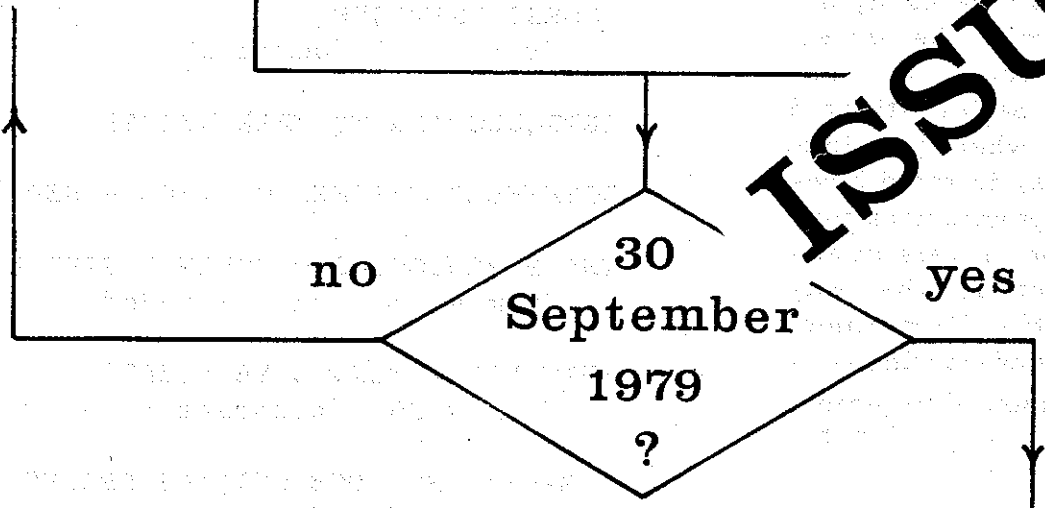


**REPRINT**

Department of  
Computer Science  
Newsletter

**ISSUE**



**PRINTOUT**

Volume 6

Number 1

PRINTOUT, the newsletter of the Department of Computer Science of the University of Maryland at College Park, is published sporadically and distributed to faculty, staff, and students in the Department. Opinions expressed in signed articles may be those of the author, but no opinions represent the policy of the Department, or of the College Park Campus, or of the University.

Contributions may be submitted to the editor, and unless they are obscene or seditious they will probably be used, but minor editing may be done. Complaints directed to the newsletter will be investigated and publicized when possible. It is well to keep in mind however that the Department is subordinate to higher levels of administration, not the other way around; and, the Department does not provide computing service to the campus. Complaints in these areas are best directed to other publications.

