◐ 21 ◑

The Fixed Point Principle

Two days later, Craig had another session with Professor Griffin.

"Today," said Griffin, "I wish to show you an important principle known as the *fixed point principle*, which will have many applications to various topics that I plan to discuss with you later on. A special case of this principle you already know—namely, that every bird here is fond of at least one bird. Before telling you the general principle, I think it would be helpful to consider a couple of special cases. If you can solve these special cases, I'm sure you will have no trouble grasping the fixed point principle."

• 1 •

"How do you find a bird A such that for any bird y, Ay = yA(AyA)?"

• 2 •

"How do you find a bird A such that for any birds y and z, Ayz = (z(yA))(yAz)?"

SOLUTIONS

Inspector Craig happened to be exceptionally alert that day, and he solved the two problems in a surprisingly short time.

"I can see two different ways of going about this," said

Craig. "One method uses the fact that every bird is fond of at least one bird; the other method proceeds, as it were, from scratch.

"Using the first method, here is how I solve your Problem 1. Consider the expression yx(xyx)—it is like yA(AyA) except that it has the letter x in place of the letter A. Now, by taking an x-y-eliminate of yx(xyx), we can find a bird A_1 such that for any birds x and y, $A_1xy = yx(xyx)$."

"Right, so far," said Griffin.

"Well, this bird A_1 is fond of some bird A—specifically, the bird $LA_1(LA_1)$, with L as a lark. Thus $A_1A = A$."

"Excellent!" said Griffin.

"Since $A_1A = A$," continued Craig, "then for any bird y, $A_1Ay = Ay$. But also, $A_1Ay = yA(AyA)$, because for *any* bird x, $A_1xy = yx(xyx)$. Since $A_1Ay = yA(AyA)$ and also $A_1Ay = Ay$, then $Ay = yA(AyA)$. This solves the problem."

"Great!" exclaimed Griffin. "But I am curious as to the second method you had in mind—the method that 'proceeds from scratch.' What method is that?"

"Well," replied Craig, "in the expression yx(xyx), just replace x by (xx), thus obtaining the expression y(xx)((xx)y(xx)). Then there is a bird A_2 such that for any birds x and y, $A_2xy = y(xx)((xx)y(xx))$. Then, taking A_2 for x, $A_2A_2y = y(A_2A_2)((A_2A_2)y(A_2A_2))$. And then we take for A the bird A_2A_2, and so $Ay = yA(AyA)$."

"Ah, yes!" said Griffin.

"Actually," said Craig, "I imagine the first method would, in general, yield a much shorter expression for A. The prospect of finding an x-y-eliminate of the expression y(xx)((xx)y(xx)) strikes me as pretty grim compared to finding an x-y-eliminate of the expression yx(xyx). So in practice, I think I would use the first method.

"Of course, the same method—either one, in fact—works for your second problem. To find a bird A satisfying the condition that Ayz = (z(yA))(yAz), let A_1 be an x-y-z-eliminate of the expression (z(yx))(yxz), and let A be the bird $LA_1(LA_1)$. Then A_1A = A, so A_1Ayz = Ayz, so Ayz = A_1Ayz = (z(yz))(yAz), and A is the desired bird.

"The same method would work for any expression with four variables instead of three. For example, take the expression x(zwy)(xxw). If we let A_1 be an x-y-z-w-eliminate of this expression and let A be the bird $LA_1(LA_1)$, then for any birds y, z, w, Ayzw = A(zwy)(AAw). Indeed, the same method would work for any expression with *any* number of variables. Is this the principle you call the *fixed point* principle?"

"You have the idea," said Griffin. "To state the fixed point principle in its most general form, suppose we take any number of variables x, y, z . . . and write down any equation of the form Axyz . . . = (———), where (———) is any expression built from these variables and the letter A. For example, (———) might be the expression yA(wAA)(xAz). The fixed point principle is that the equation can always be solved for A—in other words, there is a bird A such that for any birds x, y, z . . . it is true that Axyz . . . = (———). In the above example, there is a bird A such that for any birds x, y, z, w it is true that Axyzw = yA(wAA)(xAz). You will see the importance of this principle when we come to the study of arithmetical birds.

"I might remark," added Griffin, "that the existence of a sage bird is only a special case of the fixed point principle—the case where (———) is the expression x(Ax). By the fixed point principle, there is then a bird A such that for every bird x, Ax = x(Ax)—such a bird A is a sage bird."

"That's interesting!" said Craig. "I hadn't seen a sage bird in that light before."

The following exercises should give the reader further insight into the uses of the fixed point principle.

Exercise 1 (Sage birds revisited): Let us look again at the problem of finding a sage bird A—only now from the point of view of the fixed point principle.

We are to find a bird A satisfying the equation Ax = x(Ax)—for all birds x. In this chapter, we have seen two different methods of solving such an equation. Try both methods and see what birds you get. Both of them have been encountered in Chapter 13.

Exercise 2 (Commuting birds revisited): Using both methods, find a bird A such that for every bird x, Ax = xA. Such a bird A *commutes* with every bird x (recall Problem 18, Chapter 11). One of the solutions will be the same as that of Problem 18, Chapter 11; the other solution will be new. What new solution do you get?

Exercise 3: In each case, find a bird A satisfying the given requirement. (Better use the first method.)

a. Ax = Axx
b. Ax = A(xx)
c. Ax = AA(xx)

Exercise 4: Find a bird A such that for every bird x, Ax = AA.

Exercise 5: In each case, find a bird A satisfying the given requirement.

a. Axy = xyA
b. Axy = Ayx
c. Axy = x(Ay)

Exercise 6: By the fixed point principle, there is a bird A such that for any birds x, a, and b, Axab = x(Aaab)(Abab). Using this fact, prove the following theorem (known as the *double fixed point theorem*): For any birds a and b there are birds c and d such that acd = c and bcd = d. This constitutes a new and quite simple proof of the double fixed point theorem.

SOLUTIONS

Ex. 1: Using the first method, we must first find a bird A_1 such that for all x and y, $A_1yx = x(yx)$, and any bird of which A_1 is fond will be a solution. Well, the owl O is such a bird A_1, and LO(LO)—or any other bird of which O is fond—is a sage bird. We thus get the same solution as we got in Problem 14, Chapter 13.

Using the second method, we must first find a bird A_1 such that for all x and y, $A_1yx = x(yyx)$, and then A_1A_1 will be a solution. Well, the Turing bird U is such a bird A_1, and so we see again that our old friend UU is a sage bird.

Ex. 2: Using the first method, you should get LT(LT)—or any other bird of whom the thrush T is fond—as a solution. This is the same as Problem 18, Chapter 11.

Using the second method, you should get the solution W′W′, where W′ is the converse warbler—W′xy = yxx. If you get CW(CW) you are also right, since CW is a converse warbler. You can easily check that W′W′x = x(W′W′).

Ex. 3:

a. LW(LW)

b. LL(LL)

c. L(LL)(L(LL))

Ex. 4: Some of you may have been stumped by this, since A is the only letter on the righthand side of the equation. However, either method still works; we will use the first.

We must first find a bird A_1, such that for every x and y, $A_1yx = yy$. Well, BKM is such a bird, as you can easily check, and so L(BKM)(L(BKM)) is a solution.

Ex. 5:

a. LR(LR)
b. LC(LC)
c. LQ(LQ)

Ex. 6: For *any* birds x, a, and b, Axab = x(Aaab)(Abab). Therefore, if we take a for x, we see that Aaab = a(Aaab)(Abab). If, instead, we take b for x, we see that Abab = b(Aaab)(Abab). Therefore, if we let c = Aaab and d = Abab, we see that c = acd and d = bcd.

◐ 22 ◑

A Glimpse into Infinity

SOME FACTS ABOUT THE KESTREL

"You know," said Griffin to Craig, in another of their daily chats, "despite the fact that Professor Bravura dislikes the 'lowly' kestrel, this bird has some interesting properties."

• 1 •

"For example," continued Professor Griffin, "suppose we have a bird forest in which there are at least two birds. You know that a kestrel cannot be fond of itself?"

"I remember that," replied Craig. He was thinking of Problem 19, Chapter 9.

"Did you know that if the forest contains at least two birds, then it is impossible for a kestrel to be fond of an identity bird?"

"I never thought about that," said Craig.

"The proof is quite easy," remarked Griffin.

What is the proof?

• 2 •

"I hate these silly forests having only one bird," said Griffin. "In all the problems I will give you today, I am making the underlying assumption that the forest has at least two birds.

"Prove that if K is a kestrel and I is an identity bird, then I $\neq$ K—in other words, no bird can be both an identity bird and a kestrel."

• 3 •

"Another thing," said Griffin: "No starling can be fond of a kestrel. Can you see why this is so?"

• 4 •

"It follows from this," continued Griffin, "that no starling can also be an identity bird. Can you see why?"

• 5 •

"I see now," said Craig, "that no bird can be both a starling and an identity bird and no bird can be both a kestrel and an identity bird. Is it possible for a bird to be both a starling and a kestrel?"

"Good question!" said Griffin. "The answer is not difficult to figure out."

What is the answer? Remember, we are assuming that the forest contains at least two birds.

• 6 •

"Here is a simple but important principle," said Griffin. "You have already agreed that no kestrel K can be fond of itself. This means that $KK \neq K$. This fact can be generalized: For *no* bird x is it the case that $Kx = K$! Can you prove this?"

Note: It will be helpful to the reader to recall the cancellation law for kestrels, which we proved in Chapter 9, Problem 16—namely, that if $Kx = Ky$, then $x = y$.

• 7 •

"Another fact," said Griffin: "We have proved that a kestrel K cannot be fond of an identity bird I. This means that $KI \neq I$. This fact can also be generalized: Prove that there cannot be any bird x such that $Kx = I$."

. . .

"Well," said Griffin, "I will soon tell you an extremely important fact about kestrels. But first, how about a nice cup of tea?"

"Capital idea!" said Craig.

SOME NONEGOCENTRIC BIRDS

While Craig and Griffin are taking time out for tea, let me tell you about some other nonegocentric birds. We shall assume that the forest contains the birds K and I and that K $\neq$ I. On this basis, we have already proved that the kestrel K cannot be egocentric; recall that by an egocentric bird is meant a bird x such that xx = x. Many other birds can also be proved nonegocentric. We shall look at a few.

