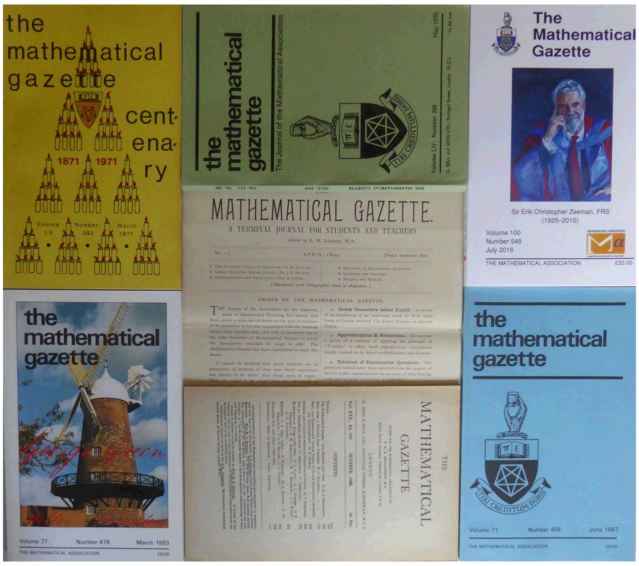
An offprint from

The Mathematical Gazette



The *Mathematical Gazette*, 100 not out

Volume 100
Number 549
November 2016

MATHEMATICAL ASSOCIATION
supporting mathematics in education

THE MATHEMATICAL ASSOCIATION £33.00

Letter Games: a metamathematical taster

A. C. PASEAU

1. *Gödel*

Metamathematics is the mathematical study of mathematics itself. Two of its most famous theorems were proved by Kurt Gödel in 1931. In a simplified form, Gödel's first incompleteness theorem states that no reasonable mathematical system can prove all the truths of mathematics. Gödel's second incompleteness theorem (also simplified) in turn states that no reasonable mathematical system can prove its own consistency. Another famous undecidability theorem is that the Continuum Hypothesis is neither provable nor refutable in standard set theory.* Many of us logicians were first attracted to the field as students because we had heard something of these results. All research mathematicians know something of them too, and have at least a rough sense of why ‘we can't prove everything we want to prove’.

The aim of this article is to give students (sixth-formers/high school seniors and juniors, or beginning undergraduates) a small sense of what metamathematics is—that is, how one might use mathematics to study mathematics itself. School or college teachers could base a classroom exercise on the letter games I shall describe and use them as a springboard for further exploration. Since I shall presuppose no knowledge of formal logic, the games are less an introduction to Gödel's theorems than an introduction to an introduction to them. Nevertheless, they show, in an accessible way, how metamathematics can be mathematically interesting.

2. *The vowel game* (*+2-variant*)

The following one-player ‘game’ is played with the letters of the English alphabet. At any stage, you can perform one of the following three actions:

(i) Action 1: immediately below the last letter to be written down (if any), write a vowel;

(ii) Action 2: immediately below the last letter to be written down, write any letter that is the double successor in the alphabet of any previous letter;

(iii) Action 3: end the game.

(The double successor of *a* is *c*, that of *b* is *d*, etc.) A simple example of a game is:

i (Action 1)

k (Action 2)

* The Continuum Hypothesis can be expressed by saying that the size of the real numbers is the next size of infinity after the smallest infinite size, that of the natural numbers; its undecidability was proved by Gödel and Paul Cohen.

m (Action 2)
End (Action 3)

The last letter of a game, if it has one, is called that game's terminal letter; in the example above, *m* is the terminal letter. A possible terminal letter is the terminal letter of a possible game. The question is: what are the possible terminal letters? Think about it before reading on.

3. *Solution*

The answer turns on a curious feature of the English alphabet. Let's number the letters in the alphabet, so that *a* is 1, *b* is 2, … and *z* is 26. The trick is to notice that the vowels occupy odd-numbered positions:

a: 1 st
e: 5 th
i: 9 th
o: 15 th
u: 21st

Since adding 2 to an odd number results in an odd number, and the number of letters in the English alphabet (26) is even, the set of possible terminal letters is precisely the set of letters in odd-numbered positions: *a*, *c*, *e*, *g*, *i*, *k*, *m*, *o*, *q*, *s*, *u*, *w*, *y*.

