The Case of the Vanishing Tuples

ANNA SFARD

The story which I will tell in a minute is a small case study in mathematical discovery. Its hero — the vanishing tuple — may be quite insignificant as a mathematical concept, but the investigation which disclosed its real nature was a real drama. What started with an innocent-looking problem ended only after hours of intensive work. And there was one other thing that made this story special: it was a true mathematical exploration, but it was not a one-actor play. Here, there were two dramatis personae: *my PC and I.*

Even so, the reader may wonder why I find this personal experience worth telling. My first answer would be that what happened to me may happen to anybody. It is true that in my "experiment" I had no other guinea-pig except myself, but I believe in the illuminating power of introspection. I am confident that many of the phenomena brought to light by analyzing a person's own experience often turn out to be quite universal. Also, this case study has some important implications regarding the role the computer may play in stimulating a student's mathematical creativity. In the sequel, I will devote several paragraphs to this important topic. But let me tell my story first.

Prologue: seeing the problem

> I have had my results for a long time; but I do not know yet how I am to arrive at them. [Gauss, quoted by Lakatos, 1976, p. 9]

It was toward the end of my visit to the Weizmann Institute that somebody opened the door and said:

> Take a sequence of four integers, for instance (12, 7, 3, 79). Build a new sequence composed of the absolute values of the differences between two adjacent elements of the first sequence (the last element in the sequence is regarded as "adjacent" to the first): (5, 4, 76, 67).
>
> Repeat this operation: (1, 72, 9, 62)
>
> Perform the subtractions again and again. Where would these repetitions lead you?

This problem, the person explained, popped out of an instructional activity for elementary school children learning subtraction (see "Dify Game" in [2]). "It seems that you must *always* get a row of zeros in the end, but the annoying thing is I cannot see why," he complained.

On my way back to Jerusalem that day I could not shift this problem from my mind. It was not because of its importance (I could not see that it had any), and certainly not because I was a "professional" problem-solver (I am not), but because it was upsetting that the puzzle would not yield to the usual strategies in spite of its apparent simplicity. Also, at this time I was learning a new computer language called ISETL,* and it seemed that it would be fun to practice the programming while tackling a mathematical brain-teaser. The "vanishing tuples problem" (which is what I christened it right away) seemed just perfect for such a computerized inquiry and ISETL appeared to be an ideal instrument for the purpose.

Act one: Conjecturing

> The real aim of a "problem to prove" should be to improve ... the original "naive" conjecture into a genuine "theorem." [Lakatos, 1976, p. 41]

I embarked on a systematic investigation that very evening. My PC was there, beckoning. First of all, to make things clear, I explicitly stated the conjecture I was going to investigate:

> C: All 4-tuples composed of natural numbers must vanish.

By *vanishing tuple* I meant a tuple which, after a finite number of consecutive dify-operations (as I chose to name our special transformations), would turn into a string of zeros — into a "null-tuple."

The almost inevitable result of my decision to use the computer for checking this conjecture was its instant generalization: instead of dealing with strings of four numbers, I would try to check the proposition for strings of *any* length. Such generalization is in the very bones of computer programming — here it is only natural to use variables instead of constants. (By comparison, think about the cost of making such a generalization when it must be followed up *without* a computer!) The availability of the computer, therefore, was the first reason behind the following generalization of C:

> C_1: Every n-tuple, $n \in \mathbf{N}$, must vanish.

The new supposition appeared thoroughly plausible. Indeed, I could not see what was so special about the number 4. This number had been picked by chance, I thought, or just because it was big enough to be non-trivial (thus "representative"), but at the same time small enough not to embroil the solver in complex computations.

I quickly wrote a simple ISETL program (Figure 1) which would help me check the vanishing of the n-tuples

*ISETL, the Interactive Set Language, was designed for teaching basic mathematical concepts, such as set, function and logical relation. It is a very high level language whose similarity to the symbolic system of mathematics makes it particularly suitable for mathematical investigations. For further details see [3]

```
DIFY: = func(I0;
s:=T;
for i in {1 . . #I} do
       if i < #I then
              s(i):=abs(I(i+1)-I(i));
       else
              s(i):=abs(I(1)-I(i);
       end;
end;
return s;
end;
while (exists × in I(×>0) do
       T:=DIFY(I); print T; end;
```