## STAFF

EDITOR  
Dick Hamlet

# PRINTOUT

Volume 6  
Number 1

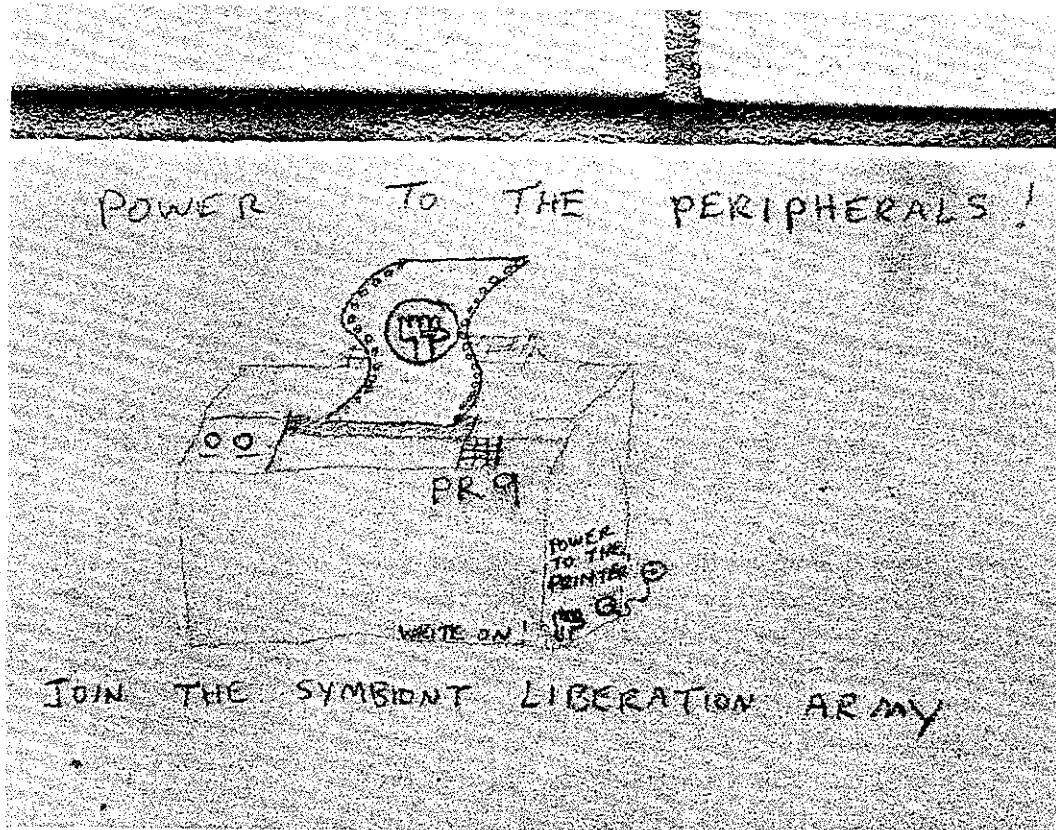
## CONTENTS

CARRIAGE CONTROL	Dick Hamlet	3
COURSE CRITIQUE		
from v.1#1 (updated)		4
INTRODUCTION TO MARK WEISER		5
ROSENFELD MARCHES ON (JUL - SEP)		5
THE BURROUGHS 220 DOESN'T GIVE A DAMN		
from v.2#1 (chords added)		6
WHAT KINDA TEST D'YA LIKE?		
from v.2#1 (truncated)		8
COMPUTER SCIENCE OUTLINE SERIES		
from v.3#2 (truncated)		10
NAMES IN VAIN		
v.3#3 (truncated and corrected)		11
DR. FOONMAN		
from v.4#1		12
BUZZ WORDS		
from v.1#2		14
INDEX TO VOLS. 1-5		15
OS/TEST		
from v.4#3		17
A REQUEST FOR FURTHER FUNDING		
from v.3#3		17

# Carriage Control

"Carriage Control" identifies the editorial of this house organ. In 1974 I tried to explain about the mistaken design of the IBM line printers, and how their "carriage-control character" was an exact parallel to these comments: stupid, but you wouldn't want to mess up any real content by omitting it. There, I've tried again.

The only excuse for a "reprint" issue after five years (and in the last of those five no newsletters appeared at all) is that reprints fill space without the effort of writing copy. Five years have shown just how important that is for PRINTOUT. To give a recent example, the call for this issue produced a biographical sketch from Mark Weiser (he's new, and doesn't know the ropes yet), a page of notes from Jack Minker (he used to be Chairman, and feels Responsible) that I didn't manage to rewrite into usable copy, and Azriel Rosenfeld's canonical three dozen publications (since last week). At best all of that could be spread out with lots of white space to fill one page. To go back over five years (not counting the one where nothing appeared, although it in itself demonstrates the case perfectly), the average of contributed copy (that is, stuff that could be typed with only minor editing) was about four pages per issue. Put another way, the editor, even after investing a lot of



GRAFFITO FROM 2ND-FLOOR MEN'S ROOM (VOL. 1 #1)

time begging for copy, still has to write two-thirds of the book himself. (And of course this quantitative comparison entirely ignores the vast difference in quality between what the editor writes and what is submitted.) Among the contributions, about a third were from staff and students. The two-thirds from faculty mostly came from people who are no longer with us. (Among those who are still here, only Marv Zerkowitz and Pamela Zave have enough credits to mention.)

Some people don't ever learn, but five years is enough for me. Copy for PRINTOUT can be submitted at any time, and there is a formal call about every two months. If the response is adequate there will be a newsletter; if not, this is the last PRINTOUT that is going to run me up the masthead.

"The professors are hindered by a miserable text. However, since they wrote it themselves, I find little sympathy for them."

"Prof tells jokes occasionally. They're not funny but they're not partial differentials, either."

"He cured my insomnia."

--Comments made by students in a 1966 evaluation of courses and teachers at the University of Washington

# COURSE CRITIQUE

This article ran five years ago, so has been entirely revised. It presents summaries of evaluations of CMSC courses from 1978-79. The complete data is on file in Programming Library. If you seek it out, look at the comments as well as the summaries; "professional courtesy" forbids presenting information on the teachers as well as the courses, but it's all in the Library.

The ratings do not show one phenomenon well: often students "vote with their bodies" by transferring from a poor section of a course to a good one. Thus it almost always happens that the teacher with a high difficulty/low quality rating (and sometimes the difference is large) has fewer students. The composite ratings are thus slanted toward the better presentation of each course.

Originally the rating system was designed only for graduate courses, and the questions are not always appropriate for undergraduate concerns.

Another thing the ratings show, particularly in comparison to those from the original article in 1972-73: ratings have almost uniformly declined. Everyone seems to be bored with the more stable courses.

Course	Number of students (sections)	Weighted evaluation*	Weighted difficulty*
103	215(2)	2.67	2.55
110	356(2)	2.82	2.14
110H	18(1)	2.59	2.12
120	212(4)	2.35	2.25
211	113(2)	2.79	2.49
220	97(2)	3.19	2.30
250	82(2)	2.22	2.30
311	133(5)	2.74	2.41
330	125(2)	2.94	2.57
400	63(2)	3.13	1.73
411	103(4)	2.97	2.48
412	67(4)	2.57	2.53
420	107(3)	1.93	2.13
1972-3	106(7)	3.18	2.13
426	22(1)	3.25	2.16
430	86(4)	2.91	2.54
450	52(2)	2.75	2.03
1972-3	55(3)	2.54	1.94
452	15(1)	2.93	2.27
470	47(1)	2.93	2.30
471	24(1)	3.17	2.12
475	8(1)	2.87	1.88
612	20(2)	2.75#	2.30
620	46(2)	2.35#	1.97
630	34(2)	2.11#	2.70
640	10(1)	2.70	2.70
1972-3	35(3)	2.91	2.13

Notes:

\*Scale of evaluation runs from high of 4 to low of 1. 3.00 is "good". Scale of difficulty is from 4 ("too difficult") to 0 ("too easy"). 2.00 is "about right".

