

Needless Hell is a horrible place! All those blondes try to impale me with their spiked heels - figures I'd pick the best-dressed place on campus to get lost in. I can't take it any more! Boo hoo hoo! Help me, dan! Signed, Monkey.

-overheard in Morse code tapped on a heating duct in Needles Hall by C. of G.

math NEWS

Volume 36 Number 9

Processing cost: \$0.04

Friday November 23, 1984

What I Did And Didn't Learn In 2B

Didn't

- (i) Who voted for Tom Allison as Fed. President? Who voted for him for a second term? But more importantly, is there anyone who will ever vote for him again?
- (ii) Aside from the aesthetic pleasure and intellectual stimulation derived from studying Group Theory, is it useful for anything?
- (iii) Who are the people that write on bathroom walls, and why don't they come out of the stalls to share their wisdom? Has anyone ever considered writing a book, "Portrait Of The Bathroom Wall Philosopher"?

Did

- (i) If anyone is looking for an elective and has any interest in Astronomy, **don't take PHYS 250**, take SCI 238.
- (ii) Never pull two all-nighters in three nights, especially if you don't have 18 hours to spare on the fourth night.
- (iii) If you have a big old German fellow for a landlord whose biceps are bigger than your thighs, and who has a bad temper, always follow the silly-assed house rules he posts in the kitchen.
- (iv) Never discuss nuclear disarmament with anyone over 40 who votes conservative, unless you don't care if your parents disown you.
- (v) Did you know that a golf ball will travel farther in air than in a vacuum, and at a 10° incline rather than 45° ? For more information, contact Hugh Morrison in the Physics Dept.
- (vi) Finally, and most important, pay attention to administrative details while you are here. This can avoid unpleasant situations like walking into your PHYS 256 (or any other) midterm 40 minutes late (PHYS 256 has a 55 minute midterm). This is not a nice feeling.

Pink Day Perspective

Wednesday, November 14th: Pink Day, signaled by the reappearance of a certain rather large pink tie (although Imprint got its significance sort of confused ...). And if you happened to be in the Math building on that day, especially on the 3rd floor, you may have noticed a profusion of pink—especially on such people as Karen Thompson and Renée Frisc, who were the first and second pinkest people, with 31 and 30 items respectively (and the guys in MathSoc did enjoy the peeling show!)

Pink people enjoyed free hot drinks (over 150 given out), 75 pizzas (the first 25 disappeared so fast that the servers never got any!), and free pieces of cake. Special mention goes to Pete Zeransky for the "piggist pink person"—he wasn't too proud to take a specially cut very large piece of the first cake. (Not to mention all those spiritless people who wore no pink and took cake anyways!)

Prizes (tickets which can be exchanged if necessary) were given for the largest and smallest pinkies to Phil Beale and Carol Page (Sorry, but I don't have their measurements). [Ed: Phil was 7.5 cm.]

The day was finished off with a successful video night, featuring "Educating Rita", "Splash", "Body Heat", and "Deathtrap", so that the last pink people finally went home at 3:00 a.m. Thanks to Chris Jones for organising a very Math day, and to all the people who helped out.

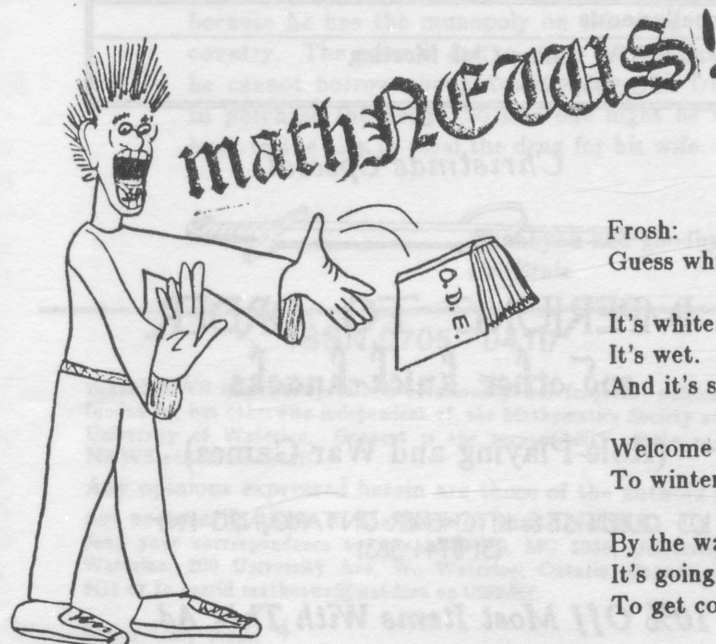
British Petroleum

math NEWS

Dear Editor,

Hi, I just wanted to see my name in **mathNEWS**.

[Ed: To respect the confidentiality of our contributors, we have withheld the author's name]



Frosh:
Guess what?

It's white.
It's wet.
And it's snow.

Welcome
To winter.

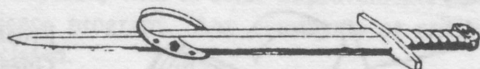
By the way,
It's going
To get colder.