Figure 1

```
I:= [13, 45, 0, 2];

    [32, 45, 2, 11];
    [13, 43, 9, 21];
    [30, 34, 12, 8];
    [4, 22, 4, 22];
    [18, 18, 18, 18];
    [0, 0, 0, 0];
```

Figure 2

Obviously I applied it first of all to a 4-tuple (Figure 2), Yes, it worked. So now it was only natural to check some other tuple-length. I decided to use a 5-tuple as an input. And here came the surprise: the program was supposed to stop the moment the null-tuple was obtained — but it continued to work until artificially interrupted (Figure 3) Instead of vanishing, a non-zero output became recurrent

```
[6, 66, 48, 30, 54];
[60, 18, 18, 24, 48];
[42, 0, 6, 24, 12];
[42, 6, 18, 12, 30];
[36, 12, 6, 18, 12];
[24, 6, 12, 6, 24];
[18, 6, 6, 18, 0];
[12, 0, 12, 18, 18];
[12, 12, 6, 0, 6];
[0, 6, 6, 6, 6];
[6, 0, 0, 0, 6];
[6, 0, 0, 6, 0];
[6, 0, 6, 6, 6];
[6, 6, 0, 0, 0];
[0, 6, 0, 0, 6];
[6, 6, 0, 6, 6];
[0, 6, 6, 0, 0];
[6, 0, 6, 0, 0];
[6, 6, 6, 0, 6];
[0, 0, 6, 6, 0];
[0, 6, 0, 6, 0];
[6, 6, 6, 6, 0];
[0, 0, 0, 6, 6];
[0, 0, 6, 0, 6];
[0, 6, 6, 6, 6];
```

Figure 3

```
[8, 3, 2, 9, 9, 13];
[5, 1, 7, 0, 4, 5];
[4, 6, 7, 4, 1, 0];
[2, 1, 3, 3, 1, 4];
[1, 2, 0, 2, 3, 2];
[1, 2, 2, 1, 1, 1];
[1, 0, 1, 0, 0, 0];
[1, 1, 1, 0, 0, 1];
[0, 0, 1, 0, 1, 0];
[0, 1, 1, 1, 1, 0];
[1, 0, 0, 0, 1, 0];
[1, 0, 0, 1, 1, 1];
[1, 0, 1, 0, 0, 0];
[1, 1, 1, 0, 0, 1];
[0, 0, 1, 0, 1, 0];
[0, 1, 1, 1, 1, 0];
[1, 0, 0, 0, 1, 0];
```

Figure 4

```
[385, 459, 74];
[74, 385, 311];
[311, 74, 237];
[237, 163, 74];
[74, 89, 163];
[15, 74, 89];
[59, 15, 74];
[44, 59, 15];
[15, 44, 29];
[29, 15, 14];
[14, 1, 15];
[13, 14, 1];
[1, 13, 12];
[12, 1, 11];
[11, 10, 1];
[1, 9, 10];
[8, 1, 9];
[7, 8, 1];
[1, 7, 6];
[6, 1, 5];
[5, 4, 1];
[1, 3, 4];
[2, 1, 3];
[1, 2, 1];
[1, 1, 0];
[0, 1, 1];
[1, 0, 1];
[1, 1, 0];
```

Figure 5

All right, I thought, so the conjecture is not true for *every* tuple. It holds, however, not only for n = 4, but also for n = 2 — this I knew even without checking. So it is probably true at least for all *even* tuples, I concluded. By an *even tuple* I meant a tuple with an even number of elements. C_1 underwent a modification:

C_2: Every *even* tuple must vanish.

I decided to check the new supposition for a 6-tuple — only to find out I was wrong again (Figure 4)!

I still believed, however, that the property of vanishing (or any other, for that matter) could not hold *only* for four. I began to look for another generalization, this time in a rather arbitrary way. I just took a 3-tuple and 7-tuple — and the new disappointments appeared just before my eyes (Figures 5 and 6).