The description of the game was somewhat vague, in that it did not specify whether the alphabet 'wraps around', i.e. whether for the purposes of the game the successor of *z* is *a*. As we can now see, that doesn't matter: the terminal letters are the odd-numbered ones in both cases. The answer is equally unchanged if one takes *y* to be a vowel, since *y* occupies an odd-numbered position in the alphabet.*

4. *The vowel game* (*+3-variant*)

Suppose Action 2 is emended as follows:

(ii) Action 2*: immediately below the last letter to be written down, write any letter that is the *triple* successor in the alphabet of any previous letter.

What are the possible terminal letters in this variant of the game?

The answer is that all of them are, assuming that the alphabet wraps around. To see this, notice that starting from *a*, one can reach all the letters whose position in the alphabet leaves remainder 1 when divided by 3 (1, 4, 7, etc.). Once one has reached *y*, the 25 th letter of the alphabet (notice that

---

* Although in English the letter *y* is traditionally taken to be a consonant, it can in fact have a vocal value as well as a consonantal one, depending on the word it appears in. For example, the value of *y* in 'why' is vocal—the word is a homophone of 'wi' as in 'wi-fi'—whereas its value in 'yes', 'yak' or 'you' is consonantal.

25 leaves remainder 1 when divided by 3), one then reaches *b*, since *b* is *y*'s triple successor. Starting from *b*, reachable from *a* via *d*, *g*, … , *y*, one reaches all the letters whose position in the alphabet leaves remainder 2 when divided by 3 (2, 5, 8, etc.). From *z*, which is the 26 th letter of the alphabet (notice that 26 leaves remainder 2 when divided by 3), one then reaches *c*, since *c* is *z*'s triple successor. Finally, starting from *c*, one can reach all the letters whose position in the alphabet leaves remainder 0 when divided by 3 (3, 6, 9, etc.), in other words all the multiples of 3. Since any number leaves remainder 0, 1 or 2 when divided by 3, that covers all the letters. This proves that all the letters of the alphabet are possible terminal letters of the +3-variant of the game.

The +3-variant is mathematically a little more interesting than the original +2-version. It makes it clear that what underpins the solutions, as many readers will have realised, is modular arithmetic. The letters reachable from *a* in the original +2-version of the game are just the letters whose positions are of the form $1 + 2N \pmod{26}$ for non-negative N. The fact that all vowels are in position $1 + 2N \pmod{26}$ for some N explains why all and only letters whose positions are equal to $1 + 2N \pmod{26}$ are attainable. A similar explanation can be given for the +3-variant: the numbers equal to $1 + 3N \pmod{26}$ for non-negative N are all the numbers from 0 to 25 inclusive, so that everything is reachable from *a* (and a fortiori from a vowel).

5. *The consonant games*

We turn now to the consonant game, or better, games, as there are infinitely many of them. In contrast to the vowel games, the aim of these games is to reach *all* the consonants subject to the constraint that *no* vowels are reached. Different versions of this game arise from different specifications of the two actions, analogous to those of the vowel game. Suppose $\mathcal{S}$ is a set of consonants and N a non-negative integer. The $(\mathcal{S}, N)$-consonant game is the following:*

(i) Action 1: immediately below the last letter to be written down (if any), write any element of $\mathcal{S}$;

(ii) Action 2: immediately below the last letter to be written down, write any letter that is the Nth successor in the alphabet of any previous letter;

(iii) Action 3: end the game.

Terminal letters are then defined in the same way as in the vowel game. The $(\mathcal{S}, N)$-consonant game, for specific values of $\mathcal{S}$ and N, is said to be *sound* if its set of terminal letters contains no vowels, and *unsound* if this set contains

* I use capital letters as variables or constants for numbers, and script capital letters as variables or constants for sets. Letters are autonymously denoted by themselves. As earlier, the Nth successor of a letter is the letter that appears N places after it in the (cyclic) alphabet; e.g. the 3rd successor of *a* is *d* and that of *z* is *c*.

at least one vowel. It is said to be *complete* if its set of terminal letters contains all the consonants and *incomplete* if it omits at least one. The idea behind these labels is that, in contrast to the vowel game, in consonant games vowels are deemed ‘bad’, so we don't want any of them to end up as terminal letters; and consonants are deemed ‘good’, so we want as many of them as possible to be terminal letters.