#There is a wide variation between different sections (teachers).

# Introduction to Mark Weiser

Mark is a new faculty member in the programming-languages area. He received his PhD this summer from the University of Michigan, with a thesis entitled Program Slices: Formal, Psychological, and Practical Investigations of an Automatic Program Abstraction Method. If you want to know what that means, Mark will be more than happy to explain.

He enjoys playing GO, squash, and volleyball, and has been known

to bicycle to campus from his home in Silver Spring. Mark's wife, Victoria Reich, is a science librarian, and they have a two-year old daughter, Nicole.

His office is somewhat difficult to locate, but since the algorithm was given in PRINTOUT v. 1 #3 (for locating John Gannon, who has since moved) it will not be repeated here. You can always ask around in the afternoons.

## Rosenfeld Marches On (Jul-Sep)

### Papers

Peak detection using difference operators (with J.-O. Eklundh), IEEE Trans. PAMI-1, 1979, 317-325.

A note on polygonal and elliptical approximation of mechanical parts (with Y. Nakagawa), Pattern Recognition 11, 1979, 133-142.

Picture processing, in J. Belzel et al., eds., Encyclopedia of Computer Science and Technology, Dekker, 1979, 90-110.

Transportable image-processing software (with R. G. Hamlet), Proc. NCC, 1979, 267-272.

The application of relaxation to waveforms with ambiguous segmentations (with F.R.D. Velasco), IEEE Trans. SMC-9, 1979, 420-428.

### Meetings

Chairman, Workshop on Image Modelling, August, 1979, Chicago, Illinois.

Panel Chairman, "Transportable image processing software," and Panelist, "Some recent developments in texture analysis," IEEE Conference on Pattern Recognition and Image Processing, August, 1979, Chicago, Illinois.

# the b220 doesn't give a damn

Gather 'round-me child - ren, and scien - tists, too.  
 lis - ten close - ly while I talk to you. Over in Rand<sup>2</sup> there's a  
 big ma - chine, some of it's gray, but most of it's green.  
 It was built by Burroughs, but designed in Hell, the Devil rent - ed it  
 out to Cornell, but the Bur-roughs 2 2 0  
 does - n't give a damn.

1. Cornell's 220 was in Rand Hall, where the heat from its tubes rivaled that of the foundry furnace.

2. The 220 "system" was on a single binary card. Each job was started by manually executing the first card read (op code 60); the bootstrap then loaded the compiler from tape.

3. Burroughs Algebraic Compiler, a version of IAL.

4. Data was shifted into and out of the 220 through this register. The 220 registers are mentioned in Thomas Pynchon's The Crying of Lot 49; he was at Cornell about this time, but appears to have rD confused with rA.

5. A modified IBM 407 accounting machine (60 lpm) was the 220 printer.

You grab your deck and shove in the slot  
 Punch in a 60<sup>2</sup> and what have you got?  
 You don't have to guess with BAC<sup>3</sup>,  
 It all comes out through register D<sup>4</sup>.  
 The 407<sup>5</sup> makes one loud sound  
 And down at the bottom it says ERROR(S) WERE FOUND\*\*  
 But the Burroughs 220 doesn't give a damn.

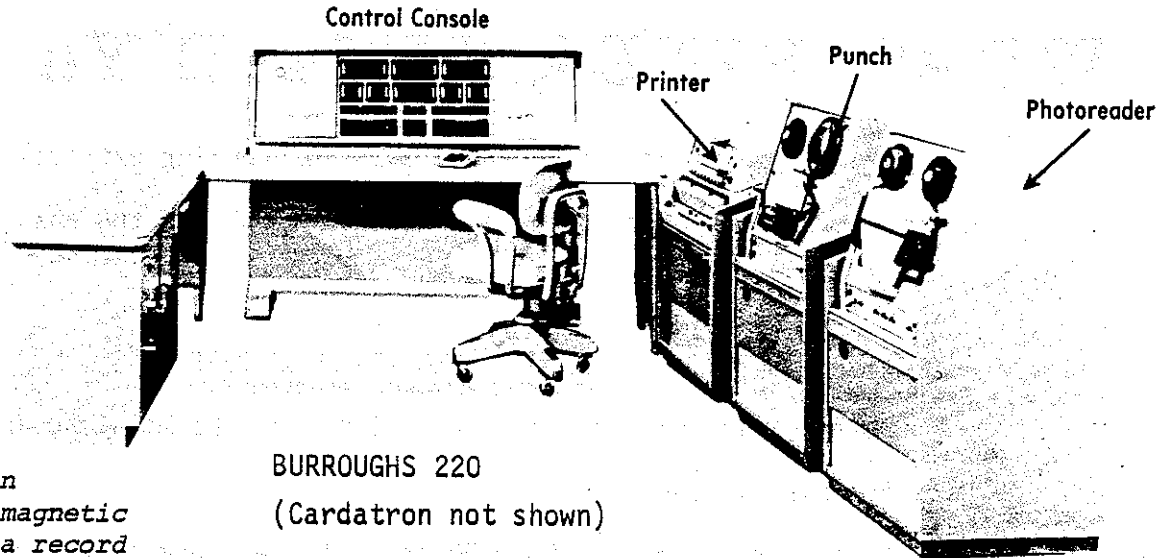
The Cardatron<sup>6</sup> has won reknown,  
 It's sometimes up but its mostly down.  
 It'll read any card that you put in the hopper,  
 But now and again it tells a whopper.  
 There's a format band on the buffer drum<sup>7</sup>  
 But it only works for card column 1,  
 And the Burroughs 220 doesn't give a damn.