I could no longer believe the conjecture true for *any* number, not even for 4. I was convinced now that I had just fallen victim to an unfortunate choice of initial examples. My new conjecture was clearly that of a quite desperate person:

C_3: The claim about vanishing *is not generally true* for any number

So I wrote a new program, which would check for me all the 4-tuples composed of the numbers (3,4,5) (Figure 7). Why did I decide to choose such unimaginative numbers? Well, I had already noticed that in all the cases I tackled, only one non-zero number remained eventually in the tuple, so I supposed the initial numbers were not very important.

```
[18, 8, 0, 12, 52, 22, 16];
[10, 8, 12, 40, 30, 6, 2];
[2, 4, 28, 10, 24, 4, 8];
[2, 24, 18, 14, 20, 4, 6];
[22, 6, 4, 6, 16, 2, 4];
[16, 2, 2, 10, 4, 2, 18];
[14, 0, 8, 4, 12, 16, 2];
[14, 8, 4, 8, 4, 14, 12];
[6, 4, 4, 4, 10, 2, 2];
[2, 0, 0, 6, 8, 0, 4];
[2, 0, 6, 2, 8, 4, 2];
[2, 6, 4, 6, 4, 2, 0];
[4, 2, 2, 2, 2, 2, 2];
[2, 0, 0, 0, 0, 0, 2];
[2, 0, 0, 0, 0, 2, 0];
[2, 0, 0, 0, 2, 2, 2];
[2, 0, 0, 2, 0, 0, 0];
[2, 0, 2, 2, 0, 0, 2];
[2, 2, 0, 2, 0, 2, 0];
[0, 2, 2, 2, 2, 2, 2];
[2, 0, 0, 0, 0, 0, 2];
```

Figure 6

```
HOPEFULLY:=func(str);
s:=[str];
while (exists x in s(#s) x > 0) do
        s:=s+[DIFY(s(#s))];
end;
for s in ([a,b,c,d] a,b,c,d in {3..5}) do
        print HOPEFULLY(s);
end;
```

Figure 7

```
[17, 12, 5, 12, 17, 2, 1, 16];
[5, 7, 7, 5, 15, 1, 15, 1];
[2, 0, 2, 10, 14, 14, 14, 14];
[2, 2, 8, 4, 0, 0, 10, 2];
[0, 6, 4, 4, 0, 10, 8, 0];
[6, 2, 0, 4, 10, 2, 8, 0];
[4, 2, 4, 6, 8, 6, 8, 6];
[2, 2, 2, 2, 2, 2, 2, 2];
[0, 0, 0, 0, 0, 0, 0, 0];
```

Figure 8

The program ran, and ran, and ran — but it did stop, eventually. All the $3^4 = 81$ different 4-tuples did vanish. It was not very encouraging to find that I was probably wrong again.

Whether I liked it or not, four seemed to be a very special number after all. But in spite of the ample evidence I would not surrender. I continued to brood on something that could well be regarded as a lost cause when the sudden enlightenment came: among all the lengths I had checked: 2, 3, 4, 5, 6, and 7, the only distinctive feature of the numbers 2 and 4 I could think of was that they are powers of 2. Could it be that the conjecture was true for all such numbers? I ventured the new supposition:

C_4: Every 2^k-tuple, $k \in N$, must vanish.

With new hope in my heart I ran the first program for 8 and for 16 (Figures 8 and 9). Yes, it did work. Now I had a very interesting conjecture before me. The plausibility of this new supposition was enhanced by the fact that the special numbers, 2^k, seemed to hint at a reason for its validity: the 2^{k+1}-tuple can probably be presented as a combination of two 2^k-tuples, so the conjecture would be true on the strength of an inductive argument.

```
[1, 8, 4, 2, 3, 0, 1, 7, 11, 14, 14, 22, 11, 2, 11, 3];
[7, 4, 2, 1, 3, 1, 6, 4, 3, 0, 8, 11, 9, 9, 8, 2];
[3, 2, 1, 2, 2, 5, 2, 1, 3, 8, 3, 2, 0, 1, 6, 5];
[1, 1, 1, 0, 3, 3, 1, 2, 5, 5, 1, 2, 1, 5, 1, 2];
[0, 0, 1, 3, 0, 2, 1, 3, 0, 4, 1, 1, 4, 4, 1, 1];
[0, 1, 2, 3, 2, 1, 2, 3, 4, 3, 0, 3, 0, 3, 0, 1];
[1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 3, 3, 1, 1];
[0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 2, 0, 0];
[0, 0, 0, 0, 0, 0, 2, 2, 0, 0, 0, 2, 2, 0, 0];
[0, 0, 0, 0, 0, 2, 0, 2, 0, 0, 2, 0, 2, 0, 0];
[0, 0, 0, 0, 2, 2, 2, 2, 0, 2, 2, 2, 2, 0, 0];
[0, 0, 0, 2, 0, 0, 0, 2, 2, 0, 0, 0, 2, 0, 0];
[0, 0, 2, 2, 0, 0, 2, 0, 2, 0, 0, 2, 2, 0, 0];
[0, 2, 0, 2, 0, 2, 2, 2, 2, 0, 2, 0, 2, 0, 0];
[0, 2, 2, 2, 2, 2, 0, 0, 0, 2, 2, 2, 2, 2, 0, 0];
[2, 0, 0, 0, 0, 2, 0, 0, 2, 0, 0, 0, 0, 2, 0, 0];
[2, 0, 0, 0, 2, 2, 0, 2, 2, 0, 0, 0, 2, 2, 0, 2];
[2, 0, 0, 2, 0, 2, 2, 0, 2, 0, 0, 2, 0, 2, 2, 0];
[2, 0, 2, 2, 2, 0, 2, 2, 2, 0, 2, 2, 2, 0, 2, 2];
[2, 2, 0, 0, 2, 2, 0, 0, 2, 2, 0, 0, 2, 2, 0, 0];
[0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2];
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2];
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0];
```