Let's illustrate the definition with some examples. Start with a simple class of games, in which $\mathcal{S} = \varnothing$ (the empty set), and N is any non-negative integer. Since in these games we can never perform either Action 1 or 2—we can never get started—it's clear that the set of terminal letters is empty. Thus all the $(\varnothing, N)$-consonant games are sound but incomplete.

Next, consider the class of games for which $N = 0$. Since in these games Action 2 boils down to ‘write down any letter previously written’, the terminal letters of the $(\mathcal{S}, 0)$-consonant game are simply the elements of $\mathcal{S}$. The only $(\mathcal{S}, 0)$-game that is sound and complete is the one for which $\mathcal{S}$ is equal to $\mathcal{C}$, where $\mathcal{C}$ is the set of consonants in the English language. For if $\mathcal{S}$ omits a consonant, then the $(\mathcal{S}, 0)$-game is incomplete; and if $\mathcal{S}$ contains a vowel, then the game is unsound.

Turn finally to the case $N = 2$. When discussing the vowel game, we remarked that the vowels occupy odd-numbered positions in the alphabet, and that the alphabet consists of an even number of letters. The $(\mathcal{S}, 2)$-game is therefore unsound if $\mathcal{S}$ contains any consonant whose position in the alphabet is odd; for example if *c* is in $\mathcal{S}$, then one can reach *e* by writing down *c* (Action 1) followed by *e* (Action 2), which is a vowel. Conversely, if $\mathcal{S}$ contains no consonant whose position in the alphabet is odd, then it is sound but incomplete, since the set of terminal letters in that case cannot contain any of the odd-numbered consonants, that is to say, none of *c*, *g*, *k*, *m*, *q*, *s*, *w*, *y*. The underlying reason, once more, is that the +2-action preserves parity and that the English alphabet has an even number of letters.

A natural question now arises: for which $\mathcal{S}$ and N is the $(\mathcal{S}, N)$-game both sound and complete?

6. *A little theorem*

The answer is:

Theorem: The $(\mathcal{S}, N)$-game is sound and complete if, and only if, $\mathcal{S} = \mathcal{C}$ (the set of consonants) and N is a multiple of 26.

Those willing to take this theorem on trust can skip to the next section. For the rest, we begin the proof with a lemma:

Lemma: Let integers M and N have greatest common divisor D. If A is an integer, then every equation in unknown X of the form $NX \equiv DA \pmod{M}$ has a solution.

The proof of the lemma is easy enough if we assume the following fact: if D is the greatest common divisor of M and N then there are integers X and

Y such that $XM + YN = D$. For, multiplying both sides of the last equation by A, we see that $AXM + AYN = AD$ and hence that $AYN \equiv AD \pmod{M}$, i.e. AY solves $NX \equiv DA \pmod{M}$. The fact itself is a corollary of the Euclidean algorithm for finding the greatest common divisor of two integers; its proof may be found in any elementary textbook on number theory.

Turning to the theorem, it's clear that the set $\mathcal{T}$ of terminal letters of the $(\mathcal{S}, N)$-game consists of all and those letters equal to $S + NX \pmod{26}$ where S is the position in the alphabet of some element of $\mathcal{S}$ and X is any non-negative integer. We split the argument into four cases. Since the $(\mathcal{S}, N)$-game is equivalent to the $(\mathcal{S}, N + 26K)$-game, we need only consider values of N from 0 to 25 inclusive. Observe also that if $\mathcal{S}$ is empty then $\mathcal{T}$ is also empty, as previously argued. Thus if the $(\mathcal{S}, N)$-game is to stand a chance of being sound and complete, $\mathcal{S}$ must contain at least one letter, a fact we henceforth assume.

(i) If $N = 0$ then $\mathcal{T} = \mathcal{S}$. So if the $(\mathcal{S}, 0)$-game is to be sound and complete, $\mathcal{S}$ must be equal to $\mathcal{C}$, the set of consonants.

(ii) If N is odd and not divisible by 13, then $\mathcal{T}$ is the set of all letters of the alphabet, since the values of $NX \pmod{26}$ are all the integers from 0 to 25 inclusive (apply the Lemma with $N = N$, $M = 26$ and $D = 1$). It follows that if N is odd and not divisible by 13 then the $(\mathcal{S}, N)$-game is unsound.