LookAhead

A glance at upcoming events

Math Events
Nov. 24 Vanier Cup Trip
Fed Flicks
8 pm. in AL116 Cost: \$1 feds, \$2 others
Nov. 23-25 Never Cry Wolf
Cinema Gratis
starts 9:30 pm. in the CC
Nov. 28 Jesus Christ Superstar
Dec. 5 A Christmas Story
Computers in Education
Brown-Bag Seminars
All lectures at 12:30 - 1:30 in CPH 3385
Nov. 26 CAD/CAM-Computer-Aided Design and Computer-Aided Manufacturing
mathNEWS dates
Nov. 25 mathNEWS articles deadline
Nov. 26 mathNEWS production night, MC9088. ←
Nov. 30 Next mathNEWS available
UW Arts Centre
Call Humanities Theatre Box Office at 885 4280 for more info
Nov. 26 Primadonna
Nov. 28 Tafelmusik
DCS Courses
Free! One to three one-hour lectures; contact DCS to register.
Nov. 27,29 Introductory SAS
Dec. 3,5,7 Introduction to UNIX
Dec. 10,11,12 UNIX for Documentation
Watsfic
Nov. 29 General meeting
Nov. 30 Games day. Starts at 2:30
Miscellaneous
Nov. 24 UW Games Club Meeting CC 135, 12 to 7

Christmas Special



IMPERIUMS TO ORDER and other knick-knacks

(Role-Playing and War Games)

103 QUEEN ST. S. KITCHENER ONTARIO, N2G-1W1
(519) 744-3831

10% Off Most Items With This Ad

Valid Until December 25 Only

(Maasthead starts on page 10 silly, not here - go away) Irishman, (why are you mad?); and Slash Sinatra. What a list, no wonder we have 10 pages.

Thanks also to our large production/consumption crew. Specifically: not Barb and not Ernie but Barry Turner, they claim he showed up; Center of Gravity - typing proof-reading and a contribution; Paul - typing, layout and prof-reading; Johnny O - typing, writing, proof-reading, ordering bad pizza; Master surgeon Rick - typing proof-reading and most of the layout (but i did the front page, as if you couldn't guess); Richard Lewis - alias Jene (i found this very confusing) - layout, Richard's "first time" and he went "all the way".

Some obscure fellow named Cary Timar showed up and got in the way. Harry F. Davis thinks Cary is famous but then he also thinks i resemble Stephen Lewis (of course i've never seen Stephen Lewis in drag so i wouldn't know).

Dr. Stranglove came by to give complete technical support and sing the Bruces' philosopher song. He still insists on six page issues. (Sexual frustration).

Then there's Jane. She proofreads. She cuts and pastes with delicate hands. She juggles. She even brings smarmies. Most impressively, she tolerates me! You and me Jane, CPR, name the day. (Deb only tolerates me when she loses her voice.)

Once again - Megathanks to all. You've made it fun. I feel like the guy in our cartoon - i cast aside a "fun" AM 270 assignment to be here tonight as SPECIAL GUEST EDITOR. Now, 3AM, i resume that assignment.

dan schnabel

Platter Spatter

Once again I take time out from my busy schedule to keep you up to date on current record releases.

Two debut albums recently out and definitely worth your attention are those by **Alphaville** (*Forever Young*) and **Savage Progress** (*Celebration*). The former is a tasty outing for this German "instant" band. The disc contains "Big in Japan" and "Sounds Like A Melody", and is quite good overall. The latter disc is from a UK foursome who have been around for a while but are just now breaking through. This one features "Burning Bush", "Heart Begin To Beat" and "My Soul Unwraps Tonight." The lead singer Glynnis has a very fluid voice that just sends me.

Also released in the last fortnight is the new LP from local faves **Depeche Mode**. The album features a very cynical outlook on humanity, love relationships, religion, and has some strong statements to make here. Glad to see the boys continuing the instrumental experimentation that they began on their last record. One song even features a solo acoustic piano. Hard to believe from the pioneers of electropop. *Some Great Reward* is worth the purchase and contains "People Are People" and "Master and Servant". Recommended. (Highly - Ed)

See you next week for my 1984 retrospective. Toodle-oo!

Slash Sinatra

More On Job Placement In The SAC

Well, at our last Student Advisory Council meeting we covered a multitude of topics. The first item might be of interest to you regulars out there as well: Grad Placement would like everyone to know about them. The response this term has been good, with applications from both on-campus students and those presently on work terms. Grad Placement is a service not offered by every university, and they want you to start thinking about career goals; they believe every should have 2-3 years to get ready before graduation.

An item of concern to those working for IBM is the uncertainty of last-minute placement. Well, the managers there say that with so many students they can't change the process. (IBM is the largest co-op employer, with 324 Waterloo students there this fall.) All they will say is that senior students in marketing will probably work downtown, while everyone else can expect to be within a ten-mile radius of the new main office at Steeles and Victoria Park. The SAC suggested that IBM give students an idea of the general area (eg. Steeles, Don Mills, etc.) where they will be placed, but we don't know if that will work.

Another topic that was discussed was confidential work reports and the possible standardization of the same. It seems that there are far too many "outstanding" confidential reports (marked by the employer instead of the University). Engineering (boo, hiss!) is trying to pass some new guidelines that restrict confidential reports to a "satisfactory" grade; if the student is really after an "outstanding", he or she can change the topic.

Last but not least, it was brought to the attention of the Math SAC reps that a number of students found their resumes not being sent out, because they were "withdrawn from co-op" or application lists were messed up. The SAC would like to have a "production" list, of all students who applied to a particular job, placed in the Career Information Centre. Concerned students could then check that their resumes were sent. Whether or not this is feasible remains to be seen. As far as mark transcripts are concerned, Coordination has to send out the copy they receive from the registrar at the start of term; if any changes have been made, the student should inform the Department. The SAC suggested that if a transcript causes a student to be withdrawn from co-op a letter should be sent out to confirm this. Official policy holds, however, that it is the student's responsibility to make sure all items in a co-op file are accurate and up-to-date.