• 8 •

Prove that no bird can be both a kestrel and a thrush.

• 9 •

Now prove that no thrush T can be egocentric.

• 10 •

Prove that if R is a robin, then RII $\neq$ I. It can also be proved, by the way, that RI $\neq$ I and that R $\neq$ I. The reader might try these as exercises.

• 11 •

Now prove that no robin R can be egocentric.

• 12 •

Prove that no cardinal C can be egocentric.

• 13 •

Prove that no vireo V can be egocentric. [Vxyz = zxy]

• 14 •

Show that for any warbler W:

a. W is not fond of I.

b. W is not egocentric.

• 15 •

Show that for any starling S:

a. SI is not fond of I. It can also be shown that S is not fond of I.

b. S is not egocentric.

• 16 •

Prove that for any bluebird B:

a. BKK $\neq$ KK

b. B cannot be egocentric.

• 17 •

Can a queer bird Q be egocentric?

The reader might have fun looking at some other familiar birds and seeing which ones can be shown to be nonegocentric. The reader might also find it a good exercise to show that of the birds B, C, W, S, R, and T, no pair can be identical—i.e., B $\neq$ C, B $\neq$ W, . . . , B $\neq$ T, C $\neq$ W, . . . , C $\neq$ T, etc.

KESTRELS AND INFINITY

"Well," said Griffin, after they had had a delicious tea, complete with buttered crumpets, "some of the little problems I have given you about kestrels lead to a highly significant fact. Again, we consider a forest having at least two birds. Did you know that if the forest contains a kestrel K, then it must contain *infinitely* many birds?"

"That sounds most interesting!" exclaimed Craig.

"Some of my former students have given me fallacious proofs of this fact," said Griffin. "I recall that when I told this to one student, he instantly replied: 'Oh, of course! Just consider the infinite series K, KK, KKK, KKKK, . . .'

"You see why this proof is fallacious?"

• 18 •

Why is this proof fallacious?

• 19 •

"Of course I see why the proof is fallacious," replied Craig. "However, suppose we instead take the series K, KK, K(KK), K(K(KK)), K(K(K(KK))), . . . Will that work?"

"You got it!" said Griffin.

"To tell you the truth, that was only a *guess*," replied Craig. "I haven't really verified in my mind that all these birds are really different. For example, how do I know that K(KK) isn't really the same bird as K(K(K(K(K(KK)))))?"

"I'll give you a hint," replied Griffin. "To simplify the notation, let K_1 be the bird K; let $K_2 = KK_1$, which is KK; let $K_3 = KK_2$, which is K(KK); let $K_4 = KK_3$, which is K(K(KK)), and so forth. Thus for each number n, $K_{n+1} = KK_n$. The problem is to show that for two different numbers n and m, it cannot be that $K_n = K_m$. For example, $K_3 = K_8$ cannot hold; $K_5 = K_{17}$ cannot hold. First recall the cancellation law for kestrels: If $Kx = Ky$, then $x = y$. Then divide your proof into three steps:

Step 1: Show that for any n greater than 1, $K_1 \neq K_n$—that is, K_1 cannot be any of the birds K_2, K_3, K_4, . . .

Step 2: Show that for any numbers n and m, if $K_{n+1} = K_{m+1}$, then $K_n = K_m$. For example, if K_4 were equal to K_7, then K_3 would have to be equal to K_6.

Step 3: Using Step 1 and Step 2, show that for no two *distinct* numbers m and n can it be the case that $K_n = K_m$, and therefore there really are infinitely many birds in the sequence $K_1, K_2, K_3, \ldots$"

With these hints, Craig solved the problem. What is the solution?

SOLUTIONS

1 • Suppose a kestrel K is fond of an identity bird I. Then KI = I. Therefore, for any bird x, KIx = Ix, and since Ix = x, then KIx = x. Also KIx = I, since K is a kestrel. This means that KIx is equal to both x and I, hence x = I. Therefore, if K is fond of I, then *every* bird x is equal to I and hence I is the only bird in the forest. But we are given that there are at least two birds in the forest; hence K cannot be fond of I.

2 • This follows from the last problem. Suppose K = I. Then KI = II, hence KI = I. This means that K is fond of I, which, according to the last problem, cannot happen.

3 • Suppose SK = K. Then for any birds x and y, SKxy = Kxy. Hence SKxy = x, since Kxy = x. Also, SKxy = Ky(xy) = y. Therefore SKxy is equal to both x and y, hence x and y are equal. So, if SK = K, then any birds x and y are equal, which means that there is only one bird in the forest.

4 • Suppose S = I. Then SK = IK = K, hence SK = K. But SK ≠ K, as we showed in the last problem; therefore S ≠ I.

5 • Suppose S = K. Then SIKI = KIKI. Now, SIKI = II(KI) = I(KI) = KI, whereas KIKI = II = I. Therefore, if S = K, then KI = I. But KI ≠ I, by Problem 1; hence S ≠ K.

6 • Suppose there were a bird x such that Kx = K. Then for every bird y, it would follow that Kxy = Ky, and hence that x = Ky. Then for any birds y_1 and y_2, it would follow that $Ky_1 = Ky_2$, because x is equal to each of them. Then by the cancellation law—Problem 16, Chapter 9—it would follow that $y_1 = y_2$. And so the assumption that there is a bird x such that Kx = K leads to the conclusion that for any birds y_1 and y_2, the bird y_1 is equal to y_2—in other words, that there is only one bird in the forest!

7 • This is easier. Suppose there is a bird x such that Kx = I. Then KxI = II, hence x = I, since KxI = x and II = I. Then, since Kx = I and x = I, it follows that KI = I. But KI ≠ I, according to Problem 1. Therefore, there is no bird x such that Kx = I.

8 • Suppose T = K. Then TIK = KIK, hence KI = KIK = I, but KI ≠ I, according to Problem 7.

9 • For this and the next several problems, I will make the solutions more condensed. By now, the reader should have enough experience to fill in any missing steps. I will illustrate what I mean by "missing steps" in the solution to this problem.

Suppose TT = T. Then TTKI = TKI, hence KTI = IK. *Missing steps*: "because TTKI = KTI and TKI = IK." Therefore T = K. *Missing steps*: "because KTI = T and IK = K." But T ≠ K according to Problem 8. Therefore it cannot be that TT = T.

10 • Suppose RII = I. Then RIIK = IK, hence IKI = K by simplifying both sides of the equation, hence KI = K, contrary to Problem 7. In Problem 7 we proved that there is *no* bird x such that Kx = K, so in particular, KI ≠ K.

11 • Suppose RR = R. Then RRII = RII. Now, RRII = IIR = R, so R = RII, hence RII = RIIII = IIII = I. We then have RII = I, contrary to the last problem.

12 • Suppose CC = C. Then CCIKI = CIKI = IIK = K. Also, CCIKI = CKII = KII = I. We then have I = K, contrary to Problem 2.

13 • Suppose VV = V. Then VVIII = VIII = III = I. Also VVIII = IVII = VII, and so VII = I. Then VIIK = IK = K. Also VIIK = KII = I, and so we have K = I, contrary to Problem 2.

14 • a. Suppose WI = I. Then WIK = IK = K. Then IKK = K, hence KK = K, which we know is not so; no kestrel is egocentric.

b. Suppose WW = W. Then WWI = WI. Now, WWI = WII = III = I. Hence we would have WI = I, contrary to part a of the problem.

15 • a. Suppose SI were fond of I. Then SII = I. Then SIIK = IK, hence IK(IK) = IK, so KK = K. But KK ≠ K, so SII ≠ I.

b. Suppose SS = S. Then SSIII = SIII = II(II) = I. Also SSIII = SI(SI)I = II(SII) = SII. Hence we have SII = I, contrary to part a of the problem.

16 • a. Suppose BKK = KK. Then BKKI = KKI, hence K(KI) = K. This is again contrary to Problem 6, which states that there is *no* bird x such that Kx = K.

b. Suppose BB = B. Then BBIK = BIK, hence B(IK) = BIK. Therefore BK = BIK. Therefore BKK = BIKK = I(KK) = KK and we have BKK = KK, contrary to part a of the problem.

17 • Suppose QQ = Q. Then QQIKI = QIKI = K(II) = KI. Also, QQIKI = I(QK)I = QKI. Hence QKI = KI. Then QKII = KII, so I(KI) = I, hence KI = I, contrary to Problem 1. Therefore, Q, queer as it may be, is definitely *not* egocentric.

18 • The fallacy is that all the infinitely many of the expressions of the series name only two different birds—namely K and KK. Clearly KKK = K, hence KKKKK = KKK = K, and indeed all the expressions with an odd number of K's boil down to K; all those with an even number of K's boil down to KK.

19 • The series Craig named really works!

Step 1: We proved in Problem 6 that for every bird x, K $\neq$ Kx. Hence K cannot be any of the birds KK_1, KK_2, KK_3, . . . Thus K_1 is not any of the birds K_2, K_3, K_4 . . .

Step 2: Suppose, for example, that $K_3 = K_{10}$. Then $KK_2 = KK_9$, hence by the cancellation law for kestrels, $K_2 = K_9$.

Of course the proof works for any numbers n and m: If $K_{n+1} = K_{m+1}$ then $KK_n = KK_m$, and so $K_n = K_m$.

Step 3: Suppose, for example, that $K_4 = K_{10}$. Then by successively applying Step 2, we would have $K_3 = K_9$, $K_2 = K_8$, $K_1 = K_7$, violating Step 1.

Obviously the proof works for any two distinct numbers.

◐ 23 ◑
Logical Birds

"I am very proud of this forest," said Professor Griffin one day. "Some of the birds here can do very clever things. For example, did you know that some of them can do propositional logic?"

"I am not sure I understand what you mean by that," replied Griffin.