Figure 9

Act II: Proving

Scene 1: still induction

> I propose to retain the time-honoured technical term "proof" for . a decomposition of the original conjecture into subconjectures or lemmas [Lakatos, 1976, p 9]

It is time now to look for a proof, I thought Did it mean that the inductive exploration was over and from now on I was going to restrict myself to the good old deduction? Not necessarily. As before, I turned for inspiration to the computer output. Maybe the most striking feature of all the sequences I checked was that, from a certain point, each one of them contained tuples composed of two numbers only, one of them zero. These special tuples seemed to be central enough to the problem to deserve a special name I decided to call them *binary* It seemed now that C_4 may be proved by the help of the following two lemmas:

L_1: Every tuple must become binary after a certain number of transformations.

L_2: The same as C_4, but for binary tuples only.

The emerging system of claims could be refined even more The vanishing binary tuples had a distinctive feature which could not go unnoticed: they vanished pretty fast (see Figures 2, 8, and 9). In any case it seemed that there was a clear limit to the number of steps necessary for this to happen Nothing was more natural than to count the steps needed for a binary tuple to vanish. The reader is invited to do the same I immediately found that the number never exceeded the length of the tuple. My feeling was that this fact would be the most natural outcome of the inductive process (I mean the *mathematical* induction, this time) which will be used to prove the vanishing of the binary 2^k-tuples.

All this lengthy and most enjoyable investigation ended with a neat system of claims, presented in Figure 10

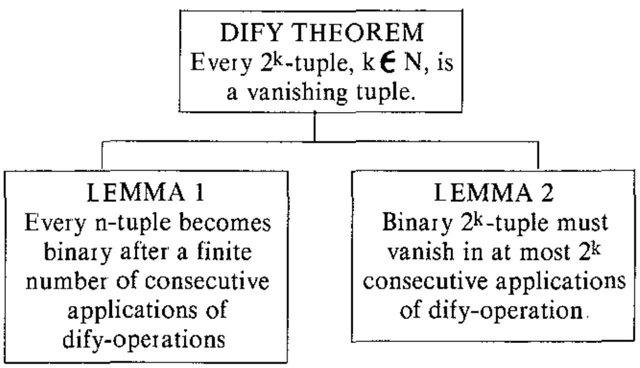

Figure 10

Scene 2 mathematical induction

> Induction is the process of discovering general laws by the observation and combination of particular examples. It is used in all sciences, even in mathematics. Mathematical induction is used in mathematics alone to prove theorems of a certain kind. It is rather unfortunate that the names are connected because there is very little logical connection between the two processes. There is, however, some practical connection: we often use both methods together [Polya, 1957, p. 114]

Yes, mathematical induction is often the most natural way by which mathematicians can give a finishing touch to the results of their "empirical" investigations (it is only in school that "proof by induction" is almost never preceded by inductive inquiry!) But it is also a little more than that Usually, an inductive argument can disclose the mechanism which makes a certain property into a "hereditary trait" of natural numbers. Thus I turned to mathematical induction not only to know *that* my conjecture was true, but also to find out *why*