While I was coding the questionnaires that circulated at the beginning of term, I noticed that students seemed to be generally pleased with Coordination & Placement, but not so happy about coordinators and visits. In the meeting we discussed the purpose of those visits: primarily to check whether the student is happy with the job, and whether the employer is happy with the student and will

stick with the co-op program. Other things that come up in visits are learning objectives, work reports, and career paths. The usefulness of the visit is really up to the student.

So, you finally made it through this long-winded report! One last word on placement stats: as of Nov. 9/84, there were 699 math students requiring jobs and 924 jobs available, plus some jobs held for second round. This means that even if you didn't get a job the first time around, you still have great chances. Good luck!

British Petroleum

Dr. Ernie and Responsibility

I have to assume that I am responsible for my actions. Therefore, I have a certain responsibility for your actions.

I am certainly not responsible for all of your actions, but if you are influenced by anything that I write (or say or do) then there is some responsibility on my part. This is one of those statements that is clearly true. The problem arises when one attempts to judge my responsibility in any specific example.

This is a dangerous concept. While we can't get away from the responsibility for our actions, we cannot afford to get carried away. How many books would have been written if the author had been unwilling to take the responsibility for his book. Yet, the problem is how much responsibility should the author have?

The question of responsibility goes much deeper. Here is a moral dilemma. Can the actions of either of the protagonists be justified?

In one of the poorer European countries, the wife of a farmer is dying of cancer. There is only one thing that might save her: it is a rare drug only recently developed. The farmer travels to the city to try and purchase some of the drug. The chemist is selling the drug at ten times its normal price because he has the monopoly on this drug in this country. The price is far too steep for the farmer; he cannot borrow enough money from his friends to purchase the drug. So late one night he went back to the city to steal the drug for his wife.

Thankyou and goodbye
Dr. Ernie

ISSN 0705-0410

mathNEWS is a weekly (albeit occasionally less frequent) publication funded by, but otherwise independent of, the Mathematics Society at the University of Waterloo. Content is the responsibility of the mathNEWS editors and staff.

Any opinions expressed herein are those of the authors and not necessarily those of MathSoc or mathNEWS.

Send your correspondence to: mathNEWS, MC 3035, University of Waterloo, 200 University Ave. W., Waterloo, Ontario, Canada, N2L 3G1 or to userid mathnews@watdcsu on USENET.

Editor: Cary Timar &
dan schnabel

Hacker's Hexadecathlon

Wheel wars have been a part of on-campus computing for some time (especially around Stanford and MIT). Waterloo has been spared much of this, but the urge to display knowledge and ability still runs high in many people around this place. To facilitate this, the hexadecathlon has been developed to test all-around hacking ability. There are sixteen events, as follows:

0. 100 Megabyte Core Dump

The object of this event is to warm up the competitors for the other fifteen. Contestants must simply write a program that causes a core dump of at least 100 megabytes of data. All software must be written by the competitor, but he/she [are there any female hackers?—wjj] may make any calls to the operating system kernel or lower levels of the machine if they know them. Points are awarded for size of the dump caused and time required. Bonus points for creating a dump that escapes the operator's notice.

1. Disk Crash

Competitors take turns at crashing the disk heads using only software. Each competitor gets six attempts, with the best one counting toward the final score. Points are awarded for the volume of magnetic oxide removed from the platter surface, with bonus points for scraping right down to the aluminum.

2. 110 Baud Hurdles

The object of this event is to break down international data communication security. Competitors must log on to a VAX in Australia using a specified home computer and 110 baud modem, making every wasted byte count. Points are awarded according to the time taken to complete the event, with penalty points for each warning issued by Datapac. Bonus points are awarded for completing the call through the USSR.

3. Compiler Composition

Competitors write a C compiler on a computer of their choice, according to the specifications in Kernighan and Ritchie. Use of YACC or similar programs is forbidden. Points are awarded for efficiency of code generation and time taken for completion, with bonus points for writing it in a WATCOM language.

4. Terminal Conversion

This preliminary test of hardware skills consists of converting a standard VC303 terminal to a VT-

202 using only parts listed in the most recent Radio Shack catalogue. Bonus points are awarded for full colour T-500 (Telidon) graphics support.

5. Stack Frame Vault

This is a simple event. Competitors must write a recursive program that will blow a limited stack. Points are awarded for the depth of recursion, with bonus points for successful computation of Ackermann(20, 20).

6. Interface Race

Each competitor is supplied with an identical non-standard bizarre peripheral device. Competitors must devise and fabricate the appropriate interface cable. Points are awarded according to the time taken, with bonus points for not using a rainbow-coloured ribbon cable.

7. Datapacman

This title can be read two ways, and both describe this event. Competitors must modify a national database from another city via Datapac. Points are awarded according to the size of the database, the subtlety of the modification, and the length of the routing of the call. Points are deducted for each warning from the Bell Police. Bonus points are given for using BC Tel and for creating a new database.

8. Smallest Useful FORTRAN Program

At last, a software challenge. Competitors must write a useful program in FORTRAN (the judges decide what is useful and what is not). Points are awarded for the shortness of the program, as well as for the time taken.

