0, \dots, a_n) \in \{0, \dots, m\}^{n+1}$,

$$E_n \supseteq \{1 = x_k : (k \in \{0, \dots, n\}) \wedge (1 = a_k)\} \cup$$

$$\{x_i + x_j = x_k : (i, j, k \in \{0, \dots, n\}) \wedge (a_i + a_j = a_k)\} \cup$$

$$\{x_i \cdot x_j = x_k : (i, j, k \in \{0, \dots, n\}) \wedge (a_i \cdot a_j = a_k)\}$$

□

Lemma 2. *For every $n, m \in \mathbb{N}$, the number printed by Flowchart 1 does not exceed the number printed by Flowchart 2.*

Proof. Let $n, m \in \mathbb{N}$. For every system of equations $S \subseteq E_n$, if $(a_0, \dots, a_n) \in \{0, \dots, m\}^{n+1}$ and $(a_0, \dots, a_n)$ solves S, then $(a_0, \dots, a_n)$ solves the following system of equations:

$$\{1 = x_k : (k \in \{0, \dots, n\}) \wedge (1 = a_k)\} \cup$$

$$\{x_i + x_j = x_k : (i, j, k \in \{0, \dots, n\}) \wedge (a_i + a_j = a_k)\} \cup$$

$$\{x_i \cdot x_j = x_k : (i, j, k \in \{0, \dots, n\}) \wedge (a_i \cdot a_j = a_k)\}$$

□

Theorem 2. *For every $n, m \in \mathbb{N}$, Flowcharts 1 and 2 print the same number.*

Proof. It follows from Lemmas 1 and 2. □

Definition 1. *An approximation of a tuple $(x_0, \dots, x_n) \in \mathbb{N}^{n+1}$ is a tuple $(y_0, \dots, y_n) \in \mathbb{N}^{n+1}$ such that*

$$(\forall k \in \{0, \dots, n\}\ (1 = x_k \Rightarrow 1 = y_k)) \wedge$$

$$(\forall i, j, k \in \{0, \dots, n\}\ (x_i + x_j = x_k \Rightarrow y_i + y_j = y_k)) \wedge$$

$$(\forall i, j, k \in \{0, \dots, n\}\ (x_i \cdot x_j = x_k \Rightarrow y_i \cdot y_j = y_k))$$

Observation 1. *For every $n \in \mathbb{N}$, there exists a set $A(n) \subseteq \mathbb{N}^{n+1}$ such that*

$$\mathrm{card}(A(n)) \leqslant 2^{\mathrm{card}(E_n)} = 2^{n+1+2\cdot(n+1)^3}$$

and every tuple $(x_0, \dots, x_n) \in \mathbb{N}^{n+1}$ possesses an approximation in $A(n)$.

Observation 2. *For every $n \in \mathbb{N}$, $f(n)$ equals the smallest $b \in \mathbb{N}$ such that every tuple $(x_0, \dots, x_n) \in \mathbb{N}^{n+1}$ possesses an approximation in $\{0, \dots, b\}^{n+1}$.*

Observation 3. *For every $n, m \in \mathbb{N}$, Flowcharts 1 and 2 print the smallest $b \in \{0, \dots, m\}$ such that every tuple $(x_0, \dots, x_n) \in \{0, \dots, m\}^{n+1}$ possesses an approximation in $\{0, \dots, b\}^{n+1}$.*

Theorem 3. *No algorithm takes as input non-negative integers n and m and decides whether or not*

$$\forall (x_0, \dots, x_n) \in \mathbb{N}^{n+1}\ \exists (y_0, \dots, y_n) \in \{0, \dots, m\}^{n+1}$$

$$((\forall k \in \{0, \dots, n\}\ (1 = x_k \Rightarrow 1 = y_k)) \wedge$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 February 2026 doi:10.20944/preprints202508.0363.v8

$$(\forall i,j,k \in \{0,\dots,n\}\ (x_i + x_j = x_k \Rightarrow y_i + y_j = y_k)) \wedge$$
$$(\forall i,j,k \in \{0,\dots,n\}\ (x_i \cdot x_j = x_k \Rightarrow y_i \cdot y_j = y_k)))$$

Proof. Since the function f is not computable, it follows from Observation 2. □

Theorem 4. *No algorithm takes as input a non-negative integer n and decides whether or not*

$$\exists p,q \in \mathbb{N}\ ((n = 2^p \cdot 3^q) \wedge$$
$$\forall (x_0,\dots,x_p) \in \mathbb{N}^{p+1}\ \exists (y_0,\dots,y_p) \in \{0,\dots,q\}^{p+1}$$
$$((\forall k \in \{0,\dots,p\}\ (1 = x_k \Rightarrow 1 = y_k)) \wedge$$
$$(\forall i,j,k \in \{0,\dots,p\}\ (x_i + x_j = x_k \Rightarrow y_i + y_j = y_k)) \wedge$$
$$(\forall i,j,k \in \{0,\dots,p\}\ (x_i \cdot x_j = x_k \Rightarrow y_i \cdot y_j = y_k))))$$