It seemed obvious that mathematical induction was the right tool for both L_1 and L_2. Yet, the task was not as straightforward as it appeared. It took me two days and one sleepless night to arrive at ideas that worked It was L_1 which turned out the more problematic, but in the end I overcame all the difficulties. For L_1 I have found a proof which, I am sorry to say, this article is too narrow in its scope to contain (let those to whom this statement sounds like Fermat's famous remark about his last theorem be assured that I *really* have a solution, and I would be pleased to make it available upon request). Anyway, it was the proof of L_2 with which I was particularly pleased as I felt there is some real beauty in it. Let me tell a few words about it

As I said before, showing that a 2-tuple must vanish was not a problem. As usual the difficulty began when the inductive transformation from 2^k to 2^{k+1} had to be made It was quite clear that to prove the vanishing of a binary 2^{k+1}-tuple one had to find a way of splitting such a tuple into two 2^k-tuples I could see many ways of doing it, but it was most natural to present an 8-tuple like T = [0, 1, 1, 0, 1, 1, 1, 1] either as the *concatenation* of its two halves, T_1 = [0, 1, 1, 0] and T_2' = [1, 0, 1, 1] (T_1' is composed of odd-numbered elements of T and T_2' of the others) But there was a problem: the dify-operation did not preserve either of these combinations When the dify-operation was applied to "the halves" separately the result was not the same as when it was applied to the original tuple For instance, in the case of T_1, T_2, and T, the resulting tuples would be [1, 0, 1, 0], [0, 0, 0, 0], and [1, 0, 1, 1, 0, 0, 0, 1] respectively, and it is clear that the last tuple is not the concatenation of the first two. Also this last tuple is not an alternation of [1, 0, 0, 1] and [1, 1, 0, 0] which result from T_1' and T_2' Thus I could not finish the job just by saying that the 2^{k+1}-tuple must vanish because its halves have to vanish.

At this point, a conjecture about the maximal number of dify-operations necessary to "annihilate" a binary tuple proved helpful It said that a 2^k-tuple would vanish in at most 2^k steps It seemed reasonable, therefore, that a series of 2^{k+1} operations on a 2^{k+1}-tuple could somehow be translated into a series of only 2^k operations on each of its "halves." With this idea in mind, I needed just one look (well, maybe two or three) at a simple example to realize that two applications of dify to T are equivalent to one application to both T_1' and T_2' (see Figure 11; with an asterisk I denote the operation of alternation)

T		T_1'		T_2'
[0, 1, 1, 0, 1, 1, 1, 1]	=	[0, 1, 1, 1]	*	[1, 0, 1, 1]
[1, 0, 1, 1, 0, 0, 0, 1]				
[1, 1, 0, 1, 0, 0, 1, 0]	=	[1, 0, 0, 1]	*	[1, 1, 0, 0]
[0, 1, 1, 1, 0, 1, 1, 1]				
[1, 0, 0, 1, 1, 0, 0, 1]	=	[1, 0, 1, 0]	*	[0, 1, 0, 1]
[1, 0, 1, 0, 1, 0, 1, 0]				
[1, 1, 1, 1, 1, 1, 1, 1]	=	[1, 1, 1, 1]	*	[1, 1, 1, 1]
[0, 0, 0, 0, 0, 0, 0, 0]				
[0, 0, 0, 0, 0, 0, 0, 0]	=	[0, 0, 0, 0]	*	[0, 0, 0, 0]

Figure 11

Indeed, as can easily be seen in Figure 11, each of the subsequent 4-tuples on the right (the "halves" of the 8-tuple) is just the result of the dify-operation on the 4-tuple written right above it. But if one looks at the corresponding sequence of 8-tuples on the left, "just above" means two rows above!

Now it was quite obvious how the truth of the claim for 2^{k+1} followed from its truth for 2^k: if each of the halves of T must vanish in at most 2^k steps, then T itself must vanish in not more than 2^{k+1} stemps. To complete the proof one thing had yet to be done: I had to show that the relation discovered in the example was *always* true. To prove it, I had to demonstrate that if S = [. . , a, b, c, .] was a binary tuple and S″ = [. , a_2, b_2, c_2, . .] was the result of two successive applications of the dify-operation to S, then a_2 could be presented not only as $||a-b|-|b-c||$ but also as a simple difference between a and c. In other words, I had to show that the equation