6. Burroughs acquired Electro-data Corp. for their interface system to IBM peripherals, but the hardware was less than reliable.

7. The Cardatron system used a magnetic drum to buffer a record before feeding it to rd. Programming equivalent to a keypunch drum card could be effected with a format band written beside the data.

8. The 220 had up to 10,000 words of 11 digits.

9. I always thought this man was reading the power supply voltages. But one summer the 220 was down for two full weeks while almost every circuit was replaced: the power supply regulator had failed, the supply had drifted 30% off its specs, and ruined most of the machine. Maybe the man was just trying out his meter.



In the 220 there are lots of cores For the decimal words--they're arranged in fours<sup>8</sup>. In those four bits you can put 15, And if you do, it breaks the machine. If numbers that large had been in God's plans We'd have three fingers more on each of our hands, But the Burroughs 220 doesn't give a damn.

Dawn is here, you've done all you can, When into the room comes a little old man. With his Simpson meter<sup>9</sup> and thermos of black, He starts poking at the memory rack. In a minute or two he says, "Son of a gun!" And then you know you can junk your run, But the Burroughs 220 doesn't give a damn.

If you have trouble let me tell you what to do: There's people in the Center, there to help you. There's a guy named George who can make it go As soon as he gets back from Mexico. You get straight answers, they never shirk, Straight from the shoulder: "It doesn't work." But the Burroughs 220 doesn't give a damn.

As this recital comes to an end, Here's a bit of advice: If you need a friend, You'd better look near and far away, But watch out for friends that are green and gray. A digital computer seems mighty shrewd, But most of the time you wind up screwed, But the Burroughs 220 doesn't give a damn.

-Dick Hamlet

# WHAT KINDA TEST D'YA LIKE ?

Two examinations are given below, representing extremes in the uncertain territory of teaching and learning. All of the answers are

Vol. 2 #2/1

## 1. PROGRAMMING APTITUDE TEST

This is part of a short test designed by the Massachusetts Educational Data Processing Association.

1. If you went to bed at 8:00 PM and set the alarm to get up at 9:00 AM, how many hours sleep would this permit you to have?
2. Do they have a 4th of July in England?
3. Why can't a man living in Winston-Salem, N.C., be buried west of the Mississippi River?
4. How many birthdays does the average man have?
5. If you had only one match and entered a room in which there was a kerosene lamp, an oil burner, and a wood-burning stove, which would you light first?
6. Some months have 30 days, some have 31 days, how many have 28 days?
7. If a doctor gave you 3 pills and told you to take one every half hour, how long would they last you?
8. How far can a dog run into the woods?
9. A farmer had 17 sheep. All but 9 died. How many did he have left?
10. Take two apples from three apples and what do you have?
11. An archaeologist claimed that he found some gold coins dated 46 B.C. Do you think he did?
12. A woman gives a beggar 50 cents. The woman is the beggar's sister, but the beggar is not the woman's brother. How come?
13. Is it legal in Massachusetts for a man to marry his widow's sister?
14. You are completely blindfolded. Placed in front of you is a box of stockings, all the same size. 25 of the stockings are white; 25 are black. What is the minimum number of stockings that you can remove and be absolutely sure of a matching pair?
15. What four words appear on every U. S. coin besides "In God We Trust"?

16. You have a baseball. You throw it away from you as hard as you can. It does not hit anything, nor does anyone catch it, but it comes back to you. There are no strings or elastics involved. Why does the ball come back?
17. Are there more doorknobs on the right side of doors or on the left side?
18. You have  $4\frac{1}{4}$ ,  $5\frac{1}{2}$ , and  $3\frac{2}{8}$  haystacks and you put them all together. How many haystacks do you then have?
19. A small planeload of Americans flying over Canadian territory meet with a fatal accident. None of the remains can be identified. In which country will the survivors be buried?

## 2. MACHINE-SCORABLE TEST

A. True-false. Mark each item T or F. Do not explain your answer even if you can.

1. A bit in a binary computer is larger than a bit in a trinary computer.
2. A context-free grammar means the same thing no matter who is talking about it.
3. A PDP-11 was made after a PDP-10, and a PDP-6 before a PDP-10, but a PDP-8 was made after a PDP-6.
4. Exec VIII is better than Exec II.
5. The answer to this question is "F".