"Let me first explain some of the basics of propositional logic," said Griffin. "To begin with, I am using *Aristotelian* logic, according to which every proposition p is either true or false but not both. We use the symbol t to stand for *truth* and f to stand for *falsehood*. And so the value of any proposition p is either t or f—t if p is true and f if p is false. Now, logicians have a way of constructing more complex propositions out of simpler ones. For example, given any proposition p, there is the proposition *not p*—symbolized ~p—which is false when p is true and true when p is false. This is simply schematized: ~t = f; ~f = t. It is usually displayed as the following table, called the *truth table* for negation:

Negation

p	~p
t	f
f	t

"Next, given any propositions p and q, we can form their *conjunction*—the proposition that p and q are both true. This proposition is symbolized p & q. It is true when p and q are both true, and false otherwise. In other words, t & t = t; t & f = f; f & t = f; and f & f = f. These four conditions are tabulated by the following table—the so-called truth table for conjunction:

Conjunction

p	q	p & q
t	t	t
t	f	f
f	t	f
f	f	f

"Also, given propositions p and q, we can form the proposition p v q, which is read 'p or q, or maybe both' and is called the *disjunction* of p and q. This proposition is true if *at least* one of the propositions p and q is true; otherwise it is false. The disjunction operation has the following truth table:

Disjunction

p	q	p v q
t	t	t
t	f	t
f	t	t
f	f	f

"As you see, the proposition p v q is false only in the last of the four possible cases—the case when p and q both have the value f.

"Next, from propositions p and q we can form the so-called *conditional* proposition p → q, which is read 'if p, then q,' or 'p implies q.' The proposition p → q is taken to be *true* if either p is false or p and q are both true. The only case when p → q is false is when p is true or q is false. Here is the truth table for p → q:

Conditional

p	q	p → q
t	t	t
t	f	f
f	t	t
f	f	t

"Since p → q is true when and only when p is false or p and q are both true, it can also be written: (~p) v (p & q). It can be written even more simply as (~p) v q, or as ~(p & ~q).

"Finally, given any propositions p and q, there is the proposition p ↔ q, which is read 'p if and only if q,' which asserts that p implies q and q implies p. This proposition is true just in the case that p and q both have the value t or both have the value f.

Equivalence

p	q	p ↔ q
t	t	t
t	f	f
f	t	f
f	f	t

"These five symbols— ~ (not), & (and), v (or), → (if—then), ↔ (if and only if)—are called *logical connectives.* Using them, one can form from simple propositions propositions of any complexity. For example, we can form the proposition p & (q v r), which is true if and only if p is true and also at least one of q and r is true. Or we could form the very different proposition (p & q) v r, which is true just in case either p and q are both true, or r is true. One can easily compute their truth values, given the truth values of p, q, and r, by combining the tables for & and v. Of course, since there are now three variables involved—p, q, and r—we now have eight possibilities instead of four. Here is the truth table for (p & q) v r.

p	q	r	(p & q)	(p & q)v r
t	t	t	t	t
t	t	f	t	t
t	f	t	f	t
t	f	f	f	f
f	t	t	f	t
f	t	f	f	f
f	f	t	f	t
f	f	f	f	f

"On the other hand, here is the truth table for p & (q v r).

p	q	r	q v r	p & (q v r)
t	t	t	t	t
t	t	f	t	t
t	f	t	t	t
t	f	f	f	f
f	t	t	t	f
f	t	f	t	f
f	f	t	t	f
f	f	f	f	f

"You see, the two propositions have different truth tables," said Griffin.

"I understand all this," said Craig, "but how does it relate to the birds?"

"I am coming to that," replied Griffin. "To begin with, I have chosen for t and f two *particular* birds. The first, t, represents *truth*, or it can be thought of as being the representative of all true propositions. The second bird, f, of course, represents *falsehood*, or is the representative of all false propositions. I call t the *bird of truth*, or the *truth bird*, or more briefly, just *truth*. I call f the *falsehood bird*, or the *bird of falsehood*, or more briefly, just *falsehood*."

"What birds are they?" asked Craig.

"For t, I take the kestrel K; for f, I take the bird KI. And so, when we are discussing propositional logic, I use t synonymously with K and f synonymously with KI."

"Why this particular choice?" asked Craig. "It seems quite arbitrary!"

"Oh, there are many other choices that would work," replied Griffin, "but this particular one is technically convenient. I have adopted this idea from the logician Henk Barendregt. I will tell you the technical advantage in a moment.

"The birds t and f are collectively called *propositional birds*. Thus, there are only two propositional birds—t and f. From now on, I shall use the letters p, q, r, and s as standing for arbitrary *propositional birds*, rather than propositions. I call p *true* if p is t and *false* if p = f. Thus t is called *true* and f is called *false*.

"Now, the advantage of Barendregt's scheme is this:

"For any birds x and y, whether propositional birds or not, txy = x, since Kxy = x, and fxy = y, since fxy = KIxy = Iy = y. And so for any *propositional* bird p, pxy is x if p is true, and pxy is y if p is false. In particular, if p, q, and r are all propositional birds, then pqr = (p & q) v (~p & r)—or what is the same thing, pqr = (p → q) & (~p → r). This can be read 'if p then q; otherwise r.'"

"You still haven't told me what you mean when you say that some of the birds here can *do* propositional logic," said Craig. "Just what do you mean by this?"

"I was just coming to that!" replied Griffin. "What I mean is that for any simple or compound truth table, there is a bird here that can compute that table."

• 1 •

"For example, there is a bird N—called the *negation* bird—that can compute the truth table for negation. That is, if you call t to N, N will respond by naming f; if you call f to N, N will respond by naming t. Thus Nt = f and Nf = t. In other words, for any propositional bird p, Np is the bird ~p. The first problem I want you to try is to find a negation bird N."

• 2 •

"Then we have a *conjunction bird* c such that for any propositional birds p and q, cpq = p & q. In other words, ctt = t; ctf = f; cft = f; and cff = f. Can you find a conjunction bird c?"

• 3 •

"Now find a *disjunction* bird d—a bird such that for any propositional birds p and q, dpq = p v q. In other words, dtt = t; dtf = t; dft = t; but dff = f. Can you find such a bird d?"

• 4 •

"Then there is the *if-then* bird—a bird i such that itt = t; itf = f; ift = t; and iff = t. In other words, ipq = p → q. Can you find an if-then bird i?"

• 5 •

"Now find the *if-and-only-if* bird e—also called an *equivalence* bird—such that for any propositional birds p, q, epq = (p ↔ q). In other words, ett = t; etf = f; eft = f; and eff = t."

SOLUTIONS

1 · Since the Master Forest is combinatorially complete, we can find a bird N such that for all x, Nx = xft. Specifically, we can take N to be Vft, where V is the vireo. Then Vftx = xft. So Nt = tft = f; Nf = fft = t. Thus N is a negation bird.

2 · Consider c such that for any x and y, cxy = xyf. *Note:* We can take c to be Rf, where R is the robin. Then Rfxy = xyf.

1. ctt = ttf = t
2. ctf = tff = f
3. cft = ftf = f
4. cff = fff = f

Thus c is a conjunction bird.

3 · Take d such that for all x and y, dxy = xty. We can specifically take d to be Tt, where T is the thrush. Then Ttxy = xty. The reader can verify that d is a disjunction bird by working out the four cases.

4 · Take i such that ixy = xyt. We can take i to be Rt, where R is the robin. The reader can verify that this bird i works.

5 · Take e to be such that for all x and y, exy = xy(Ny). We can take e to be CSN, where C is the cardinal, S is the starling, and N is the negation bird. The reader can easily verify that epq = p ↔ q.

◐ 24 ◑

Birds That Can Do Arithmetic

In this episode and the next, Craig found out the true wonders of Griffin's forest.

Shortly before his departure, Craig visited Griffin in his study one late-summer day. The weather was beautiful, and all the windows of the study were open. Craig was quite surprised to see several birds perched on the windowsills engaged in lively conversation with Professor Griffin—all in bird language, of course. As the birds already there left, others would come.

"Ah yes!" said Griffin, after the last bird had departed. "I have been testing some of my arithmetical birds. Did you know that some of the birds here can do arithmetic?"

"Will you please explain that?" asked Craig.

"Well, I'd better start at the beginning," replied Griffin. "We will work with the natural number series 0, 1, 2, 3, 4 . . . When I use the word 'number' I will always mean either 0 or one of the positive whole numbers. These numbers are called *natural* numbers. By the *successor* n^+ of a number n, I mean n + 1. Thus $0^+ = 1$; $1^+ = 2$; $2^+ = 3$, and so forth.

"Now each number n is represented by some bird; I use the notation $\bar{n}$ to mean the bird that represents n. Thus n is a *number*; $\bar{n}$ is a bird—the bird that represents the number n. In the scheme I am about to show you for representing numbers by birds, the vireo V plays a major role: We will let σ be the bird Vf—which is V(KI)—and we will call σ the *successor* bird. For $\bar{0}$, we take the identity bird I. We take $\bar{1}$ to be the bird

$\sigma\bar{0}$; $\bar{2}$ to be $\sigma\bar{1}$; $\bar{3}$ to be $\sigma\bar{2}$, and so forth. Hence $\bar{0} = I$; $\bar{1} = \sigma\bar{0}$; $\bar{2} = \sigma(\sigma\bar{0})$; $\bar{3} = \sigma(\sigma(\sigma\bar{0}))$, and so forth. Thus $\bar{0} = I$; $\bar{1} = VfI$; $\bar{2} = Vf(VfI)$; $\bar{3} = Vf(Vf(VfI))$, and so on."

"Again, this choice strikes me as arbitrary," said Craig. "What's so special about the bird Vf?"

"You will see that shortly," replied Griffin. "Actually there are many other possible choices. The first numerical scheme was proposed by Alonzo Church. The scheme I am using has several technical advantages over Church's; the combinatorial logician Henk Barendregt is responsible for it. Anyway, I want to start explaining to you how birds here do arithmetic. First for some preliminaries:

"The birds $\bar{0}$, $\bar{1}$, $\bar{2}$, $\bar{3}$, and so on I call *numerical* birds—these are identified with the respective numbers 0, 1, 2, 3, . . . Now, if I call out a numerical bird $\bar{n}$ to a bird A, A doesn't necessarily respond by calling back a *numerical* bird; it might call back a nonnumerical bird. Well, a bird A is said to be an *arithmetical bird of type 1* if for every numerical bird $\bar{n}$, the bird $A\bar{n}$ is also a numerical bird. Loosely speaking, this means that A operating on a number gives you a number. A bird A is called an *arithmetical bird of type 2* if for any numbers n and m, the bird $A\bar{n}\bar{m}$ is a numerical bird. Equivalently, A is a numerical bird of type 2 if for every number n, the bird $A\bar{n}$ is an arithmetical bird of type 1. Similarly, we define arithmetical birds of types 3, 4, 5, and so on. Thus, for example, if A is an arithmetical bird of type 4, then for any numbers a, b, c, and d, the bird $A\bar{a}\bar{b}\bar{c}\bar{d}$ is a numerical bird.