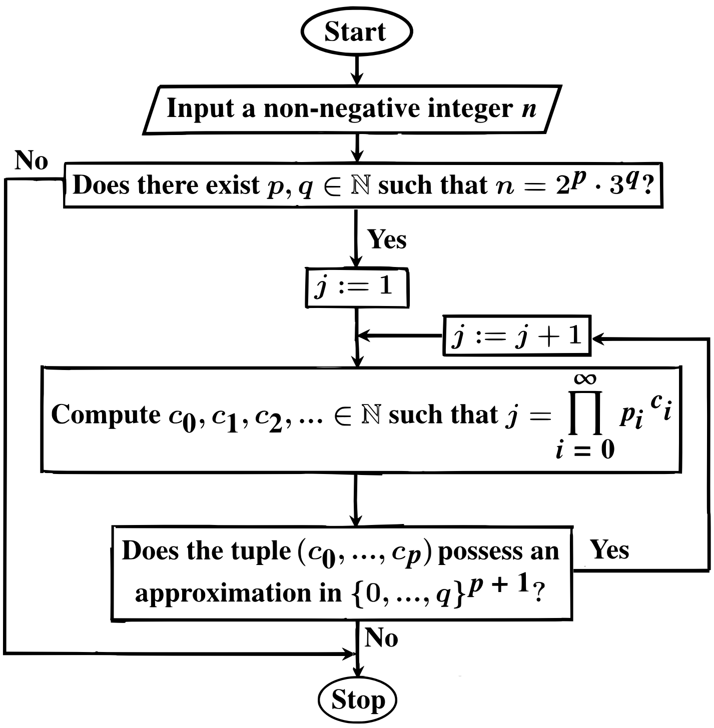
Proof. It follows from Theorem 3. □

Let

$$T = \{n \in \mathbb{N} :\ \exists p,q \in \mathbb{N}\ ((n = 2^p \cdot 3^q) \wedge$$
$$\forall (x_0,\dots,x_p) \in \mathbb{N}^{p+1}\ \exists (y_0,\dots,y_p) \in \{0,\dots,q\}^{p+1}$$
$$((\forall k \in \{0,\dots,p\}\ (1 = x_k \Rightarrow 1 = y_k)) \wedge$$
$$(\forall i,j,k \in \{0,\dots,p\}\ (x_i + x_j = x_k \Rightarrow y_i + y_j = y_k)) \wedge$$
$$(\forall i,j,k \in \{0,\dots,p\}\ (x_i \cdot x_j = x_k \Rightarrow y_i \cdot y_j = y_k))))\}$$

Theorem 5. *The set $\mathbb{N} \setminus T$ is recursively enumerable.*

Proof. For $i \in \mathbb{N}$, let p_i denote the i-th prime number. Flowchart 3 shows a semi-algorithm which takes as input $n \in \mathbb{N}$ and terminates if and only if $n \in \mathbb{N} \setminus T$.



Flowchart 3

doi:10.20944/preprints202508.0363.v8

A semi-algorithm which takes as input $n \in \mathbb{N}$ and terminates if and only if $n \in \mathbb{N} \setminus T$
□

Theorem 6. *The set T is not recursively enumerable.*

Proof. It follows from Theorems 4 and 5. □

A more sophisticated proof shows that the set

$$\{n \in \mathbb{N} : \exists p, q \in \mathbb{N} \; ((n = 2^p \cdot 3^q) \wedge$$

$$\forall (x_0, \ldots, x_p) \in \mathbb{N}^{p+1} \; \exists (y_0, \ldots, y_p) \in \{0, \ldots, q\}^{p+1}$$

$$((\forall j, k \in \{0, \ldots, p\} \; (x_j + 1 = x_k \Rightarrow y_j + 1 = y_k)) \wedge$$

$$(\forall i, j, k \in \{0, \ldots, p\} \; (x_i \cdot x_j = x_k \Rightarrow y_i \cdot y_j = y_k))))\}$$

is not recursively enumerable, see [3].

References

1. J. S. Royer and J. Case, *Subrecursive Programming Systems: Complexity and Succinctness,* Birkhäuser, Boston, 1994.
2. A. Tyszka, *All functions $g : \mathbb{N} \to \mathbb{N}$ which have a single-fold Diophantine representation are dominated by a limit-computable function $f : \mathbb{N} \setminus \{0\} \to \mathbb{N}$ which is implemented in MuPAD and whose computability is an open problem,* in: Computation, cryptography, and network security (eds. N. J. Daras, M. Th. Rassias), Springer, Cham, 2015, 577–590, https://doi.org/10.1007/978-3-319-18275-9_24.
3. A. Tyszka, *Three undecidable decision problems about a non-negative integer n which have a short description in terms of arithmetic*, https://philarchive.org/rec/TYSATA and https://ssrn.com/abstract=4710446.