9. Most Obscure APL One-Liner

Competitors must write a one-line APL function (the number of arguments is up to the competitor). Points are awarded for the number of fellow competitors who cannot understand the code, provided at least one judge can decipher it.

A. Single Board Design

Given a specified CPU, a hex keypad and an array of 7-segment displays, competitors design and fabricate a single-board computer to specifications handed out at the beginning of this event. Points are awarded for time, with bonus points for building one more powerful than a Cray X-MP.

Congratulations to all those who have been placed.

B. SNOBOL Fight

Competitors must write a garbage collection/memory management routine for use in interpreted languages like SNOBOL and Icon. Points are awarded for time, efficiency of storage allocation and garbage collection, and defeating the default system memory allocation limit.

C. Tightest Self-Modifying Code Loop

Self-explanatory. Competitors must write a program containing a self-modifying loop. Points are awarded for the speed of the loop, with bonus points for the elegance of the design and type of program the loop is in (e.g., a chess program).

D. Binary Patch

Each competitor is given an identical piece of non-working core image. Points are awarded for the expediency of the patch, with a bonus for doing it without proper access permissions.

E. "Home, James" Device Driver

Competitors must write a daemon to drive a peripheral device such as a laser printer. Points are awarded for time, speed of execution and complexity of use, with bonus points awarded for making it a part of the standard operating system.

F. User Loser

The hexadecathlon ends on a fun note, with competitors trying to remove the most ordinary users from a timesharing system in the least amount of time. Points are awarded for the user/time ratio, with bonus points for throwing the master user off, too.

The ACM is lobbying to have the hexadecathlon listed as a medal event in the 1988 Olympic Games in Seoul, South Korea. Training for Canada's team will begin immediately. Happy hacking!

dwarf,Dr J

Minimizing Math

When working on math problems, many people do more work than is necessary. There are many little "tricks" to save time, effort, and sometimes even reduce errors. Although the following suggestions might not be too helpful or might already be known, they will hopefully stimulate you to create your own shortcuts.

The mean of a given set of numbers can be calculated in this fashion: subtract a given number from the set, take the average of the resulting numbers and add onto it the

number that was subtracted. The result is the average. For example, given 1877, 1874, 1888, and 1882, subtract 1880 from all the numbers giving -3, -6, 8, and 2. The average of these is .25. Add on 1880 giving 1880.25 as the average of the original 4 numbers.

If one is working with formulae or series of formulae, then in most cases one should not plug in numbers until they have simplified the formulae algebraically. The reasons for this is that the formulae might simplify greatly with terms cancelling and common factors being pulled out. Also, there is less chance of an error since if you end up with a very messy algebraic expression, it probably means that an error was made, while with numbers being plugged in this might not be noticed. Finally, with the formula in a simplified form, if new data is to be used a lot of work will be saved as opposed to starting from scratch.

For example,

$$d = a^2 + ac + bc$$

$$e = \frac{d+ab}{a+b}$$

Plugging this in gives

$$e = \frac{a^2 + ac + bc + ab}{a+b}$$

$$= \frac{(a+b)(a+c)}{a+b}$$

$= a+c$ if $(a+b) \neq 0$ which is much simpler to work with.

Symmetry is a very powerful tool for reducing a mathie's work overload. One can recognise symmetry in graphs by noting that the graph on one side of a line is the same as on the other side (e.g. $f(-x) = f(x)$ or $f(-x) = -f(x)$). With this type of symmetry, one needs to only plot half of the function with the other half following from symmetry.

Symmetry also occurs when the variables can be switched with the function not changing, e.g. $f(x,y) = xy + 2(x+y)$, then $f(y,x) = yx + 2(y+x) = f(x,y)$. In this case any properties of x also hold for y . Thus, if one is trying to prove something about this function, once it is done for x , then all that is needed for y is "Similarly, ...".

There is also an easier way to multiply together certain numbers. Note that $(x-y)(x+y) = x^2 - y^2$. Thus, for example, $(398)(402) = (400-2)(400+2) = 159996$.

If you are dividing a number and are only interested in the remainder, then there are short cuts for some divisors. There are the obvious ones of 2 and 5. For 9, the remainder for the number is the same as the sum of the digits. For example, 203 has the remainder $2 + 3 = 5$. For 11, the remainder is the same as that of the alternating sum of the digits. For example, 2345 has the same remainder as $5 - 4 + 3 - 2 = 2$.

These are but a few examples of ways to reduce work. Using some imagination and common sense, anyone can come up with their own. After all, mathematics is tedious enough as it is with students doing just the minimum work required.

The Mad Irishman

CS AND MATH

In 1968 when I was an undergraduate, many students were advised to major in Library Science: there was a need in large libraries for a new generation of librarians conversant with the modern concepts of data storage and retrieval. Within a few years, the need for such librarians dried up; indeed many find it hard to believe that there ever was such a need.

Today the hot college major is Computer Science. At UCLA nearly one third of entering freshmen express an interest in majoring in Computer Science. The situation is very similar on campuses across the nation. Computer Science is very sexy stuff these days. Glossy magazines are filled with stories about computers and hackers. Kids as young as 14 are becoming wealthy by designing video games and software. What college freshman would not be seduced by this apparently glamorous profession? But there's the rub, *What profession?* Who has taken the time to identify this profession? What are the employment prospects for Computer Science majors? The glossy magazine articles deal primarily with startling advances in computer literacy; but little is said about employment for the *professionals trained in Computer Science*. When will these people become a glut on the market?