"Now come some interesting things. There is a bird here called the *addition* bird, symbolized by $\oplus$, such that for any numbers m and n, $\oplus\bar{m}\bar{n}$ is the sum of m and n—or rather, the numerical bird representing that sum. That is, $\oplus\bar{m}\bar{n} = \overline{m + n}$. Thus, for example, $\oplus\bar{2}\,\bar{3} = \bar{5}$; $\oplus\bar{3}\,\bar{9} = \overline{12}$.

"Then we have a bird $\otimes$ called a *multiplication* bird such that for any numbers n and m, $\otimes\bar{n}\bar{m}$ is the bird $\overline{n \cdot m}$. So, for example, $\otimes\bar{2}\,\bar{5} = \overline{10}$; $\otimes\bar{3}\,\bar{7} = \overline{21}$.

"We also have an *exponentiating* bird Ⓔ such that for any numbers n and m, Ⓔ$\bar{n}\bar{m} = \bar{k}$, where k is the number n^m—the result of multiplying n by itself m times. So, for example, Ⓔ$\bar{5}\,\bar{2} = \overline{25}$; Ⓔ$\bar{2}\,\bar{5} = \overline{32}$; Ⓔ$\bar{2}\,\bar{3} = \bar{8}$; Ⓔ$\bar{3}\,\bar{2} = \bar{9}$.

"Having these birds," continued Griffin," we can easily combine them to form any arithmetical combination we want. For example, we can find a bird A such that for any numbers a, b, and c, $A\bar{a}\bar{b}\bar{c} = \bar{d}$, where d, say, is $(3a^2b + 4ca)^5 + 7$.

"In fact," continued Griffin, in growing excitement, "given *any* numerical operation that can be performed by one of these modern electronic computers, there is a bird here that can perform the same operation! For any computer, there is a bird here that can match it!

"Do you realize what this means?" asked Griffin, waxing more excited still. "It means that the birds here could totally take over the job of the computers. Maybe one day the computers of the world will one by one be replaced by birds until there are no computers left—only birds! Wouldn't that be a beautiful world?"

Craig thought this idea somewhat visionary, but intriguing, nevertheless.

"All this sounds most interesting," said Craig, "but I am in the dark as to how you find even the basic arithmetic birds that add, multiply, and exponentiate. What birds are they?"

"I am coming to that," replied Griffin, "but first for some preliminaries."

• 1 •

"To begin with," said Griffin, "we should be sure that the birds $\bar{0}$, $\bar{1}$, $\bar{2}$, $\bar{3}$, . . . are all distinct—that is, for any numbers n and m, if $n \neq m$, meaning n is unequal to m, then the bird $\bar{n}$ is distinct from the bird $\bar{m}$. Can you see how to prove this?"

2 • The Predecessor Bird P

"For any *positive* number n," said Griffin, "by its *predecessor* n^- is meant the next lower number. That is, for any positive n, n^- is the number $n - 1$. Of course, for any number n, the number n^+ is positive and the predecessor of n^+ is n.

"What we now need," said Griffin, "is a bird that calculates predecessors. That is, we want a bird P such that for any number n, $P\overline{n+} = \bar{n}$. Can you see how to find such a bird P?"

• 3 •

"We recall the propositional birds t and f. We now need a bird Z called a *Zero-tester* such that if $\bar{0}$ is called out to Z, you will get the response t—meaning, 'True, the number you called is 0'—whereas if you call out any number other than 0, you will get the response f—meaning, 'False, the number is not 0.' That is, we want a bird Z such that $Z\bar{0} = t$, but for any *positive* number n, $Z\bar{n} = f$. Can you find such a bird Z?"

• 4 •

"Let me ask you a question," said Griffin. "Do you have any reason to believe that there is a bird A such that for any number n and any birds x and y, if $n = 0$, then $A\bar{n}xy = x$, but if n is positive, $A\bar{n}xy = y$? That is, is there a bird A such that $A\bar{0}xy = x$; $A\bar{1}xy = y$; $A\bar{2}xy = y$; $A\bar{3}xy = y$; and so forth?"

"Oh, of course!" replied Craig, after a moment's thought.

How did Craig realize this?

"And now," said Griffin, "we come to some of the more interesting birds. Before we consider the problem of finding an addition bird, let us consider a slightly simpler problem. Let us take any particular number—say 5. How can we find a bird A that adds 5 to any number that you call to it? That is, we want a bird A such that $A\bar{0} = \bar{5}$; $A\bar{1} = \bar{6}$; $A\bar{2} = \bar{7}$—and for any number n, $A\bar{n} = \overline{n + 5}$."

Craig thought about this, but could not find a solution.

"The idea is based on a principle known as the *recursion* principle," said Griffin. "Suppose A is a bird such that the following two conditions hold:

1. $A\bar{0} = \bar{5}$
2. For every number n, $A\overline{n+} = \sigma(A\bar{n})$.

"Do you see that such a bird A would do the required job?"

"Let us see now," said Craig. "It is given that $A\bar{0} = \bar{5}$. What about $A\bar{1}$? Well, by the second condition, $A\bar{1} = \sigma(A\bar{0}) = \sigma\bar{5}$, since $A\bar{0} = \bar{5}$, and $\sigma\bar{5} = \bar{6}$. Therefore $A\bar{1} = \bar{6}$. Now that we know that $A\bar{1} = \bar{6}$, it follows that $A\bar{2} = \bar{7}$, because $A\bar{2} = \sigma(A\bar{1}) = \sigma\bar{6} = \bar{7}$. Yes, of course I see why it is that for every number n, $A\bar{n} = \overline{n + 5}$. We successively prove $A\bar{0} = \bar{5}$, $A\bar{1} = \bar{6}$, $A\bar{2} = \bar{7}$, $A\bar{3} = \bar{8}$, and so forth!"

"Good!" said Griffin. "You have grasped the recursion principle."

"I am still in the dark, though, about how one finds a bird A satisfying those conditions," said Craig. "How does one?"

"Ah, that's the clever part," said Griffin with a smile. "It is based on the fixed point principle, which I have already explained to you."

"Really!" said Craig in amazement. "I can't see any connection between the two!"

"I will now explain," said Griffin. "First of all, do you see that Condition 2 can be alternately described as follows?

2′. For every number n greater than 0, $A\bar{n} = \sigma(A(P\bar{n}))$."

"Yes," said Craig, "because for any number n greater than zero, n = m+, where m is the predecessor of n. Therefore Condition 2′ says that $A\overline{m+} = \sigma(A(P\overline{m+}))$, but since $P\overline{m+} = \bar{m}$, then Condition 2′ simply says that $A\overline{m+} = \sigma(A(P\overline{m+}))$, or what is the same thing, $A\bar{n} = \sigma(A(P\bar{n}))$. But, of course, this holds only when n is positive."

"Good!" said Griffin. "And so you see that what we want is a bird A such that $A\bar{n} = \bar{5}$ if $n = 0$, and $A\bar{n} = \sigma(A(P\bar{n}))$ if $n \neq 0$."

"I see that," said Craig.

"Well, we use the zero-tester Z," said Griffin. "The bird $Z\bar{n}\bar{5}(\sigma(A(P\bar{n})))$ is $\bar{5}$ if $n = 0$, and is $\sigma(A(P\bar{n}))$ if $n \neq 0$, and so we want a bird A such that for every number n, $A\bar{n} = Z\bar{n}\bar{5}(\sigma(A(P\bar{n})))$. Well, by the fixed point principle, there *is* such a bird A—in fact, there is a bird A such that for *any* bird x, whether a numerical bird or not, $Ax = Zx\bar{5}(\sigma(A(Px)))$. That solves the problem!

"In case you have forgotten," added Griffin, "we can obtain the bird A by first taking a bird A_1 such that for any birds x and y, $A_1yx = Zx\bar{5}(\sigma(y(Px)))$, and then you can take for A any bird of which A_1 is fond—for example, we can take $LA_1(LA_1)$ for A."

"That is indeed clever!" said Craig in genuine admiration. "Who thought of it?"

"The idea of using the fixed point principle to solve problems like this is attributable to Alan Turing—the same logician who discovered the Turing bird. Turing has done some extremely clever things!"

• 5 •

"Of course," said Griffin, "the number 5 has no special significance; I could have taken, say, 7, and asked for a bird A such that for all n, $A\bar{n} = \overline{n + 7}$. However, we want something better than that. We want an arithmetic bird ⊕ of type 2 such that for *any* two numbers n and m, $\oplus\bar{m}\bar{n} = \overline{m + n}$. Only a slight modification of what I have shown you is necessary. Can you see how to find such a bird ⊕?"

• 6 •

"Next, can you see how to find a bird ⊗ such that for any numbers n and m, $\otimes\bar{n}\bar{m} = \overline{n \cdot m}$? Of course, you are free to use the bird ⊕ that you have just found."

• 7 •

"Now that we have the birds ⊕ and ⊗, can you find an exponentiating bird Ⓔ such that for any numbers n and m, Ⓔ$\bar{n}\bar{m}$ $= \bar{k}$, where $k = n^m$?"

PREPARATION FOR THE FINALE

"I understand you must leave this forest in a couple of days. Is that correct?" asked Griffin.

"Alas, yes!" replied Craig. "I have been called back home on a strange case involving a bat and a Norwegian maid."

"That *does* sound strange!" remarked Griffin. "At any rate, tomorrow I would like to tell you one of the most interesting facts of all about this forest. This fact is related to Gödel's famous incompleteness theorem, as well as to some results discovered by Church and Turing. But today, I must give you the necessary background. I must tell you more about arithmetical birds as well as something about property birds and relational birds."

"What are *they*?" asked Craig.

• 8 •

"Well, by a *property* bird is meant a bird A such that for any number n, the bird $A\bar{n}$ is a propositional bird—one of the two birds t or f. A set S of numbers is said to be *computable* if there is a property bird A such that $A\bar{n} = t$ for every n in the set S, and $A\bar{n} = f$ for every n not in the set S. Such a bird A is said to *compute* the set S. And a set S is called *computable* if there is a bird A that computes it.