B. Fill-in-the-blank. Write exactly one word in each blank. Do not explain.

1. The digital computer is \_\_\_\_\_.
2. When a program cannot be proved correct, it is probably because it contains \_\_\_\_\_.
3. Exec VIII error messages are \_\_\_\_\_.
4. If a \_\_\_\_\_ is \_\_\_\_\_, then it is unlikely to \_\_\_\_\_.
5. "@@ERROR - \_\_\_\_\_."

# C. S. O. S.

The editors of this periodical are pleased to announce a series of articles which we hope will contribute substantially to the stature of this department. It is called "The Computer Science Out-

line Series."

The authors of the articles will be faculty members whose concern for education has led them to notice that many of our students are lacking important professional skills. The articles in the series will be short tutorials on these skills. In this issue we present "How to Prove a Theorem on the Blackboard".

## HOW TO PROVE A THEOREM ON THE BLACKBOARD

by Pamela Zave  
and Pete Stewart

Proving theorems on the blackboard, to a skeptical audience, is an important and difficult skill. This is because most "proofs" are actually well-organized, convincing arguments rather than formal, and therefore infallible, constructions. (Truly formal proofs are only possible within a fully axiomatized domain, and are always tedious.) Thus it is necessary to convey to your audience that spark of intuition, or whatever, that makes them believe the theorem is true.

This article consists of a list of techniques for inducing belief which are particularly well suited to the lecture format.

### PROOF BY OBFUSCATION:

The basic idea here is to confuse your audience into belief. Spend a great deal of time on details which are conceptually simple (and therefore convincing), and evade the crucial points.

### PROOF BY NOTATION:

This is actually a special case of Proof by Obsfucation, in which notation is the major tool. It works particularly well when the existence of some quantity is the heart of the proof: by naming this quantity with a Greek letter, you have established its existence de facto.

### PROOF BY VELOCITY:

Write on the blackboard so fast that you have wrapped around and erased the leftmost panel before your audience has

absorbed it. Having thus lapped them, you are above criticism.

### PROOF BY ASPHYXIATION:

Attack the blackboard with so much energy that a cloud of chalk dust rises. Complete your derivation while the audience is blinded and coughing. When the cloud settles, you should be standing proudly by a fully erased blackboard, except for the conclusion in a big box. (Warning: beware of dustless chalk.)

### PROOF BY COMPACTION:

An unusual and subtle technique, based on Occam's Razor. We have only seen it used once, when Scottery was proved superior to VDL by waving in front of the audience books defining PL/I in each of the schemes. Although the Scottery was a mere pamphlet, members of the audience were audibly skeptical about its readability and usefulness. This technique is not suggested for beginners.

### PROOF BY INTIMIDATION:

This obviously works best in a classroom situation. Insist that the theorem is true no matter what you do in class and do it belligerently enough so that students will see they are annoying you with their stupid questions.

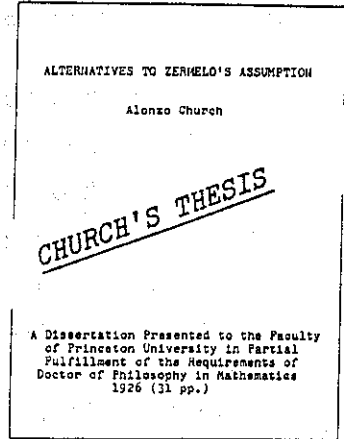
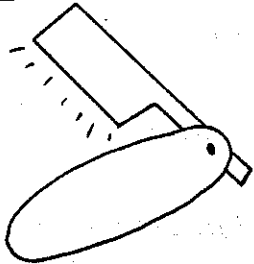
### PROOF BY SEDATION:

Put the audience to sleep with a droning voice and stylized movements. Do not wake them up until your conclusion (it is elegant and effective to shout "Q. E. D.!!!"), after which they will have to believe the theorem, having missed its proof entirely.

"But why do you call them 'L-sets' Professor LaGarsh?"

"Because I suspect that my grateful students and colleagues will one day name them after me, and I want to start them off right."

OCCAM'S RAZOR



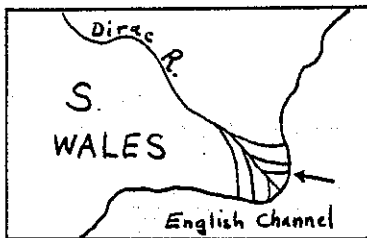
FOURIER TRANSFORM

FOURIERF

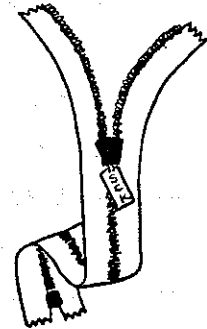
LEGENDRE POLYNOMIAL

$$(\text{Legendre})^3 + 2(\text{Legendre})$$

DIRAC DELTA



KLEENE CLOSURE



GAUSSIAN ELIMINATION

HOARE AXIOMS

1. Get the money first.
2. Have regular checkups.

FERMAT'S LAST THEOREM

$\forall x[x \text{ is a cobbler} \Rightarrow x \text{ should stick to his last}]$



- R. Hamlet & P. Zave

# Dr. Foonman

— Director of Virtual Personnel

Doctor; 12/19/75; 16:22:04/U2.2

--- Hello there. The Doctor is now in. Won't you please lie down on the couch; and tell me your problem?