(iii) If N is odd and divisible by 13, then $N = 13$. If $\mathcal{T}$ is equal to $\mathcal{C}$ then it must contain the letter *b* and hence its antipode *o* (a letter's antipode being 13 places after and before it in the cyclic alphabet), so the game cannot be both sound and complete.

(iv) If N is even and non-zero, then N is not divisible by 13 but is divisible by 2. By an application of the Lemma (with $N = N$, $M = 26$ and $D = 2$), every equation in unknown X of the form $NX \equiv 2A \pmod{26}$ has a solution, where A is an integer. And if the $(\mathcal{S}, N)$-game is to be complete then the set $\mathcal{T}$ of terminal letters must contain *c*. But if $\mathcal{T}$ contains *c*, which is the third letter of the alphabet, then it must also contain all odd-numbered letters, by what has just been argued. Hence no such game can be both sound and complete.

That completes the proof that the only sound and complete games are the ones for which $\mathcal{S} = \mathcal{C}$ and N is divisible by 26. In other words, these are the games in which one may write down any consonant and any previously written letter. These games are trivially sound and complete; and no other games are sound and complete.

Our proof used some modular arithmetic that may not be familiar to some students; but such students can still prove the theorem by efficiently running through the 26 possible values of N, for different $\mathcal{S}$.

7. *Analogy*

What do the vowel game and the consonant games have to do with metamathematics? We can draw a fairly exact metamathematical analogy between consonant games and mathematical systems.

We may think of the letters of the alphabet as mathematical statements, e.g. statements of arithmetic, or geometry, or combinatorics. Consonants may be thought of as true statements, such as $2 + 3 = 5$, and vowels as false statements, such as $2 + 3 = 7$.* The consonants in $\mathscr{S}$, which one may write down at any point, are akin to axioms: true statements one may assume at any point in the argument. Applying Action 1 is thus akin to assuming an axiom—a statement one need not argue for but may simply posit—in a mathematical argument. Applying Action 2, in contrast, is akin to applying a rule of inference: it allows us to derive a new statement from previously derived ones. A terminal letter is akin to a theorem: a statement proved using only axioms as a starting point via acceptable rules of inference. An axiom system is sound just when its theorem set consists only of true statements. This is akin to a sound consonant game, which 'proves' only consonants (true statements) and no vowels (false statements). An axiom system is complete just when its theorem set consists of all the true statements. This is akin to a complete consonant game, which 'proves' all the consonants (true statements). What we would ideally like is a sound and complete proof system, since we would like our proof system to prove all and only true statements. Finally, the mathematics used in investigating whether a particular consonant game is sound or complete is modular arithmetic. In our analogy, it plays the role of the metamathematics used to investigate the properties of mathematical proof systems, in particular their soundness and completeness.

One of the most famous proof systems is Peano Arithmetic. Named after the Italian mathematician Giuseppe Peano, Peano Arithmetic is a system containing axioms for intuitively true principles about arithmetic (such as the principle of induction) and logical rules of inferences (such as modus ponens, i.e. from '*A*' and 'if *A* then *B*' infer '*B*'').† Peano Arithmetic is a sound system in that its axioms are true statements and its rules of inference preserve truth, so its theorem set is a subset of all true statements one can make about arithmetic in the language of the system. The question is: is Peano Arithmetic complete? That is to say, as well as proving *only* true statements of arithmetic (because it is sound), does Peano Arithmetic also

* In the consonant games, that is. In the vowel game, the analogy would be reversed: the vowels would play the role of true statements. We focus on the consonant games here.

† The name 'Peano Arithmetic' is in fact a misnomer, due to Bertrand Russell, since Richard Dedekind first came up with these axioms, which Peano later investigated, with honest attribution to Dedekind. The usual English pronunciation of 'Peano Arithmetic', be it the disyllabic 'Pia-no' or the trisyllabic 'Pi-a-no' is unfaithful to the (trisyllabic) Italian pronunciation 'Pe-a-no', with 'Pe' pronounced as in 'pen' and the stress falling on the middle syllable.

prove *all* true statements? This is the question Gödel posed and answered in the negative in 1931.* As he showed, Peano Arithmetic is incomplete: it expresses some true arithmetical statements which it itself cannot prove. The true but unprovable statements generated by Gödel's methods are now called Gödel sentences. In fact, Gödel's arguments showed that such statements arise for most reasonable mathematical systems, including any consistent system that includes basic arithmetic and whose theorem set can be effectively generated.