"The nice thing about a computable set S is that given any number n, you can find out whether n belongs to the set or whether it doesn't; you just go over to the bird A, which computes S, and call out $\bar{n}$. If A responds by calling out t, you know that n is in the set S; if A calls out f, you know that n is not in the set S.

"As an example, the set E of all even numbers is computable—there is a bird A such that $A\bar{0} = t$; $a\bar{1} = f$; $A\bar{2} = t$; $A\bar{3} = f$; and for *every* even number n, $A\bar{n} = t$, whereas for *every* odd number n, $A\bar{n} = f$. Can you see how to find A? You might try using the fixed point principle."

9 • The Bird g

"By a *relational* bird—or more properly, a relational bird of degree 2—is meant a bird A such that for any numbers a and b, $A\bar{a}\bar{b} = t$ or $A\bar{a}\bar{b} = f$.

"You are probably familiar with the symbol >, meaning 'greater than,'" continued Griffin. "For any numbers a and b, we write $a > b$ to mean that a is greater than b—so, for example, $8 > 5$ is true; $4 > 9$ is false; also $4 > 4$ is false. We now need a relational bird that computes the relation 'is greater than'—that is, we need a bird g such that for any numbers a and b, if $a > b$, then $g\bar{a}\bar{b} = t$, but if $a \leq b$, meaning that a is less than or equal to b, then $g\bar{a}\bar{b} = f$. Can you see how to find such a bird?

"This is a bit tricky," Griffin added, "so I had best point out the following facts. The relation $a > b$ is the one and only relation satisfying the following conditions, for any numbers a and b:

1. If $a = 0$, then $a > b$ is false.
2. If $a \neq 0$, then:
   a. If $b = 0$, then $a > b$ is true.

b. If $b \neq 0$, then $a > b$ is true if and only if $(a - 1) > (b - 1)$.

"Now, using the fixed point principle, do you see how to find the bird g?"

10 • The Minimization Principle

"Now comes an important principle known as the *minimization* principle," said Griffin.

"Suppose that A is a relational bird such that for every number n, there is at least one number m such that $A\bar{n}\bar{m} = t$. Such a relational bird is sometimes called *regular*. If A is regular, then for every number n there is obviously the *smallest* number k such that $A\bar{n}\bar{k} = t$. Well, the minimization principle is that given any regular relational bird A, there is a bird A′, called a *minimizer* of A, such that for every number n, $A'\bar{n} = \bar{k}$, when k is the *smallest* number such that $A\bar{n}\bar{k} = t$. So, for example, if $A\bar{n}\bar{0} = f$ and $A\bar{n}\bar{1} = f$ and $A\bar{n}\bar{2} = f$, but $A\bar{n}\bar{3} = t$, then $A'\bar{n} = \bar{3}$. Can you see how to prove the minimization principle?"

Craig thought about this for some time.

"I'd better give you some hints," said Griffin. "Given a regular bird A, first show how to find a bird A_1 such that for all numbers n and m, the following two conditions hold:

1. If $A\bar{n}\bar{m} = f$, then $A_1\bar{n}\bar{m} = A_1\bar{n}\overline{m^+}$.
2. If $A\bar{n}\bar{m} = t$, then $A_1\bar{n}\bar{m} = \bar{m}$.

"Then take A′ to be $CA_1\bar{0}$, where C is the cardinal, and show that A′ is a minimizer of A."

11 • The Length Measurer

"By the *length* of a number n," said Griffin, "we mean the number of digits in n, when n is written in ordinary base 10 notation. Thus the numbers from 0 to 9 have length 1; those from 11 to 99 have length 2; those from 100 to 999 have length 3, and so forth.

"We now need a bird ℓ that measures the length of any number—that is, we want ℓ to be such that for any number n, $\ell\bar{n} = \bar{k}$, when k is the length of n. So, for example, $\ell\bar{7} = \bar{1}$; $\ell\overline{59} = \bar{2}$; $\ell\overline{648} = \bar{3}$. Can you see how to find the bird ℓ?"

Craig thought about this for some time. "Ah!" he finally said. "I get the idea! The length of a number n is simply the smallest number k such that $10^k > n$."

"Good!" said Griffin.

With this, the reader should have no trouble finding the bird ℓ.

12 • Concatenation to the Base 10

"Now for the last problem of today," said Griffin. "For any numbers a and b, by a∗b we mean the number which, when written in base 10 notation, consists of a in base 10 notation, followed by b in base 10 notation. For example, 53∗796 = 53796; 280∗31 = 28031."

"That's a curious operation on numbers!" said Craig.

"It is an important one, as you will see tomorrow," replied Griffin. "This operation is known as *concatenation to the base 10.* And now we need a bird ⊛ that computes this operation—that is, we want ⊛ to be such that for any numbers a and b, $⊛\bar{a}\bar{b} = \overline{a*b}$. So you see how to find such a bird?"

SOLUTIONS

1 • We first show that $\bar{0}$ is different from all the birds $\bar{1}$, $\bar{2}$, $\bar{3}$, . . . $\overline{n+}$, . . .

Well, suppose there were some number n such that $\bar{0} = \overline{n+}$. Then I = V$f\bar{n}$. Then IK = V$f\bar{n}$K = K$f\bar{n}$ = f. Hence we would have K = KI, since IK = K and f = KI, but we already know that K ≠ KI. Therefore $\bar{0} \neq \overline{n+}$.

We must next show that for any numbers n and m, if $\overline{m+} = \overline{n+}$, then m = n. Well, suppose that $\overline{n+} = \overline{m+}$. Then

$Vf\bar{n} = Vf\bar{m}$. Hence $Vf\bar{n}f = Vf\bar{m}f$, so $ff\bar{n} = ff\bar{m}$, hence $\bar{n} = \bar{m}$, since $ff\bar{n} = \bar{n}$ and $ff\bar{m} = \bar{m}$.

Now that we know that $\bar{0} \neq \overline{m+}$ and that for every n and m, if $\overline{n+} = \overline{m+}$ then $n = m$, the proof that all the birds $\bar{0}$, $\bar{1}, \bar{2}, \ldots, \bar{n}, \ldots$ are distinct proceeds exactly as in the solution to Problem 19, Chapter 22.

2 · Take P to be Tf, where T is the thrush and f is the bird KI, as in the last chapter. Then for any number n, $P\overline{n+} = Tf\overline{n+} = \overline{n+}f = Vf\bar{n}f = ff\bar{n} = \bar{n}$.

3 · Take Z to be Tt; T is the thrush, and t is the truth bird K. Then:

1. $Z\bar{0} = TtI = It = t$. So $Z\bar{0} = t$.
2. Now take any number n. Then $Z\overline{n+} = Tt\overline{n+} = \overline{n+}t = Vf\bar{n}t = tf\bar{n} = f$.

Note: Under the particular scheme used by Griffin for representing numbers by birds, the birds σ, P, and Z are relatively easy to find. This is the technical advantage to which Griffin referred. Any other scheme that would yield a successor bird, a predecessor bird, and a zero-tester would also work.

4 · The zero-tester Z is such a bird A! *Reason:* $Z\bar{0}xy = txy$, since $Z\bar{0} = t$, and $txy = x$, so $Z\bar{0}xy = x$. But for any $n \neq 0$, $Z\bar{n} = f$, hence $Z\bar{n}xy = fxy = y$.

5 · The addition operation + is uniquely determined by the following two conditions, for any numbers n and m:

1. $n + 0 = n$
2. $n + m^{+} = (n + m)^{+}$. That is, n plus the successor of m is the successor of n + m.

We therefore seek a bird A such that for all n and m:

1. $A\bar{n}\bar{0} = \bar{n}$
2. $A\bar{n}\,\overline{m+} = \sigma(A\bar{n}\bar{m})$, or what is the same thing, for any positive m, $A\bar{n}\bar{m} = \sigma(A\bar{n}(P\bar{m}))$.

Thus A must satisfy the condition that for any n and any m, whether 0 or positive, $A\bar{n}\bar{m} = Z\bar{m}\bar{n}(\sigma(A\bar{n}(P\bar{m})))$. Such a bird A exists by the fixed point principle, and so we take ⊕ to be any such bird A.

6 • We note that multiplication is the one and only operation satisfying the following two conditions:

1. For any number n, $n \cdot 0 = 0$.
2. For any numbers n and m, $n \cdot m^+ = (n \cdot m) + n$.

We therefore want a bird A such that for every n and m, $A\bar{n}\bar{m} = (Z\bar{m})\bar{0}((\oplus)(A(\bar{n}(P\bar{m}))\bar{n}))$. Again, such a bird A can be found by the fixed point principle and we take ⊗ to be such a bird.

7 • The exponential operation obeys the following well-known laws:

1. $n^0 = 1$
2. $n^{m+} = n^m \times n$

We therefore seek a bird Ⓔ such that for all n, $Ⓔ\bar{n}\bar{0} = \bar{1}$, and for any positive number m, $Ⓔ\bar{n}\bar{m} = \otimes(Ⓔ\bar{n}\bar{m})\bar{n}$. Equivalently we want a bird Ⓔ such that for all n and m, $Ⓔ\bar{n}\bar{m} = Z\bar{m}\bar{1}(\otimes(Ⓔ\bar{n}\bar{m})\bar{n})$. Again, such a bird Ⓔ can be found by the fixed point principle.

8 • The property of being an even number is the one and only property satisfying the following two conditions:

1. 0 is even.
2. For any positive number n, n is even if and only if its predecessor is *not* even.

We therefore seek a bird A such that:

1. $A\bar{0} = t$
2. For any positive n, $A\bar{n} = N(A(P\bar{n}))$, where N is the negation bird.

We thus want a bird A such that for every n, whether positive or 0, $A\bar{n} = Z\bar{n}t(N(A(P\bar{n})))$. Again such a bird A exists by the fixed point principle.

9 · By virtue of the conditions given, we seek a bird g such that for any numbers a and b:

1. If $Z\bar{a} = t$, then $g\bar{a}\bar{b} = f$.
2. If $Z\bar{a} = f$, then:
   a. If $Z\bar{b} = t$, then $g\bar{a}\bar{b} = t$.
   b. If $Z\bar{b} = f$, then $g\bar{a}\bar{b} = g(P\bar{a})(P\bar{b})$.