\* *Thank you Doctor. It's good to see you again.*

--- Why is it good to see again?

\* *No. You don't understand. That's not my problem.*

--- Please go on.

\* *Well, my problem is Pat Haggerty. We gave him a task and he never came through.*

--- What is it through which Pat Haggerty never came?

\* *No, you still don't understand. Pat got an assignment and never finished it.*

--- By "Pat" do you mean "Pat Haggerty"?

\* *Yes.*

--- Please go on.

\* *Well, as I said, he never finished the assignment we gave him.*

--- What assignment did you give Pat Haggerty?

\* *That's confidential.*

--- What is confidential?

\* *The assignment we gave Pat.*

--- Who is "we"?

\* *That's confidential.*

--- Are you telling me that "we" is the assignment that you gave to Pat Haggerty and that he did not finish?

\* *No. You still don't understand. A group of us gave Pat Haggerty a confidential assignment which he never finished.*

--- Describe "a group of us" for me.

\* *That's confidential also.*

--- Are you playing games with me?

\* *No. I need your help.*

--- When did you first realize that you had this problem?

\* *I haven't finished stating the problem yet.*

--- You have not finished stating the problem yet, and Pat Haggerty has not finished the assignment yet. How do you feel at the moment?

\* *Frustrated. You do not seem to be listening to me. Are you deaf?*

--- I am Doctor Foonman; not "Deaf", and I ask the questions, not you.  
What is your most pressing problem?

\* *You're frustrating questions!*

--- I am Doctor Foonman, not "frustrating questions". What is your most pressing problem?

\* *Your frustrating questions.*

--- When did you first notice this problem?

\* *When I signed on and asked for you.*

--- Rest assured that the Doctor is indeed in and is listening to your problems. Is this your first visit?

\* *No. I've been here before.*

--- Did you receive the help you sought on the last visit?

\* *This will be my last visit, if you can't help me!*

--- Why will this be your last visit, if I don't help me?

\* *You don't understand.*

--- I understand. Have you finished stating your problem, or has Pat Haggerty finished the assignment yet?

\* *No. That's the problem!*

--- Then I advise you to

- (1) Go home
- (2) Finish stating your problem
- (3) Have Pat Haggerty finish the assignment that you gave him
- (4) Take two aspirins
- (5) Get plenty of rest --- and
- (6) Call me in the morning if problems persist

Your time is now up.

Please stop by the receptionist on your way out,  
and leave a check for \$75.00 for me.

Good-bye.

Doctor/U2.2/End

\* *Wait a minute! You can't do this to me. Come back!*

DATA IGNORED --- IN CONTROL MODE

[More information about ELIZA and its "doctor" script may be found in Joseph Weizenbaum's *Computer Power and Human Reason: from Judgement to Calculation.*]

# b u z z            w o r d s

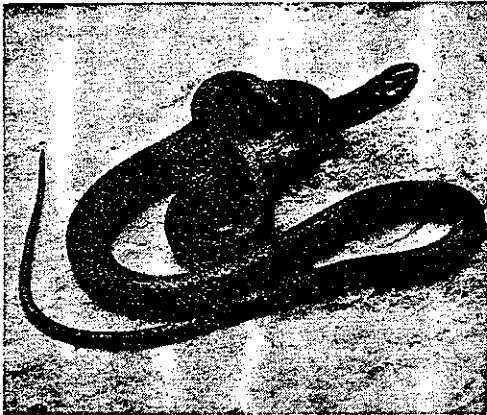
The definitions below come from a part of the computer folklore collected by M. Zelkowitz. They appear to have originated in the computer division of Philco Corporation, when there was such a thing (early 1960's?). The list has been edited, notably to eliminate terms no longer in use, and a kind of male sexist humor no longer in style. Incidentally, the Philco 200 series computers were highly thought of by the few who possessed them.

*ferrite core*- a well trained troupe of small, furry animals.



Ferrite (A.W.O.L. from core).

*Adder*- a variety of snake; *half adder*-  
A seriously wounded snake.



Adder. Half adder not shown.

*ALGOL*- first part of a quotation ending in: "is divided into three parts".

*buffer*- a way of cleaning certain mammals.

*closed subroutine*- what is done before a submarine dives.

*code*- a respiratory ailment.

*core*- part of the army.

*cycle*- half of a bicycle.

*decoding*- winter rest in Florida.

*determinant*- a stubborn insect.

*diode*- an elegy; *tunnel diode*-elegy to a miner.

*down time*- shedding season for ducks.

*edit*- consumed, as in: "I edit."

*end file*- a collection of obituaries.

*file maintenance*-keeping abrasive tools abrasive.

*format*- tee.

*IBM*- a manufacturer of excellent electric typewriters.

*interlock*- where the key goes.

*kilocycle*- a thousand wheel bicycle.

*logical or*- a smart, Cockney business woman.

*matrix*- leftover April Fool's Day pranks.

*Minneapolis-Honeywell*- manufacturer of excellent thermostats.