What Gödel showed, then, is that actual mathematical systems tend to be more like the $(\mathcal{S}, N)$-games for N indivisible by 26 than the $(\mathcal{C}, 0)$-game: they are sound but incomplete. Our 'impossibility' theorem for systems with $N \neq 0 \pmod{26}$ or $\mathcal{S} \neq \mathcal{C}$ showed that, if the system is sound, it is impossible to reach all the consonants. We thus proved that in such systems there is no 'proof' of some consonant or other. As we might now put it, pursuing the metamathematical analogy: our metalanguage proof (in normal mathematics) showed that there is no object-language proof (in the game language) of some consonant. For the $(\mathcal{C}, 0)$-game, in contrast, we have a (metalanguage) proof that every consonant (truth) is reachable (provable), i.e. that every consonant is a possible terminal letter.

We summarise the parallels between consonant games and proof systems in the following table:

Consonant Game	*Mathematical System*
alphabet	statements
consonants	true statements
vowels	false statements
terminal letters (elements of $\mathcal{T}$)	theorems
the set $\mathcal{S}$	axiom set
the $+N$-Action	rule of inference
applying Action 1	assuming an axiom
applying Action 2	applying a rule of inference
modular arithmetic	metamathematics
soundness (no vowels are reachable)	soundness (no false statement is provable)
unsoundness (a vowel is reachable)	unsoundness (a false statement is provable)
completeness (all consonants are reachable)	completeness (all true statements are provable)
incompleteness (not all consonants are reachable)	incompleteness (not all true statements are provable)

But beware: the parallel is not perfect. The arguments we gave for the incompleteness of the sound $(\mathcal{S}, N)$-games and unsoundness of the complete $(\mathcal{S}, N)$-games, for $N \neq 0 \pmod{26}$ or $\mathcal{S} \neq \mathcal{C}$, were based on modular arithmetic. Gödel's genius lay in the fact that he used arithmetic itself to code a sentence G which 'says' that G itself is not provable in the system. In fact, Gödel's argument also used some facts about modular arithmetic, including the Chinese Remainder Theorem; but it also crucially used the fact

* Gödel himself was concerned with Russell and Whitehead's system in *Principia Mathematica*, but his argument generalises.

that one can use numbers to code statements of arithmetic or indeed of any other part of mathematics. It is quite clear, in contrast, that modular arithmetic is an entirely different theory from the consonant game itself; there is no obvious way in which we could use the consonant game to investigate itself. But although the consonant game does not hint at the idea of self-reference central to Gödel's argument, it does have a great many other points of analogy, as just observed.

8. *Conclusion*

The mathematics used to solve the vowel game is simple, and requires only an implicit understanding of modular arithmetic. Indeed, I was careful to explain it in a way that does not explicitly appeal to modular arithmetic. In the past, I have even used the vowel game as an Oxford admissions question (though that use has now been forfeited). Solving the consonant games requires a little more, but still fairly elementary, knowledge of modular arithmetic. These games are therefore accessible, mathematically interesting, and a bona fide introduction to metamathematics. They are a gentle introduction to the kind of thinking involved in metamathematics, requiring no acquaintance with formal logic.

Acknowlegements

Thanks to Gerry Leversha, Robin Bhattacharyya and the referee for comments.

Reference

1. K. Gödel, ‘Über formal unentscheidbare Sätze der *Principia Mathematica* und verwandter Systeme I’, *Monatshefte für Mathematik und Physik*, **38,** (1931) pp. 173-198, translated by J. van Heijenoort as ‘On formally undecidable propositions of *Principia Mathematica* and related systems I’ and reprinted in *Collected Works Vol. 1: Publications 1929-1936*, S. Feferman *et al.* eds, Oxford University Press (1986) pp. 144-195.

10.1017/mag.2016.109

A. C. PASEAU
Wadham College, Oxford
e-mail: *alexander.paseau@philosophy.ox.ac.uk*