Equivalently, we want a bird g such that for all numbers a and b, the following holds:

$$g\bar{a}\bar{b} = Z\bar{a}f(Z\bar{b}t(g(P\bar{a})(P\bar{b})))$$

Again such a bird g exists by the fixed point principle.

10 · Suppose A is a regular relational bird. By the fixed point principle there is a bird A_1 such that for all birds x and y, $A_1xy = (Axy)y(A_1x(\sigma y))$. Then for any numbers n and m, $A_1\bar{n}\bar{m} = A(\bar{n}\bar{m})\bar{m}(A_1\bar{n}\overline{m+})$. Thus Condition 1 and Condition 2 hold, because the value of $(A\bar{n}\bar{m})\bar{m}(A_1\bar{n}\overline{m+})$ is $\bar{m}$, if $A\bar{n}\bar{m} = t$, and is $A_1\bar{n}\overline{m+}$, if $A\bar{n}\bar{m} = f$.

Following Griffin's suggestion, we assume $A' = CA_1\bar{0}$. Then for every n, $A'\bar{n} = A_1\bar{n}\bar{0}$ (because $A'\bar{n} = CA_1\bar{0}\bar{n} = A_1\bar{n}\bar{0}$). Now, given an n, let k be the smallest number such that $A\bar{n}\bar{k} = t$. For example, suppose $k = 3$. Then $A\bar{n}\bar{0} = f$; $A\bar{n}\bar{1} = f$; $A\bar{n}\bar{2} = f$; but $A\bar{n}\bar{3} = t$. We must show that $A'\bar{n} = \bar{3}$—in other words, that $A_1\bar{n}\bar{0} = \bar{3}$. Well, since $A\bar{n}\bar{0} = f$, then $A_1\bar{n}\bar{0} = A_1\bar{n}\bar{1}$, by Condition 1. Since $A\bar{n}\bar{1} = f$, then $A_1\bar{n}\bar{1} = A_1\bar{n}\bar{2}$, again by Condition 1. Since $A\bar{n}\bar{2} = f$, then $A_1\bar{n}\bar{2} = A_1\bar{n}\bar{3}$, again by Condition 1. But now $A\bar{n}\bar{3} = t$, hence $A_1\bar{n}\bar{3} = \bar{3}$, by Condition 2. And so $A_1\bar{n}\bar{0} = A_1\bar{n}\bar{1} = A_1\bar{n}\bar{2} = A_1\bar{n}\bar{3} = \bar{3}$, therefore $A_1\bar{n}\bar{0} = \bar{3}$, and so $A'\bar{n} = \bar{3}$.

We illustrated the proof for $k = 3$, but the reader can readily see that the same type of proof would work if k were any other number.

11 · A single example should convince the reader of the correctness of Craig's assertions:

Suppose n = 647. The length of 647 is 3, and $10^3 = 1000$, which is greater than 647. But $10^2 = 100$, which is less than 647. Perhaps we should also consider the case when n itself is a power of 10—suppose n, say, is 100. Then $10^3 > 100$, but 10^2, though not less than 100, is not greater than 100; it is equal to 100. So 3 is the smallest number such that $10^3 > 100$.

Now to find the bird ℓ: We let A_1 be the bird $Bg(Ⓔ\overline{10})$, where B is the bluebird. Then for any numbers n and m, $Bg(Ⓔ\overline{10})\bar{n}\bar{m} = g(Ⓔ\overline{10}\bar{n})\bar{m} = g\overline{10}\,\bar{n}\bar{m}$, which is t if $10^n > m$, and is f otherwise. And so A_1 is a relational bird such that $A_1\bar{n}\bar{m} = t$ if and only if $10^n > m$. We then let A be the bird CA_1, where C is the cardinal. Then $A\bar{n}\bar{m} = A_1\bar{m}\bar{n}$, and so $A\bar{n}\bar{m} = t$ if $10^m > n$; otherwise $A\bar{n}\bar{m} = f$. Finally, we take ℓ to be a minimizer of A, and so $\ell\bar{n}$ is the *smallest* m such that $10^m > n$—in other words, $\ell\bar{n} = \bar{k}$, where k is the length of n.

12 • We first illustrate the general idea with an example. Suppose a = 572 and b = 39. Then $572*39 = 57239 = 57200 + 39 = 572.10^2 + 39$, and 2 is the length of 39.

In general, $a*b = a \cdot 10^k + b$, where k is the length of b. We accordingly take ⊛ to be a bird such that for all x and y, $⊛xy = ⊕(⊗x(Ⓔ\overline{10}(\ell y)))y$. As the reader can easily verify, for any numbers a and b, $⊛\bar{a}\bar{b} = \overline{a \cdot 10^k + b}$, where k is the length of b.

◐ 25 ◑
Is There an Ideal Bird?

"Tomorrow I unfortunately must leave," said Craig, "but before I do, I want to tell you of a problem I have been unable to solve. Perhaps you may know the answer.

"Any expression X, built from the symbols S and K, and parenthesized correctly, is the name of some bird. Now, two different expressions might happen to name the same bird—for example, the expressions ((SK)K)K and KK(KK) are both names of the kestrel K, though the expressions themselves are different. Now, what I want to know is this: Given two expressions X_1 and X_2, is there any systematic way of determining whether or not they name the same bird?"

"A beautiful question!" replied Griffin. "And it's an amazing coincidence that you ask it today. This was just the topic I was planning to talk to you about. This question has been on the minds of some of the world's ablest logicians and has come to be known as the *Grand Question*.

"To begin with, any question as to whether two expressions name the same bird can be translated into a question of whether a certain number belongs to a certain set of numbers."

"How is that?" asked Craig.

"This is done using a clever device attributable to Kurt Gödel—the device known as *Gödel numbering*, which I will shortly explain.

"All the birds here are derivable from S and K, and their behavior—the way a bird x responds to a bird y—is strictly determined by the rules of combinatory logic. Combinatory logic is a theory that can be completely formalized. The theory uses just five symbols:

S	K	(	)	=
1	2	3	4	5

"Under each symbol I have written the number called its *Gödel* number, but I'll tell you more about Gödel numbering a bit later.

"Any expression built from the two letters S and K and parenthesized correctly is called a *term*. To be more precise, a term is any expression in the first four symbols that is constructed by the following two rules:

1. The letters S and K standing alone are terms.
2. Given any terms X and Y already constructed, we may form the new term (XY).

"In application to this bird forest, the *terms* are those expressions that are names of birds. The letter S is the name of one particular starling—which one doesn't really matter—and the letter K is the name of one particular kestrel.

"By a *sentence* is meant an expression of the form X = Y, when X and Y are terms. The sentence X = Y is called *true* if X and Y are names of the same bird and *false* otherwise. In order for a sentence X = Y to be true, the term X doesn't have to be the same as the term Y; it merely suffices that these terms name the same bird.

"Of course, for any terms X, Y, and Z, the sentence SXYZ = XZ(YZ) is true, by definition of the starling, and KXY = X is true, by definition of the kestrel. All such sentences are taken as *axioms* of combinatory logic. We also take as axioms all sentences of the form X = X; these sentences are trivially true. These are the only axioms we shall take. We then *prove* various sentences to be true by starting with the axioms and using the usual logical rules for equality, which are:

1. If we can prove X = Y, then we can conclude that Y = X.

2. If we can prove X = Y and Y = Z, then we can conclude that X = Z.

3. If we can prove X = Y, then for any term Z we can conclude that XZ = YZ and that ZX = YX.

"Now, when I said that the behavior of the birds of this forest is completely determined by the laws of combinatory logic, what I meant is that a sentence X = Y is true, in the sense that the terms X and Y name the same bird, if and only if the sentence X = Y is *provable* from the above axioms by the rules I have just mentioned. There are no 'accidental' relations between our birds; X = Y only if the fact is *provable.*

"This system of combinatory logic is known to be consistent, in the sense that not every sentence is provable—in particular, the sentence KI = K is not provable. If this one sentence were provable, then every sentence would be provable by essentially the same argument we used to show that if KI = K, then there could be only one bird in the forest. We will use f to abbreviate KI, and we will also use t synonymously with K, and so the sentence f = t is an important example of a sentence that is not provable in the system.

"And now for Gödel numbering: I have already told you that the Gödel numbers of the five symbols S, K, (,), and = are respectively 1, 2, 3, 4, and 5. The Gödel number of any compound expression is obtained by simply replacing each symbol with the digit representing its Gödel number and then reading off the resulting string of digits to the base 10. For example, the expression (SK) consists of the third symbol, followed by the first symbol, followed by the second symbol, followed by the fourth symbol, and so its Gödel number is 3124—three thousand one hundred twenty-four.

"Now, let $\mathcal{T}$ be the set of Gödel numbers of the true sentences. Given any terms X and Y, they name the same bird if and only if the sentence X = Y is true, and the sentence is true if and only if its Gödel number lies in the set $\mathcal{T}$. That's what I meant when I said that any question of whether or not two terms X and Y name the same bird can be translated into a question of whether a certain number—namely, the Gödel number of the sentence X = Y—lies in a certain set of numbers—namely, the set $\mathcal{T}$.

"Now, the question *you* are asking boils down to this: Is the set $\mathcal{T}$ a computable set? Is there some purely deterministic device that can compute which numbers are in $\mathcal{T}$ and which ones are not? As I have told you, anything a computer can do can be done by one of our birds, and so your question is equivalent to this: Is there here some 'ideal' bird A that can evaluate the truth of all sentences of combinatory logic? Is there a bird A such that whenever you call out the Gödel number of a true sentence, the bird will call back "t," and whenever you call out any other number, the bird will call back "f"? In other words, is there a bird A such that for every n in $\mathcal{T}$, $A\bar{n} = t$ and for every n not in $\mathcal{T}$, $A\bar{n} = f$? That is the question you are asking. Such a bird could settle *all* formal mathematical questions, because all such questions can be reduced to questions of which sentences are provable in combinatory logic and which sentences are not. Combinatory logic is a *universal* system for all of formal mathematics, and so any ideal bird might be said to be mathematically omniscient. That is why so many people have come to this forest in search of this bird."

"That is staggering to the imagination!" said Craig. "Is it yet known whether or not there is this 'ideal' bird?"