$$||a-b|-|b-c|| = |a-c|$$

holds for any a, b, c, $\in$ (0, 1). Since there are only eight triples of zeros and ones, I did it with the computer (Figure 12). "Real" mathematicians would probably shrug at such behavior but, once again, communicating with the computer seemed more appealing an option than just talking to myself.

```
[[0, 1, 1], ||a-b|-|b-c|| = 1, |a-c| = 1];
[[0, 1, 0], ||a-b|-|b-c|| = 0, |a-c| = 0];
[[0, 0, 1], ||a-b|-|b-c|| = 1, |a-c| = 1];
[[0, 0, 0], ||a-b|-|b-c|| = 0, |a-c| = 0];
[[1, 1, 1], ||a-b|-|b-c|| = 0, |a-c| = 0];
[[1, 1, 0], ||a-b|-|b-c|| = 1, |a-c| = 1];
[[1, 0, 0], ||a-b|-|b-c|| = 1, |a-c| = 1];
[[1, 0, 1], ||a-b|-|b-c|| = 0, |a-c| = 0];
```

(prepared by the help of the following ISETL program:

```
for T in ([a, b, c]: a, b, c, in (0 . . 1)) do
    x:=abs(abs(T(1)-T(2))-abs(T(2)-T(3)));
    y:=abs(T(1)-T(3));
    z:=[T, "||a-b|-|b-c||=",x,"|a-c|=",y];
print z;
end;)
```

Figure 12

Let T be a tuple of any length, then

T(i) = the i-th element of T; *l(t)* = the length of T;

dify(T) = [|T(2)–T(1)|, |T(3)–T(2)|, . . , |T(1)–T(l(T))|];

T^k: T^0 = T; T^{k+1} = dify (T^k);

if l(t) = 2k then

o(T) = [T(1), T(3), . , T(2k–1)], *e(T)* = [T(2), T(4), . , T(2k)]

L_2. If T is binary and $l(T) = 2^k$ for a certain $k \in N$, then T vanishes in at most 2^k steps

Proof: by induction on k

The claim is obviously true for k = 1.

Suppose it is true for k and let us prove it for k + 1

Let T be a binary 2^{k+1}-tuple.

Let us show that

(1) $o(T^2) = [o(T0]^1$, $e(T^2) = [e(T)]^1$

This will complete the proof, because the inductive assumption combined with generalization of (1) implies that

$o(T^{2k+1} = [o(T)]^{2^k} = [0, \ldots 0]$ and

$e(T^{2k+1}) = [e(t)]^{2^k} = [0, \ldots 0]$

To prove (1) it is enough to show that for every i

$||T(i+1)-T(i)|-|T(i)-T(i-1)|| = |T(i+1)-T(i-1)|$

This is equivalent to the statement: if a, b, c, $\in$ (0, 1) then $||a-b|-|b-c|| = |a-c|$

If a = c then $|a-c| = 0$ and $|a-b| = |b-c|$, so the equation holds. Otherwise, $|a-c| = 1$. Also, either b=a or b=c, and in any case, $||a-b|-|b-c|| = 1$.

Scene 3: Polishing and formalizing

> When one has constructed a fine building, the scaffolding should no longer be visible. [Gauss, quoted by Stewart, 1987, p. 142]

What has been told here as a chronicle of events had now to be cast into the concise language of formal mathematics. I indulged in inventing symbols and took pleasure in making the long story short. Yes, I really enjoyed myself while advancing toward the compact proof presented above. Naturally, deciphering it may be quite a different story!

Epilogue: new problems

> A scientific inquiry "begins and ends with problems" [Lakatos, 1976, p. 105, after Popper]

Even now, after my conjectures were proved, I could not feel that the case of the vanishing tuples was closed. Too many questions still popped up from the data provided with such generosity by my machine. First of all there were periodic tuples, which also deserved some attention. By periodic tuple I mean a tuple which instead of vanishing leads to a recurrent sequence like those in Figures 3, 4, 5, and 6. That "periodic" is the only alternative to "vanishing" just cries out from the computer output, but it can also be easily reinforced with deductive argument (the dify-operation cannot increase a tuple's maximum so the "pigeonhole principle" completes the job). Now I wondered whether the set I had determined in my theorem, together with all the constant tuples (tuples with only one repeated number), exhausted all possible vanishing tuples In other words, must all the other non-constant tuples — those whose lengths are not a power of two — be periodic?