The answer is that the glut has already begun. Job recruiters for major corporations (including IBM!) have taken a second look at Computer Science majors and decided to look elsewhere. Where are they looking? They are looking at a group which for the last 5000 years has sat quietly in the background: mathematicians. Why now? Here is one answer from a job recruiter for a big corporation who wanted to hire people to do software verification: "I know that software verification sounds like Computer Science, but I don't want to hire a Computer Science major. I want someone who can *think*. I want a math major." In a less acerbic form, this assertion is being repeated by recruiters at Westinghouse, GE, Hughes Aircraft, TRW, and other high-tech companies across the country. How can this widespread dissatisfaction with Computer Science be reconciled with the fact that there is *indeed* a computer revolution going on around us?

The answer is very simple. Of course we should all be computer literate. Even the smallest modern business office keeps its accounts on a microcomputer. Soon the preferred way to pay bills and to conduct correspondence will be on a terminal. But one doesn't need to major in Computer Science in order to do these things any more than one needs to study at the Cordon Bleu in order to prepare one's supper. I occasionally need to do computing in my mathematical research, and (with no prior experience) I learned all the Fortran that I needed in just one afternoon. Is this because I am some kind of Computer genius? No—it's because I am a mathematician. The ability to write a program requires that one be familiar with certain patterns of logical thought, and that he memorize a few details of a particular programming language. The latter are trivial, while the former are the stock in trade of the mathematician.

Students have been concentrating on learning the mechanics of using the computer and not considering what is done with a computer in the real world. Of course there are extensive real world applications to information storage and retrieval, but employment in this field consists primarily of glorified stenographic jobs. There are also reasonable job opportunities in designing software (although I, for one, would not want my future riding on the software market). But what of the Computer Science major who wants to become part of the forefront modern scientific activity? What will he be doing with a computer? The answer is that he will be doing mathematical modelling, solving differential equations, doing advanced probability theory, and manipulating matrices: in short doing mathematics.

I have helped many of my students, all mathematics majors, to obtain employment. When prospective employers telephone me for recommendations, the litany is always the same: "We have lots of engineers and computer scientists our staff but we keep coming up against all sorts of mathematics problems. We need a math whiz." I always ask what computer background is required and the answer is generally "It would be nice if he could program in FORTRAN and APL, but if not we will train him. Math majors catch on quickly."

The last remark is the most telling. Math majors are not only trained in a number of advanced mathematical techniques, but they are also trained in abstract reasoning, modelling, and problem solving. These methods lend themselves well to many professions, from engineering to law to physics to medicine. And the *professions* are finally becoming aware of the value of the mathematical method.

Computer scientists, physicists, and engineers frequently do not realize that the technical problems with which they struggle on a daily basis are *mathematics*, pure and simple. I have often received phone calls from engineers which begin "We have a problem which we've been kicking around a long time and we thought that perhaps you could help us." Invariably the problem involves techniques which are well known to the working mathematician and I can answer the question instantly. People, *even technical professionals* do not appreciate the pervasive nature of mathematics. One is reminded of the gentleman who discovered at an advanced age that all his life he had been speaking prose. For a long time now, the technical world has been speaking mathematics, but is only beginning to become aware of it.

It is a sobering fact that the invention of the carburetion system on the Volvo, the creation of the fuel efficient wing on the Boeing 767, the use of prime numbers to create a new generation of "unbreakable" codes, the use of optical fibres in communications, the use of the fast Fourier transform in the resolution of photographs from space vehicles, and the use of satellites for communication purposes are all the products of work done by so-called pure, theoretical mathematicians. Why is the public unaware of the mathematician's important role in modern technology? Mathematicians are rarely encountered in the

popular press. While Jobs, Wozniak, and other computer celebrities frequent the covers of popular magazines, only one mathematician has ever been on the cover of TIME magazine, and that was because *he invented the stored program computer!*

Mathematicians do not make good PR copy; we do not give splashy interviews. The average research mathematician cannot explain his work in a few sentences. A biologist can toss off a few generalities about genetic engineering and an astronomer can refer glibly to cosmology and black holes, but if I tell you that I work on "biholomorphic mappings of pseudoconvex domains in multidimensional complex space," you are unenlightened and perhaps even annoyed.

Abtrusive we may appear, but industries and government are waking up to the fact that mathematicians are among the most gifted, creative, and able scientists around. Many students tell me "I want to major in math but my parents won't let me because they think that I won't get a job. They told me to major in Computer Science." Pity the parents *and* the students. The Department of Labor predicts that between now and the year 2000 there will be a need for about 150,000 Computer Scientists. Countrywide, there are over half that many students enrolled in university Computer Science programs *right now*.

Computer Science is in most instances not an end in itself: it is a *tool*. The world needs only a limited number of Computer Scientists. It is perfectly clear that we should all be able to use language effectively but there should be few linguists; likewise, we should all be computer literate but there should be few Computer Scientists. I sincerely believe that many high school counselors and teachers are ignorantly misguiding their students into taking up Computer Science. Computer Science is fun and it is glamorous. Mathematics is hard work and it is not glamorous. But mathematics is important and it is rewarding. Computer Science did not exist twenty-five years ago; will it exist twenty-five years from now? Mathematics has been around for 5000 years and will be here for another 5000. While new mathematical ideas are always developing, the old ones still remain valid and useful. In Computer Science, however, last week's news is this week's trivia. The subject is all hardware and few ideas. If corporate job recruiters prefer math majors to Computer Science majors, isn't it time for a reevaluation?