"The question, in one form or another, has been on the minds of many mathematicians and philosophers from Leibniz on—and possibly earlier. It can be equivalently formulated: Can there be a *universal* computer that can settle all mathematical questions? Thanks to the work of Gödel, Church, Turing, Post, and others, the answer to this question is now known once and for all. I won't spoil it for you by telling you the answer yet, but before this day is over, you will know the answer.

"We did a good deal of the preliminary work yesterday when we derived the concatenation bird ⊛, but there are still a few preliminaries left before we can answer the Grand Question.

"You realize, of course, that for any expressions X and Y, if a is the Gödel number of X and b is the Gödel number of Y, then the Gödel number of XY is a∗b. For example, suppose X is the expression S and Y is the expression K. The Gödel number of X is 31 and the Gödel number of Y is 24. The expression XY is (SK) and its Gödel number is 3124, which is 31∗24. Now you can see the significance of the numerical operation of concatenation to the base 10."

1 • Numerals

"By a *numeral* is meant any of the terms $\bar{0}$, $\bar{1}$, $\bar{2}$, . . . , $\bar{n}$, . . . We call $\bar{n}$ the *numeral* for the number n. The term $\bar{n}$, like any other term, has a Gödel number; we let $n^{\#}$ be the Gödel number of the numeral $\bar{n}$.

"For example, $\bar{0}$ is I, which in terms of S and K is the expression ((SK)K); this expression has Gödel number 3312424. And so $0^{\#} = 3312424$.

"As for $1^{\#}$, this is already quite a large number: $\bar{1}$ is the expression $\sigma\bar{0}$, where σ is the expression (Vf), which in terms of S and K can be seen to be the expression (S(K(S(S((SK)K)(K(K((SK)K))))))K)—a horrible expression whose Gödel number is 3132313133124243232331242444444424. To avoid having to write this number again, I will henceforth represent it by the letter s. Thus s is the Gödel number of σ. Then, since $\bar{1}$ is the expression $(\sigma\bar{0})$, the number $1^{\#}$ is $3{*}s{*}0^{\#}{*}4$. Then $2^{\#} = 3{*}s{*}1^{\#}{*}4$; $3^{\#} = 3{*}s{*}2^{\#}{*}4$, and so forth. For each number n, $(n + 1)^{\#} = 3{*}s{*}n^{\#}{*}4$.

"What we now need is a bird such that when any number n is called to the bird, the bird will call back the number $n^{\#}$. That is, we want a bird δ such that for every number n, $\delta\bar{n} = \overline{n^{\#}}$. Can you see how to find such a bird δ?"

2 • Normalization

"For any expression X," said Griffin, "by $\ulcorner X \urcorner$ is meant the *numeral* designating the Gödel number of X. Thus $\ulcorner X \urcorner$ is $\bar{n}$, where n is the Gödel number of X. We call $\ulcorner X \urcorner$ the Gödel *numeral* of X.

"By the *norm* of X is meant the expression $X\ulcorner X \urcorner$—that is, X followed by its own Gödel numeral. If n is the Gödel number of X, then $n^{\#}$ is the Gödel number of $\ulcorner X \urcorner$ and so $n{*}n^{\#}$ is the Gödel number of $X\ulcorner X \urcorner$—the norm of X. And so if X has Gödel number n, then the norm of X has Gödel number $n{*}n^{\#}$.

"We now need a bird Δ called a *normalizer* such that for every number n, $\Delta n = \overline{n{*}n^{\#}}$. This bird is easy to find, now that we have the birds ⊛ and δ. Can you see how?"

3 • The Second Fixed Point Principle

"One can do some amazing things with the normalizer," said Griffin. "I will give you an example.

"We shall say that a term X *designates* a number n if the sentence $X = \bar{n}$ is true. Obviously, one term that designates n is the numeral $\bar{n}$, but there are infinitely many others. For example, $I\bar{n}$, $I(I\bar{n})$, $I(I(I\bar{n}))$, . . . are all terms that designate n. Also, taking 8 for n, the numeral 8 designates 8; so does the term $\oplus\bar{2}\bar{6}$; so does the term $\oplus\bar{3}\bar{5}$; so does the term $\otimes\bar{2}\bar{4}$. You get the idea!

"We call a term a *numerical* term if it designates some number n. Every numeral is a numerical term, but not every numerical term is a numeral—for example, the expression $\oplus\bar{2}\bar{6}$ is a numerical term, but it is not a numeral. For any number n, there is only one numeral designating it—the numeral $\bar{n}$—but there are infinitely many numerical terms designating it.

"Now, it is impossible that any numeral can designate its own Gödel number, because for any number n, the Gödel number of the numeral $\bar{n}$ is much larger than n. All I am saying is that for every n, $n^{\#} > n$. However, there *does* exist a *numerical term* X that designates its own Gödel number."

"That's surprising!" said Craig. "I have no idea why this should be."

"One can also construct a term that designates twice its Gödel number," said Griffin, "or one that designates three times its Gödel number, or one that designates five times its Gödel number plus seven. All these odd facts are special cases of a very important principle known as the *second fixed point principle*, which is this: For any term A, there is a term X such that the sentence $A\ulcorner X\urcorner = X$ is true. Stated otherwise, for any term A there is a term X such that X names the same bird as A followed by the Gödel numeral of X.

"Can you see how to prove this? Also, can you see how the oddities I just mentioned are special cases of the second fixed point principle?"

4 • A Gödelian Principle

"The second fixed point principle yields as a corollary an important principle attributable to Gödel, which I will tell you in a moment," said Griffin.

"For any set $\mathcal{S}$ of numbers, a sentence X is called a *Gödel sentence* for $\mathcal{S}$ if either X is true and its Gödel number is in $\mathcal{S}$, or X is false and its Gödel number is not in $\mathcal{S}$. Such a sentence can be thought of as expressing the proposition that its own Gödel number is in $\mathcal{S}$, because the sentence is true if and only if its Gödel number *is* in $\mathcal{S}$.

"Now, Gödel's principle is this: For any computable set $\mathcal{S}$, there is a Gödel sentence for $\mathcal{S}$. For example, since the set of even numbers is computable, then there must be a sentence such that either it is true and its Gödel number is even, or it is false and its Gödel number is odd. Again, since the set of odd numbers is computable, then there must be a sentence such that either it is true and its Gödel number is odd, or it is false and its Gödel number is even. The remarkable thing is that for *any* computable set, there is a Gödel sentence for that set. This follows fairly easily from the second fixed point principle. Do you see how?

"I'll give you a hint," added Griffin. "For any set $\mathcal{S}$, let $\mathcal{S}^*$ be the set of all numbers n such that n∗52 is in $\mathcal{S}$. First prove as a lemma—a preliminary fact—that if $\mathcal{S}$ is computable, so is $\mathcal{S}^*$."

"What is the significance of the number n∗52?" asked Craig.

"If n is the Gödel number of an expression X," replied Griffin, "then n∗52 is the Gödel number of the expression X = t."

How is Gödel's principle proved?

5 • The Negation Bird Pops Up

"One last detail before we answer the Grand Question," said Griffin. "For any set $\mathcal{S}$ of numbers, by $\mathcal{S}'$ we mean the set of all numbers not in $\mathcal{S}$. For example, if $\mathcal{S}$ is the set of all even numbers, $\mathcal{S}'$ is the set of all odd numbers. The set $\mathcal{S}'$ is called the *complement* of $\mathcal{S}$.

"Prove that if $\mathcal{S}$ is computable, so is $\mathcal{S}'$."

• 6 •

"Now we have all the pieces of the puzzle," said Griffin. "We are letting $\mathcal{T}$ be the set of Gödel numbers of all the true sentences. First ask yourself whether there could possibly be any Gödel sentence for the complement $\mathcal{T}'$ of $\mathcal{T}$. Then, using the last two results, show that the set $\mathcal{T}$ is *not* computable."

"This is amazing indeed!" said Craig, after he realized the solution. "It seems to shatter any hope of a purely mechanical device that can decide all mathematical questions."

"It certainly does!" said Griffin. "Any such mechanism could determine which numbers are in $\mathcal{T}$ and which ones are not, hence $\mathcal{T}$ would be a computable set, which we have just seen is not the case. Since $\mathcal{T}$ is not computable, then there is no mechanism that can compute it. In short, no mechanism can be mathematically omniscient.

"Since $\mathcal{T}$ is not computable," continued Griffin, "then no bird of this forest can compute it, and so there is no ideal bird here. Despite the cleverness of many of our birds, none of them is mathematically omniscient.

"But you know," said Griffin, with a dreamy look in his eyes, "there has been a rumor that in the days before I came here, a bird from another forest far, far away once visited this forest and astounded all the other birds by appearing to be mathematically omniscient. Of course, this is only a rumor, but who really knows? If the rumor is true, then that bird must have been most remarkable; no purely mechanistic explanation could possibly account for its behavior. Those philosophers who are mechanistically oriented and believe that birds, humans, and all other biological organisms are nothing more than elaborate mechanisms would of course deny that any such bird is possible. But I, who do not have complete confidence in the philosophy of mechanisms, reserve judgment on the matter. I'm not saying that I believe the rumor; I'm not saying that there is or was such a bird; I'm merely saying that I believe such a bird might be possible.

"I wish we had more time," concluded Griffin. "There are so many more facts about this forest that I believe would interest you."

"I have no doubt!" said Craig, rising. "I am infinitely grateful for all you have taught me, and I'm hoping that I might be able to visit this forest again one day."

"That would be wonderful!" said Griffin.

Craig left the forest the next day with a twinge of sadness. Although part of him looked forward to renewing his more normal life of crime detection, Craig realized that in his advancing years his interests were veering more and more to the purely abstract and theoretical.

"This vacation has been like an idyllic dream," thought Craig, as he reached the exit—also the entrance—gate. "I really *must* visit this forest again!"

"Only the elite are allowed to leave this forest!" said an enormous sentinel who blocked his way. "However, since you have entered this forest and only the elite are allowed to enter, then you must be one of the elite. Therefore you are free to leave, and God speed you well!"

"This is one ritual I will never understand," thought Craig, as he shook his head with an amused smile.

SOLUTIONS

1 · We first need a bird A such that for any number n, $A\bar{n} = \overline{3*s*n*4}$. We can take A to be $B(C\circledast\bar{4})(\circledast\overline{3*s})$, where B is the bluebird and C is the cardinal.