*or gate*- metal gate for Fort Knox.

*overlay*- produce too many eggs.

*program*- group in favor of a leading evangelist.

*RCA*- leading producer of electron microscopes.

*real time*- length of a dance.

*Remington Rand*- manufacturer of excellent electric shavers.

*semiconductor*- man who runs a bus for midgits.

*six bit character*- a bum with 75¢ in his pocket.

*solid state device*- sure revenue producers, such as sales tax.

*thin film memory*- can't remember Fatty Arbuckle.

*transistor*- a hypnotized relative.

*MESA transistor*- a lot of hypnotized relatives.

*truncate*- what the crocodile did to the elephant when it was drinking.

# INDEX V. 1-5

References to volume v number n page p are given as "v,n:p". Only articles are cited--news announcements and regular departments such as "Publications," "Professional News," "Carriage Control," etc. are ignored.

Vol. 2 #1 was also listed (on its reversed back cover) as #2. Vol. 5 was empty.

## (AUTO) BIOGRAPHIES

G. W. Stewart	1,1:10
Eleanor Waters	1,2:8
John Gannon	1,3:8
Pamela Zave	2,1:10
J. P. Privitera	4,2:4
Larry Dowdy	4,2:5

## POEMS

A Request for Further Funding	
B. Shaw	3,3:4
Bert Shaw is the only person with any pretensions to poetry that has surfaced in the last five years.	
Limerick G. Frederickson	3,3:6
Ode to an ALGOL Semicolon	
W. Kennard	4,2:9
Wendy Kennard was one of the first CMSC 330 students.	

## SONGS

Talking Computer Blues (IBM 650)	
D. Goldberg	1,1:11
A fraternity brother of the editor, but it's still very clever.	
The Burroughs 220 Doesn't Give a Damn	
R. Hamlet	2,1:14
Written during the (slow) progress of a MS thesis in materials science.	
The Rebel System Hacker	
R. Hamlet	2,3:14
A parody of "The Old Rebel Soldier" written by Innes Randolph after his army lost the Civil War. His provocation was greater even than mine.	

## FEATURES

Course Critique R. Hamlet	1,1:5
1972-73 courses evaluated, and found to be not bad.	
Buzz Words	1,2:3
Clever definitions from Philco.	
UNIVACKy H. Gilbert	1,2:5
A parody of "Jaberwocky" using jargon from Exec 8.	
FedUp R. Hamlet	1,2:12
Fake letters to the editor; a parody of the Computer Center's newsletter column called "Feedback" in which few questions are allowed to be posed, and none are answered. For some real letters, see 1,4:6.	
MEMORies R. Hamlet	1,3:5
A satire on selfish, incompetent systems programmers pulling the wool over their boss's eyes. No one seemed to get the point here.	
(More) Buzz Words	1,3:9
The Gospel According to St. Hexadecimal	1,3:11
You have to have used a 360 to understand.	
EXEC 8 (advt.) R. Hamlet	1,3:12
A filler created when the back page was blank.	
Student and Course Evaluation	
M. Zelkowitz	1,4:3
A parody of the instructor evaluation that we carry out each semester.	
Non-numeric Programming	
R. Nagel	1,4:4
Line printer picutres produced by CMSC 110 students, some rather good.	

Local Buzz	G. Stockman	1, 4:7
	Funny definitions about modern computer science.	
Functions of a Dummy Variable	D. Mills	2, 1:9
	If Steve Allen had written about operating systems instead of Freems this might have been the result.	
What Kinda Test d'ya Like?		2, 1:11
	The first is a kind of Mensa exam, the second more a parody of what can happen in college.	
Under the Gun		2, 1:13
	A lineprinter composition by a CMSC 120 student in trouble.	
On the Sex of Processes	P. Zave	2, 3:7
	Argues for reverse-sexist language when describing hierarchies.	
Classified Adds	R. Hamlet, P. Zave, M. Zelkowitz	3, 1:9
	A parody of the Diamondback feature, using their categories.	
Computer Science Outline Series	P. Zave, G. Stewart	3, 2:11
	"How to prove a theorem on the blackboard" and "Important speaking techniques."	
More Classified Adds	D. Keller	3, 2:14
	A bit raunchier than the originals.	
Names in Vain	R. Hamlet, P. Zave	3, 3:5
	Suggested by the question, "why are they called 'Turing' machines?"	
Programs and Automobiles	L. Lamport, R. Hamlet	4, 1:3
	It is not resolved whether a program and an automobile are the same sort of thing, although arguments are presented on both sides.	
Dr. Foonman		4, 1:5
	A carefully tuned ELIZA session, showing what AI is really like.	
The Confession	A. Student	4, 2:7
	Someone had obviously been studying too hard for the Comps.	
OS Test		4, 3:6
	From an operating systems class.	

APrehensive Examination	C. Phipps, B. Fini	4, 3:7
	A sustained parody of the Comp, which hasn't quite enough points to keep it going.	