My first supposition was that this was the case And I believed it until the person who had initiated the investigation came up with the surprising example: (3, 4, 5, 6, 5, 4) It was a non-constant 6-tuple that vanished! So 2^k-tuples were not the only ones, after all. Although in my story my PC was promised the role of good guy, the truth must be clearly spelled out: he, the machine, led me astray by never coming across an appropriate example I trusted him too much for what I thought was good reason: I just knew that ISETL uses a random function when choosing its input, and I went through lots and lots of examples picked out by the machine The vanishing 6-tuple put things into the right proportions.

In the end, after many more encounters with the PC and its products, I was able to compile the following list of new problems:

P_1: Is it true that every non-constant tuple whose length is an odd number must be periodic? (The answer is yes — I have proved it; now I challenge the reader to do the same)

P_2: If n is odd, it seems that the length of the period must be a multiple of n. Is this true?

P_3: Is there any limit to the number of steps needed for a tuple to become binary? In my investigation I have proved that to turn a tuple whose maximal element is M into binary, $M(M+1)/2$ steps always suffice Now I wonder whether this result may be improved

The reader who finds these problems interesting is urged to investigate them in the manner presented in this paper: by first looking at appropriate examples (produced by the computer!) and then by finding a suitable general argument

Afterword

> Mathematics presented with rigor is a systematic deductive science, but mathematics in the making is an experimental science. [Polya, 1954, p 117]

There is a moral to this story.

Of the two kinds of mathematics mentioned by Polya it is the "mathematics presented with rigor" which is usually taught in schools and in colleges Skemp laments [1971, p. 13]: "This approach . gives only the end-product of mathematical discovery . and fails to bring about in the learner those processes by which mathematical discoveries are made It teaches mathematical thought, not mathematical thinking."

In fact it is hardly surprising that the inductive components of "mathematics in the making" have been banned from the classroom Whatever subjects are covered in a secondary or undergraduate curriculum, it has always been rather difficult to find material which can serve the student as a training ground for conjecturing and proving. Indeed, most of the inductive problems which seem elementary enough to be used in school are either trivial or too time-consuming to be workable.

The point I have tried to make throughout this paper is that with the help of the computer this last difficulty can be overcome. This idea is not entirely new: important steps toward using technology to create an environment for conjecturing and generalizing have already been made, mainly in the domain of geometry [see e.g Papert, 1980; Schwartz and Yerushalmy, 1985] In this paper another mode of computer-aided mathematical investigation has been suggested.

Some readers may object to my claim that without a computer the problem I chose would not be suitable for classroom investigation "Whatever has been done with the machine would also be possible with pen and paper," somebody might say. This person would certainly be right, at least theoretically. The real question, however, is not whether such exploration would be *possible* without the computer, but whether it would be *workable*. The history of education is full of beautiful ideas which have crashed on the invisible borderline between the possible and the feasible Without a computer explorations like those presented above would be possible in theory but not in practice And feasibility is what is needed in the classroom to make things work

The computerized investigation just presented was never actually carried out in school, but in the light of my personal experience it seems probable it has considerable educational potential. True, such an exercise in mathematical investigation may not be suitable for everybody. The suggested activity, however, may be carried out in several different versions, and the degree of difficulty modulated according to the capabilities of the intended audience

Let me analyze in more depth the possible educational benefits of such instructional undertaking. While telling my story I tried to bring to light the facilitating effect the computer can have on the explorer's reasoning: thanks to the machine, with its readiness to provide the user with tens and hundreds of examples, the problem-solver finds his or her meandering path from induction to deduction in the most natural way, almost without effort. But it seems to me that the most important moral from my tale concerns the impact the computer can have on the problem-solvers' attitude — on their motivation and on their understanding of the necessity and the nature of mathematical proof

The affective gains from using technology can hardly be overestimated. My own experience clearly shows the computer's potential for stimulating *curiosity* and for sustaining *willingness* to work on a problem With the machine, with its inherent readiness for generalization and its capability of displaying otherwise inaccessible sides of mathematical structures, the most interesting questions jump straight out at the problem-solver's eyes The entire task of conjecturing and proving becomes much more *interesting* as rote calculations are traded for the much more exciting activity of programming.

Paradoxically the "dumb machine," this alleged foe of pure mathematical thought, may have a considerable effect on students' understanding of the concept of deductive proof.