Now we want a bird δ such that for every n, if n = 0 then $\delta\bar{n} = \overline{0^{\#}}$, and if n > 0, then $\delta\bar{n} = A(P\bar{n})$. Equivalently, we want a bird δ such that for all n, $\delta\bar{n} = (Z\bar{n})\overline{0^{\#}}(A(\delta(P\bar{n})))$. Such a bird δ can be found by the fixed point principle.

2 · We take Δ to be the bird $W(DC\circledast\delta)$, where W is the warbler, D is the dove, and C is the cardinal. Then for any number n, $\Delta\bar{n} = W(DC\circledast\delta)\bar{n} = DC\circledast\delta\bar{n}\bar{n} = C\circledast(\delta\bar{n})\bar{n} = \circledast\bar{n}(\delta\bar{n}) = \circledast\bar{n}\overline{n^{\#}} = \overline{n*n^{\#}}$.

3 · Let Δ be the normalizing bird—or, more precisely, the term $W(DC\circledast\delta)$, which names the normalizing bird. Then for any expression X, the sentence $\Delta\ulcorner X\urcorner = \ulcorner X\ulcorner X\urcorner\urcorner$ is true, because X has some Gödel number n; $X\ulcorner X\urcorner$ has Gödel number $n*n^{\#}$, so the above sentence is $\Delta\bar{n} = \overline{n*n^{\#}}$.

Now take any term A. Let X be the term $BA\Delta\ulcorner BA\Delta\urcorner$, where B is a term for the bluebird. We now show that the sentence $A\ulcorner X\urcorner = X$ is true.

The sentence $BA\Delta\ulcorner BA\Delta\urcorner = A(\Delta\ulcorner BA\Delta\urcorner)$ is obviously true. Also the sentence $\Delta\ulcorner BA\Delta\urcorner = \ulcorner BA\Delta\ulcorner BA\Delta\urcorner\urcorner$ is true, hence the sentence $A(\Delta\ulcorner BA\Delta\urcorner) = A\ulcorner BA\Delta\ulcorner BA\Delta\urcorner\urcorner$ is true, and so the sentence $BA\Delta\ulcorner BA\Delta\urcorner = A\ulcorner BA\Delta\ulcorner BA\Delta\urcorner\urcorner$ is true. This is the sentence $X = A\ulcorner X\urcorner$, and so the sentence $A\ulcorner X\urcorner = X$ is true. This proves the second fixed point principle.

As an application, let us take I for A. Then there is a term X such that $I\ulcorner X\urcorner = X$ is true, hence $\ulcorner X\urcorner = X$ is true, and hence the sentence $X = \ulcorner X\urcorner$ is true. If we let n be the Gödel number of X, then the sentence $X = \bar{n}$ is true, and so X designates its own Gödel number n. By the above proof, we can take X to be the term $BI\Delta\ulcorner BI\Delta\urcorner$. However, there is a simpler term that designates its Gödel number—namely, $\Delta\ulcorner\Delta\urcorner$.

A term that designates twice its own Gödel number is $B(\otimes\bar{2})\Delta\ulcorner B(\otimes\bar{2})\Delta\urcorner$. Why?

4 • Let us first prove the lemma. For any bird A, let $A^\#$ be the bird $BA(C\circledast\overline{52})$, where B is the bluebird and C is the cardinal. For any number n, $A^\#\bar{n} = A\overline{n*52}$, because $A^\#\bar{n} = BA(C\circledast\overline{52})\bar{n} = A(C\circledast\overline{52}\bar{n}) = A(\circledast\bar{n}\overline{52}) = A\overline{n*52}$. This proves that $A^\#\bar{n} = A\overline{n*52}$.

Now suppose A computes $\mathcal{S}$. Then $A^\#$ must compute S*, because for any n in $\mathcal{S}^*$, the number $n*52$ is in $\mathcal{S}$, hence $A\overline{n*52} = t$, and so $A^\#\bar{n} = t$. Also, for any number n not in $\mathcal{S}^*$, the number $n*52$ is not in $\mathcal{S}$, hence $A\overline{n*52} = f$, and so $A^\#\bar{n} = f$. This proves that $A^\#$ computes $\mathcal{S}^*$.

Now for the proof of Gödel's principle. Suppose $\mathcal{S}$ is computable. Then $\mathcal{S}^*$ is computable, as we have just seen. Let A be a bird that computes $\mathcal{S}^*$. By the second fixed point principle there is a term X such that the sentence $A\ulcorner X\urcorner = X$ is true. Let Y be the sentence $X = t$. We will show that Y is a Gödel sentence for the set $\mathcal{S}$.

Let n be the Gödel number of X. Then Y, being the sentence $X = t$, has Gödel number $n*52$.

a. Suppose that Y is true. Then the sentence $X = t$ is true and since the sentence $A\ulcorner X\urcorner = X$ is also true, then the sentence $A\ulcorner X\urcorner = t$ is true, and so the sentence $A\bar{n} = t$ is true (because $\ulcorner X\urcorner$ is the numeral $\bar{n}$). Therefore n belongs to the set $\mathcal{S}^*$ (because A computes S*, hence if n didn't belong to S*, then the sentence $A\bar{n} = f$ would be true, which is impossible, since $A\bar{n} = t$ is true). Since n belongs to $\mathcal{S}^*$, then n*52 belongs to $\mathcal{S}$, but n*52 is the Gödel number of the sentence Y! This proves that if Y is true, then its Gödel number n*52 belongs to $\mathcal{S}$.

b. Conversely, suppose n*52 belongs to $\mathcal{S}$. Then n belongs to $\mathcal{S}^*$, hence $A\bar{n} = t$ is true, which means that Y is true. And so if the Gödel number of Y is in $\mathcal{S}$, then Y is true, or what is the same thing, if Y is false, then its Gödel number does not belong to $\mathcal{S}$.

According to argument a and argument b, we see that if Y is true, then its Gödel number is in $\mathcal{S}$ and if Y is false, then its Gödel number is not in $\mathcal{S}$. And so Y is a Gödel sentence for $\mathcal{S}$.

5 · Let A compute $\mathcal{S}$. Then BNA computes $\mathcal{S}'$, where B is the bluebird and N is the negation bird. *Reason:* For any number n, $BNA\bar{n} = N(A\bar{n})$. If n belongs to $\mathcal{S}'$ then n doesn't belong to $\mathcal{S}$, hence $A\bar{n} = f$, hence $N(A\bar{n}) = t$, so $BNA\bar{n} = t$. If n doesn't belong to $\mathcal{S}'$, then n belongs to $\mathcal{S}$, hence $A\bar{n} = t$, hence $N(A\bar{n}) = f$, and so $BNA\bar{n} = f$. Therefore BNA computes $\mathcal{S}'$.

6 · There certainly cannot be any Gödel sentence Y for the set $\mathcal{T}'$, because if Y is true, then its Gödel number is in $\mathcal{T}$, not in $\mathcal{T}'$, and if Y is false, its Gödel number is in $\mathcal{T}'$, not in $\mathcal{T}$. Therefore there is no Gödel sentence for $\mathcal{T}'$.

Now, if $\mathcal{T}$ were computable, then $\mathcal{T}'$ would be computable by Problem 5, hence by Problem 4 there would be a Gödel sentence for $\mathcal{T}'$. Since there is no Gödel sentence for $\mathcal{T}'$, then the set $\mathcal{T}$ is not computable.

Epilogue

Inspector Craig arrived home not long afterward, and the first thing he did (after solving the case of the bat and the Norwegian maid) was to spend a long holiday weekend with his old friends McCulloch and the logician Fergusson.* He told them the entire story of his summer adventures.

"I have known nothing about combinatory logic till now," said McCulloch, "and I must say that the subject intrigues me enormously. But I would like to know how, when, and why the field ever got started. What was the motivation, and are there any practical applications?"

"Many," replied Fergusson (who was quite knowledgeable about all this). "Why, these days combinatory logic is one of the big things in computer science and artificial intelligence. The study of combinators started early in the twenties, pioneered by Shönfinkel. It is curious that *schön* in German means "beautiful," and *finkel* means "bird," hence *Shönfinkel* means "beautiful bird." So perhaps there's been a connection between birds and combinators all along! At any rate, the subject was further developed by Curry, Fitch, Church, Kleene, Rosser, and Turing, and in later years by Scott, Seldin, Hindley, Barandregt, and others. Their interests were purely theoretical; they were exploring the innermost depths of logic and mathematics. No one then could have dreamed of the impact the subject would one day have on computer science. In recent times, the subject has been put on a more solid foundation—largely through the efforts of the logician Dana Scott, who provided interesting models for the theory."

* A complete account of McCulloch's remarkable number machines and Fergusson's logic machines can be found in *The Lady or the Tiger?* (Alfred A. Knopf, 1982).

"How is combinatory logic related to computer science?" asked Craig. "Professor Griffin didn't say too much about that."

"Why, in the construction of *programs*," replied Fergusson. "Computers run on programs, you know, and these days all computer programs can be written in terms of combinators. The essential idea is that, given any programs X and Y, we can obtain a new program by feeding Y as input to the computer whose program is X; the resulting output is the program XY. The situation is analogous to calling out the name of one of Griffin's birds y to a bird x and getting the name of the bird xy as a response. The analogy is quite exact: Just as all combinatorial birds are derivable from the two birds S and K, so are *all* computer programs expressible in terms of the basic combinators S and K. We have here a case of what mathematicians call *isomorphism*, which in this instance means that the birds of Griffin's forest can be put into a one-to-one correspondence with all computer programs in such a manner that if a bird x corresponds to a program X and a bird y corresponds to a program Y, then the bird xy will correspond to the program XY. This is what Griffin must have meant when he said that, given any computer, there is a bird in his forest which can match it.

"I can certainly see," concluded Fergusson, "why Griffin has no need of computers: Because of the isomorphism of his forest of birds to the class of computer programs, it follows that any information a computer scientist can obtain from running his programs, Griffin can get just as surely by interrogating his birds. And yet Griffin's ideals seem to contrast strangely with those of people working in artificial intelligence. The latter are trying to simulate the thinking of biological organisms. Griffin is now turning the tables by using biological organisms—birds, in this case—to do the work of clever mechanisms. I believe the two approaches cannot but supplement each other, and it should be extremely interesting to see the outcome of all this!"

Many years later, Craig did indeed return to ~~the~~ Master Forest. But that is another story.