If you are thinking of becoming a Computer Science major, or advising someone to do so, then ask yourself this question: "What is Computer Science?" The answer is that no one knows. The successful professional of the future will have a couple of computer science courses under his belt; he will know how to use a machine. But he will be an expert in a subject with content, with a past, and with a future. How about mathematics?

Steven G. Krantz
Department of Mathematics
Pennsylvania State University
University Park
PA 16802

Fallacy of the Largest Prime

The following proof is often (e.g. Stats 231 notes) given for the nonexistence of the largest prime number. It is fallacious.

Assume that there is a largest prime p . Then take all primes less than or equal to p , call them p_1, \dots, p_n , and multiply them. Add one to this result. $p' = (p_1 p_2 \cdots p_n + 1)$ is not divisible by p_1 (it is congruent to 1 modulo p_1), or, similarly, by any other prime smaller than p . Therefore p' is prime. But p' is clearly greater than p (since p is a factor, and all factors exceed one). So there is a prime greater than p . Hence there is no largest prime.

Can you spot the fallacy? Answer next week.

Dr. Oscar

Feedbrahms

Dear Sirs:

I am writing to respond to Chef Al's letter of November 16, in which he attempted to show that all real numbers eat quiche. The "proof" involved a crass blunder in reasoning.

Chef Al had observed empirically that real numbers between 4 and 6 eat quiche, and by induction he showed that "all real numbers greater than or equal to zero" eat quiche. His argument that the ingredients of the quiche recipe can be multiplied by any common factor is legitimate, but let's remember that zero is a special case. Use the multiplying factor zero implied that each ingredient is present in zero quantity; therefore, the quiche eaten by zero in the null-quiche. Thus Chef Al's method hardly demonstrates that zero eats quiche! Indeed, I am at present constructing a proof that zero **does not** eat quiche. This becomes intuitively apparent from the following.

Chef Al's staggering *faux pas* was the extension of the argument to non-positive multiplying factors. He wrote "all negative numbers are simply -1 times a positive number," and claimed *ad arbitrium* that changing the sign has no effect on quiche consumption!

While reflecting on the fundamental absurdity of negative reals acting in the **same sense** as positive reals, I was led to this amazing and momentous insight: non-positive real numbers **vomit** quiche, yielding quantities proportional to their absolute magnitudes.

Like other mathematical breakthroughs in the past (such as the conception of non-positive numbers in the first place), this may be difficult to swallow (groan!), but it does follow logically and with sweet mathematical symmetry from what we know about the real numbers.

T.E. Rudy

Reply: Wine & Cheese

Dear Adam Hergert:

I believe that I am the appropriate person to respond to your comments on the Wine and Cheese which appeared in the last issue of *mathNEWS*. The Wine and Cheese of Summer '83, Winter '84 and Fall '84 were my responsibility. Consequently, I am thankful you have given me the opportunity to clarify a few points.

First, the question of "not enough wine" is not a problem that can be easily solved. There is a legal limit imposed on the amount of wine that can be ordered based on attendance. Since bar services is unlikely to create a new category called "expected-mathie-consumption-at-W&C's" this problem will continue. Although to my knowledge no W&C I have been involved with has ever gone dry it does look a bit sparse towards the end.

Secondly, I am slightly confused about your comments upon the quality of the wine since you offered only criticism, not suggestions. I would be the first to agree a few cases of less-than-great wine were included in the order. Basically this practice brings down the price and still satisfies those people who don't care what they drink. Although most of the wines selected retail for approximately \$8.50 per 1.5 litres, the bar bill for the last W&C exceeded \$2000.00. Therefore it is unlikely to see Rothschilds since the W&C is presently expected to lose over \$700.00. Generally the dry French wines like Mommessin, Blanc de Blanc, etc. are accepted as good quality in their price range. If you could offer some viable alternatives, I would be very interested in ordering them for the next Wine and Cheese.

Unfortunately it is not possible to please everyone when it comes to music. The D.J. for this W&C also worked the Winter of '84 and since I received only positive feedback on the quality of the music he was asked to return. But I shall let him address your comments upon his capabilities as a D.J. Personally I find it difficult to believe so many people were dancing to music that they didn't like. One of the dirtier jobs involved with the W&C is trying to get everyone to leave. Perhaps, Adam, if you had been assisting in this effort you would realize people were enjoying themselves AND the music. Also, I'm sorry you didn't like my personal request for Nina Hagen's New York, New York, perhaps you liked my other one - Frank Sinatra's New York, New York?

The most unfortunate thing is that a lot more people are willing to criticize than are willing to volunteer to help. Your suggestions and criticisms are *always* welcome. You help finance mathSOC and should therefore have an active role in it. Perhaps you'd like to volunteer to help next summer and see just how much time and work goes into putting on such an event. Just a few years ago, the W&C was 30 people in the Grad Lounge.

The door is always open to anyone who wants to comment, make suggestions or just talk. Adam, I hope that you come into the office and introduce yourself so that we

may discuss this further on a one-to-one level. My office hours are 10:30 - 11:30 on Tuesday and Thursday.

Later

Lisa Seabrooke

P.S. I'd like to repeat one line you stated; "the social event of the term, the party that everyone who's anyone is bound to go to!". Thanks, perhaps you'd like to help out with publicity?

MAPLE

Maple is a system for symbolic mathematical computation which has been under development at the University of Waterloo since December, 1980. The following is a sample of what Maple is able to do.