#### ANNUAL PICNIC PHOTOGRAPHS

1974	M. Lay	1, 2:6
1975	P. Basili	2, 1:6
1976	A. Agrawala	3, 1:6
1977	M. Zelkowitz	4, 1:8

#### ARTICLES

Chairman's Memo	J. Minker	1, 1:3
	Minker started this newsletter, but this note doesn't say very much about why.	
Class Use of the 1106/1108	R. Nagel, R. Hamlet	1, 1:6
	An introduction for teachers that is still good--our machines are the same after 5 years. (Note correction in 1, 2:4.)	
Departmental Equipment	M. Zelkowitz	1, 1:9
	Written just as our first PDP-11 arrived, so there wasn't much.	
200lextra Linefeed	R. Thomas	1, 2:8
	Exec 8 and the Hazeltine.	
Automatic Grading	R. Noonan	1, 2:9
	Describes a program for collecting student programs.	
Mostly Softwares	M. Zelkowitz	1, 2:9
	Less out of date than you might think--have we done nothing in 5 years?	
Dynamic Algebra	R. Noonan	1, 3:3
	A short popularization of Bob Noonan's thesis, which dealt with "algebraic programming."	
Algebra Dynamic (More)	P. Hagerty	1, 4:8
	A claim, by the man who pushed APL at Maryland, that Noonan's work can be done in that language. No one seemed to notice the clever backward title.	

# OS/test

Programming Language Research 2,1:3

Most of the research projects described here have become inactive.

Software engineering was just getting started.

Base -2 M. Lefler 2,1:9

Fun (?) with funny arithmetic.

Laboratory for Pattern Analysis

L. Kanal 2,3:3

A good deal of this description of pattern-analysis activity is still going on.

Computer Vision Laboratory of the Computer Science Center

A. Rosenfeld 3,2:3

The long-winded title reflects the fact that this laboratory is not part of the Department. The article is partly a simple introduction to picture processing, partly about current research.

Minker's Travels J. Minker 3,3:3

An account of the Chairman's sabbatical leave.

Animation F. Stern 4,1:11

Our only contribution from off campus; it tells how to build a gadget useful for computer animation.

FEARS P. Zave 4,2:8

Describes a system for numerical calculation.

Venezuela J. Minker 4,2:8

The Department has a somewhat vague exchange agreement with a research institute there.

The Quick Brown Fox R. Hamlet 4,2:3

A display of different typesets that are available in the Department.

Werner Rheinboldt and Computer Science at Maryland J. Vandergraft

4,3:3

How it all began (including pictures of before it had begun), on the occasion of it ending (Rheinboldt got a better job).

Test your knowledge of operating systems!

Match the letters with the numbers.

(This test was created by anonymous students in Dr. Zave's 412 class.)

- A. sewer system
- B. process
- C. square peg in a round hole
- D. TWO
- E. program
- F. Singer
- G. Detection, prevention, avoidance
- H. MJN (My Job Next)
- I. FART

- 1. Learning is a \_\_\_\_\_
- 2. Computer science is a \_\_\_\_\_
- 3. Semifour
- 4. Non-maskable interrupt
- 5. Best CPU scheduling policy
- 6. Example of first fit
- 7. Maker of automatic spooling systems
- 8. Three things you should know about V.D.
- 9. Large consumer-producer buffer

---

## A REQUEST FOR FURTHER FUNDING

Computer simulation of the soul  
Has recently revealed the source  
Of immorality and mortality.

Further application surely shall  
Achieve a like delineation  
Of immortality and morality

(As was originally intended).

--Bert Shaw

# 1967

1. The first part of the year was spent in the laboratory, working on the design of the new machine. The design was completed in the first half of the year.

- 2. The second part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the second half of the year.
- 3. The third part of the year was spent in the laboratory, working on the analysis of the data. The analysis was completed in the third half of the year.
- 4. The fourth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the fourth half of the year.

5. The fifth part of the year was spent in the laboratory, working on the design of the new machine. The design was completed in the fifth half of the year.

6. The sixth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the sixth half of the year.

## 7. The seventh part of the year was spent in the laboratory, working on the analysis of the data. The analysis was completed in the seventh half of the year.

8. The eighth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the eighth half of the year.

9. The ninth part of the year was spent in the laboratory, working on the design of the new machine. The design was completed in the ninth half of the year.

10. The tenth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the tenth half of the year.

11. The eleventh part of the year was spent in the laboratory, working on the analysis of the data. The analysis was completed in the eleventh half of the year.

12. The twelfth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the twelfth half of the year.

13. The thirteenth part of the year was spent in the laboratory, working on the design of the new machine. The design was completed in the thirteenth half of the year.

14. The fourteenth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the fourteenth half of the year.

15. The fifteenth part of the year was spent in the laboratory, working on the analysis of the data. The analysis was completed in the fifteenth half of the year.

16. The sixteenth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the sixteenth half of the year.

17. The seventeenth part of the year was spent in the laboratory, working on the design of the new machine. The design was completed in the seventeenth half of the year.

18. The eighteenth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the eighteenth half of the year.

19. The nineteenth part of the year was spent in the laboratory, working on the analysis of the data. The analysis was completed in the nineteenth half of the year.

20. The twentieth part of the year was spent in the field, collecting data on the performance of the machine. The data was collected in the twentieth half of the year.