It is only natural that proving, maybe the most advanced kind of mathematical activity, is also one of the most problematic themes in the secondary and tertiary mathematics [for a concise survey of the relevant research see Dreyfus, 1990]

Since proof is conceived by students mainly as a means of *convincing* someone about facts, the majority of pupils cannot see the difference between deductive argumentation and a persuasive demonstration. Furthermore, as the conjectures to be proved are usually provided by the teacher, their validity is taken for granted, so the student has neither the sense of necessity nor the curiosity to engage in a formal deductive proof. Apart from this, the majority of elementary geometrical theorems — by far the most common type of problem to be dealt with in the classroom — are either self-evident or their proofs are based on *deus ex machina* auxiliary constructions unlikely to be reinvented by the student So the task of proving becomes a mere classroom game, played according to certain arbitrary rules, and with the sole objective of gaining the approval of the only and ultimate judge — the teacher To all this, a "computerized" inquiry may be the best answer. Since the task of posing problems and of conjecturing can now be tackled in the classroom with a reasonable chance of success, teachers may gradually realize that there is no reason any more to confine student's activity to proving ready-made answers to never-asked questions.

The computer as a tool for mathematical inquiry also has its drawbacks As in my case (of my first conjecture about periodic tuples), the machine may breed much too much faith in false guesses. With its practically unbounded capability of providing "empirical" evidence, it may also reinforce student's belief in the power of concrete instances. The weapon, however, with which such convictions are raised and protected can be easily turned against them The students may be asked, for instance, to explore the conditions for the primeness of $n^2 - n + 41$ (n integer), or to investigate the properties of the sequence $a_n = (2^n - 2)/n$. In both cases, the first examples will probably lead the solver to a false conjecture ("the number is prime for any n" in the first case, and "a_n is an integer if n is prime" in the second) If the computer programs for finding and examining the numbers in question are available, the subsequent checks will only increase the student's confidence in the validity of both suppositions. Counterexamples will not be encountered until sufficiently large numbers are considered ($n = 41$ in the first case and $n = 341$ in the second). At this stage the surprise should be big enough to shake the student's confidence in "proving by examples" once and for all. Incidentally, both problems can hardly be tackled in the proposed way if the necessary computations have to be performed with pen and paper only

When the computer was first considered as a prospective teaching aid, some educators expressed fears about the harm this "intellectual crutch" would inflict upon the student and his mathematical skills. In this paper I have tried to show that with a little openness and good will, the number-cruncher may be turned into a powerful intellectual tool In the classroom it will certainly help mathematics revive its colors and regain its allure.

Postscript 1: Although I had made some early inquiries, it was only after I wrote this paper that I learned about previous advances in the case of the vanishing tuples. It turned out that the problem drew the attention of mathematicians some time ago, and since then nearly twenty articles concerned with its different aspects have been published in various journals (notably, *The Fibonacci Quarterly* and the *American Mathematical Monthly;* for references see Erlich, [1989]) I have also found out that the official name for our little puzzle is "the problem of Ducci sequences" or "the n-number game." Needless to say, "professional" treatments turned out to be less elementary than the one presented in this paper

Postscript 2: It was Professor Maxim Bruckheimer of the Weizmann Institute who threw me the glove and then sustained my interest by being prepared to listen and by challenging me with unexpected examples It was Jeanne Albert of the Weizmann Institute who can across the problem in materials she used for teacher training I am glad to have been at the Weizmann Institute when it happened

References

Baxter, N. Dubinsky E. & Lewin G [1988] *Discrete mathematics with ISETL*. Springer-Verlag
CDA Math (R W.Wirtz — head of the project) [1974] *Drill and practice* Washington D.C 20036
Dreyfus, T. [1990] Advanced Mathematical Thinking, in Nesher. P and Kilpatrick, J (eds), *Mathematics and cognition* Cambridge University Press. Cambridge
Erlich, Amos [1989] Periods in Ducci's n-Number Game *The Fibonacci Quarterly* (to appear)
Lakatos I. [1976] *Proofs and refutations* Cambridge University Press Cambridge
Papert, S [1980] *Mindstorms. children. computers and powerful ideas* Basic Books, New York
Polya, G. [1945] *How to solve it*. Princeton University Press, Princeton
Schwartz. J L , Yerushalmy, M. and Education Development Center [1985] *The geometric supposer*. Pleasantville. Sunburst Communications Inc
Skemp, R R [1971] *The psychology of learning mathematics* Penguin Books Harmondsworth, England
Stewart, I. [1987] *The problems of mathematics*. Oxford University Press. Oxford