One form of arithmetic in Maple is exact rational arithmetic. For example, type in '1 + 1/4 + 1/16 + 1/64 + 1/256;' to get the answer $\frac{341}{256}$.

Maple has facilities for differentiation of expressions. For example, type in 'f := sin(x) * cos(x);' and 'fp := diff(f,x);' to get the result 'fp := cos(x)² - sin(x)²'.

There is a limit function to compute the limiting value of an expression as a specified variable approaches a specified value. For example, type in 'limit((tan(x) - x)/x**3, x=0);' to get the answer '1/3'.

An important facility in Maple is analytic integration. For example, type in 'f := (x-7)/(3*x+11)**2;' and 'int(f, x);' to get the answer $-1/3 \frac{x-7}{3x+11} + 1/9 \ln(3x+11)$. Type in 'int(exp(x**2),x);' and get the answer 'int(exp(x²),x)' since this function can not be integrated in closed form.

This very brief introduction leaves many important and useful features of Maple unmentioned. Examples are the ability to find the gcd, lcm, Taylor series and asymptotic series of polynomial functions, as well as being able to do definite and indefinite summations. Procedures can be written in Maple to solve very complicated problems. As can be seen, Maple can be a very valuable tool for reducing the tedium and errors in doing mathematics.

Igor

P.S. More information can be obtained in MC3064, the Maple Lab.

The above information was obtained from the booklet entitled:

"AN INTRODUCTION TO MAPLE:
SAMPLE INTERACTIVE SESSION

Bruce W. Char
Keith O. Geddes
Gaston H. Gonnet
University of Waterloo
Research Report CS-83-16
May, 1983"

FeedBack

This article is in response to the one entitled "Why Bother With CS?", which appeared in the November 16 issue of **mathNEWS**.

It is agreed that the department of Computer Science is suffering from a chronic shortage of computing resources. However, the authors have overlooked the fact that a personnel shortage exists: There simply aren't enough qualified people to go around. It is the combination of both factors which forced the department to restrict enrollment in the CS program.

The requirement for first year students to take CS 140 and CS 180 is not an unreasonable one. In the past, it has been the practice to assume that the freshmen arriving from high school have had little or no programming experience. In many cases, it is necessary for new students to unlearn bad programming habits taught to them in high school. The primary purpose of the first-year program is to introduce the student to the principles of structured programming and program design. The creation of advanced first-year CS courses for those students who have some mastery of the basics is desirable, although probably not feasible with the resource situation as it is now.

While many first and second-year students are talented programmers, few have any real conception of what Computer Science is. Coding is a *very* small aspect of computer science. Programmers are "a dime a dozen". If I had wanted to make my living as a programmer, I could have saved a great deal of time and money by taking my courses at a community college. Computer Science encompasses a much broader range of topics; for example: program design, analysis of algorithms and data structures, numerical analysis, formal language theory, the study of machine architectures, and so on. This is in addition to application-oriented topics such as real-time programming, graphics, database design, and artificial intelligence.

The claim was made that "creativity on the part of students is stifled by the rigidity of the CS program." This is an unreasonable complaint. There is little room for creativity in chemistry or medicine — why should *Computer Science* be any different?

The suggestion to have first or second-year students do Computer Science work for professors is patently ludicrous. These students just don't have the proper background to be of any use to a professor in his research. Besides, this role is already being fulfilled by graduate students. Even if it were possible, the bureaucracy required to monitor such a program would be prohibitively expensive in terms of personnel resources.

I agree that something should be done. The fundamental problem is money: There isn't enough of it. The department needs more money, not just for new equipment, but to afford better salaries for our professors. Too many competent people have been lured away by lucrative offers from companies seeking people for their research and development work. Government funding is tight, and can be expected to remain so. The money will have to come from somewhere else. If we want quality education, we the students should be prepared to shoulder more of the expense.

Lee Johnson 4A CS

Reply²

This article is in response to Lee Johnson's response to the **mathNEWS** article entitled "Why Bother With CS".

To the author's first point, regarding a shortage of personnel, I would like to observe that other universities, like UW, are gaining new faculty members. If UW needed to increase its faculty more rapidly, it would only need to make UW more attractive than other, competing universities. Granted, this requires money, but the shortage, from a local perspective, is not one of personnel.

The requirement that first year students take CS 140 and CS 180 is an unreasonable one. In these enlightened times, four out of five frosh seem to have some (albeit often limited) computer experience. A number of these have sufficient experience to not require such low-level courses as these. What was suggested by the authors of last issue's article was that some test be administered to those frosh wishing to not take these courses, and that if these students performed well enough, they should be allowed to work for profs on more advanced subjects to get their credits.

I quite agree that programming is not all there is to CS; some frosh have constructed their own computers, written large portions of the operating systems they are using, or worked for *real* software companies (where they can learn things that are never taught here in undergraduate courses). These frosh *are* more than programmers. They very likely know more about computers than most of our graduate students.

The claim was made that all "scientific" programmes *should* stifle the creativity of the student. I beg to differ. In upper year courses, solutions are marked for creativity and elegance. This gives the student ample room to develop his own solutions, and to use whatever methods he feels are best. In lower year courses, particularly in CS, solutions are marked on how closely they correspond to the "official" solutions.

In our CS 140 course, we had to write programs in a language named "pseudo-code" which had grammar rules stricter than the *Pascal* we were expected to hand-compile it into, but which coincidentally closely resembled BASIC. I would claim that this approach *does* unnecessarily cramp the individual tastes and sensibilities of the students.

The suggestion that first and second year students are not competent to work for professors is absurd. At least five frosh were involved in the WATERLOOP project, and at least three in the development of MAPLE. Some students clearly *do* have the proper background to be of use to professors.

The inadequacies in our CS programme are evident when it is observed that of the five students who were involved in the WATERLOOP project in their first years, two have since left UW, and a third has dropped out of CS. It should be noted that the two who left were both subsequently hired by what is probably *the* leading institute for computer research in North America, and that one of them has since resumed studies (not CS) at U of T. In other years, too, many of the best CS students have left the university or transferred into other disciplines.

I would also venture to suggest that the two most ma-

continued from page 9

continued on page 10

major problems with the CS programme in this university are underfunding and poor attitude. One of the students mentioned above became unhappy with the department when he complained about the courses which he was required to take and was told by a high-ranking faculty member (paraphrased) "We realise that you are considerably more advanced than other students in your year, but there is nothing that we can do for you. Wait for the other students to reach your level." This should *never* happen to a student in a *university*.

I would not like to imply that all, or even that most, of our CS profs take this position. Most of the profs that I have spoken to are aware that there are shortcomings in the programme structure, but have no good ideas on what can be done to escape them. One of the problems might be that this programme is really not oriented towards the future *Computer Scientist*. Rather it produces a sort of glorified programmer who has been exposed to the results attained by Computer Science, but who will nonetheless be no more than a Systems Analyst or a Computer Consultant. Certainly, some of our CS students fall into this class (and will therefore be out of a job within ten years after the programmers go, unless they make it to management), but those who really want to do research into the theories of computability or numerical languages are left without *their own programme*.

This raises a question about the programme. Should a university really have a programme which trains professionals? They do right now. Engineering, architecture, medicine, optometry, law, and other professions are normally taught in universities. There are compelling reasons for some of these. Other professions, such as computer programming, are taught at the vocational college level. What should distinguish the two classes of profession? Why could "computer consulting" not be taught at a community college? Perhaps it should be given a professional school all to itself, within the university: The UW "School of Computer Studies", or some such thing, and the Department of Computer Science could be left to those who are interested in the mathematical sciences relating to computers...

These are merely suggestions; we should not cry about how much money we need, but create the alternatives, and decide how to get the money to fund them.

Alfred Newman,
3A Co-op CS

more Feedback

Well, the Grey Cup is back in the West again. Hopefully, the 47-17 final score will convince the CFL that something has to be done about the playoff structure currently in place. The East and West are not equal, so why allow 3 teams from each division to qualify? To allow the East-West rivalry to continue why doesn't the CFL give playoff spots to the winner of each division, and then take the teams with the best records over the regular season. With a full interlocking schedule this seems to be the only fair way to allocate playoff berths. Who knows, it may even improve the quality of play in the Grey Cup Game.

Now, to reply to Adam Hergert's letter in the last issue of mathNEWS. I agree with your criticism of the wine, but you neglected to mention the inferior cheese and crackers (anyone remember the great munchies at the W+C in Spring '83?). I have been at a few events where we had the same D.J. as at the W+C. I too have requested and offered music, but each time I received the same reply: "I don't think these people will dance to it." The funny thing is, some of the music you singled out in your letter is the very stuff I requested. I believe that the music played at W+C events is meant to be eclectic, so that Math events are different from other events. I think this tradition could be maintained and improved by replacing AM top-40 stuff like Prince and Madonna with so 50's and 60's tunes (i.e. Buddy Holly, Beatles, Beach Boys, ...). As for your requests, maybe you could submit a request list to mathSOC before the next W+C so the D.J. can get some of the tunes you would like to hear.

Why bother with CS?

- 1) To avoid taking $n!$ Calculus and Algebra courses
- 2) To avoid taking Non-Major CS courses

I think the rationale for point 1 is obvious, but those who are CS majors may not understand the second. I plan to pursue a career in the Computer field if I ever get out of this institute of higher learning. However, I took a course which disagreed with me in first year, so I was not granted a CS major standing by the powers that be. As a result of this I have been taking the non-major CS courses. Each term we take the same concepts (sorting, recursion, etc.) and apply them to a different language. I had the misfortune of understanding these concepts the first time around so the whole ordeal is getting a little monotonous. My point is that there are people being kept out of the CS Major program who are as gifted as those in CS. I think the CS department should try to do something for those of us who are wasting away in the back row of CS 235 and 234 lectures.

Gord Erikson 2B

Dear Mr. Editor

In response to M.R. Daigle's question:

If it takes a chicken and a half a day and a half to lay an egg and a half, how long will it take a grasshopper with a wooden leg to kick the seeds out of a dill pickle.

Please put his mind at rest as I have the solution:

It takes the same length of time as it takes for a blind man to pick the fleas out of his ears with a pair of boxing gloves on.

Sincerely, I.M. AKeener

Maasthead: (Yes, you've found it!)
Golly gosh but i have lots of people to megathank. So let's get right to that shall we.
First, thanks contributors. Specifically: The lovely and talented Betsy McLeod, who shall remain nameless; Scott H, who got $\approx 90\%$ on PWS 256 midterm; Melvin Vockler's little brother; British Petroleum; Lisa Seabrooke; Steven G. Krantz; Lee Johnson; Gord Erikson; T.E. Rudy; Igor; Mr. AKeener; Dwarf; Doctors J.; Oscar, and Ernie; the MAD