



3. ad Tutoren

Ich finde es falsch und unfair, die Tutoren als Flaschen zu bezeichnen, da sich viele weit mehr als nur in der vorgeschriebenen Zeit für die Studenten einsetzen. Es kann schon vorkommen, daß ein Student mit einem

der Tutoren nicht zufrieden ist, man soll aber nicht gleich auf die Allgemeinheit schließen.

4. ad Kooperation

Wie einige Studenten der Basisgruppe wissen müßten, war ich immer

bereit bei auftretenden Problemen gemeinsam eine Lösung zu finden. Dies gilt auch heute noch, und ich bitte daher Herrn Krempler sich wenigstens einmal bei mir einzufinden und seine Probleme zu schildern.

Richard Messnarz

Der Tutor will mich richten

Zu meinem letzten Artikel über Tutoren gab es ungewöhnlich viele und heftige Reaktionen. Zwei negative, ein gutes Dutzend positive. Ich fühle mich zum Teil mißverstanden. Wollte ich aus alter Gewohnheit meinen alljährlichen Schlag gegen die Programmierübungen führen? Nein, diesmal nicht.

Die Programmiermethoden-Übungen sind zum Handkuß gekommen, weil die Situation den meisten bekannt ist und die Probleme am deutlichsten zu Tage treten. Die Programmierübungen sind auch besonders heikel, weil sie den ersten Kontakt der Studierenden zum Tutorensystem darstellen und daher für einige Semester die Einstellung zu Übungen und Tutoren prägen. Alternativen? Als ich zu studieren begonnen habe (was schon einige Jahre her ist) gab es noch keine Tutoren, die Situation war dementsprechend katastrophal. Katastrophaler als im letzten Jahr, sogar katastrophaler als im vorletzten Jahr. Kein Wunder, 2 Assistenten für 200-300 Studierende sind einfach zu wenig.

Abhilfe wurde gesucht.

Und auch gefunden. Zuerst 8 Tutoren, dann 14. Vieles wurde besser, insbesondere die Betreuung der Übungen vor Ort. Die Aufgabenstellungen wurden genauer und nur noch 4-6 mal im Semester geändert. Die Aufgabenstellungen wurden immer früher vor ihrer Bekanntgabe fertig. Eine Woche später, zwei Tage

später, gleichzeitig, eine Woche früher... Beispielprogramme wurden zur Verfügung gestellt. Viele dieser Verbesserungen gehen auf Initiative von Herrn Messnarz zurück.

Warum also Kritik?

Weil besser als katastrophal noch lange nicht gut bedeutet. 14 Tutoren sind besser als keine, aber erst Tutoren, die ihr Wissen auch weitergeben können, sind gut. Beispiellösungen sind besser als raten, was gefordert ist. Aber erst dann, wenn die Beispiellösung so gut ist, daß sie positiv bewertet würde, bringt es Nutzen. Es ist schön, wenn es Einsichtnahme gibt, aber erst dann, wenn alle Beteiligten wissen, worum es geht, werden die hohen Ziele erreicht.

Was fehlt mir im System?

Klarheit und Information. Wenn ich mir über die Bedeutung von vorläufigen Noten und zwangsläufiger Einsichtnahme im Klaren bin, kann ich entsprechend handeln. Nur so wird die Einsichtnahme zum Segen statt zum Glücksspiel.

Vorbereitung und Ausbildung. Es kann nicht genügen, zu sagen, daß keine Probleme mehr auftreten können, weil es ja 14 Tutoren und ein Konzept gibt. Sind die Tutoren, auf ihre Aufgabe, Wissen zu vermitteln vorbereitet? Dafür sind sie doch eigentlich angestellt worden, nicht? Oder ist es einfach nur Poker, ob ich an jemanden gerate, der Wissensvermittlung zufälligerweise im Blut hat?

Aufgabentrennung. Gebt den Tutoren, was der Tutoren ist, und den Assistenten, was der Assistenten ist. (Das könnte man noch einige Stufen so weiterführen, ungefähr bis Gott). Es sollte klar sein, was Tutoren tun sollen, und wenn sie es nicht können, gefeuert werden. Ebenso sollte klar sein, was sie nicht zu tun brauchen.

All das fehlt mir nicht nur bei der angesprochenen Übung. **Alle Übungen, die von Tutoren betreut werden, sind an den oben genannten Punkten gefährdet.** Manche mehr, manche weniger, nicht immer wird es so deutlich sichtbar wie bei den Programmiermethoden. Über allen schwebt die beruhigende Aussage "Wir haben etwas unternommen, wir haben Tutoren eingestellt, alles ist in Butter". Jeder Tutor ist ein potentielles Feigenblättchen, das die Sicht auf tiefergehende Probleme verstellt. Bei den Programmiermethoden-Übungen wurde heuer so ein tiefergehendes Problem angegangen. Die Übungsaufgaben wurden überschaubarer, somit die Aufgabenstellung leichter zu erstellen und zu erklären, das Programm leichter zu korrigieren. Weiter so.

Ich möchte nochmals betonen: Die Programmiermethoden liefern nur besonders griffige Beispiele, wie man es gut oder schlecht machen kann. Alle, die Übungen von Tutoren betreiben lassen, sind zum mitdenken eingeladen.

(ak)



Berufsinformation MathematikerINNEN

Am Abend des 18. Jänner 1993 fand erneut eine Veranstaltung zum Thema "Quo vadis studens mathematicae" statt. 3 Mathematiker referierten über ihre Eindrücke aus ihrer Arbeitswelt.

DI Kotnig, Geschäftsführer des Mikrozentriums und begeisterter Hobby-Segler, begann mit einer Situationsschilderung an der Technischen Uni/Graz Anfang der 70-iger. Auch damals gab es schon zu wenig Computer. Dafür war es damals schon möglich, in der Nacht zu arbeiten, wenn auch gezwungenerweise und nur von 23 bis 1 Uhr früh. (Es wäre übrigens toll, wenn das Computerzentrum zumindest am Abend offen hätte!) Fortgefahren ist er mit der Schilderung der eher traurigen Zukunftsaussichten kleiner Softwarefirmen. "Die Kunden erwarten sich um den Preis von MSDos die Qualität von MSDos." Daß ein kleiner Softwareanbieter sein Produkt teurer an den Mann bringen will, ist verständlich, wenn man bedenkt, wieviele Jahre man an einem Programm arbeitet, bis es dann (in nur kleinen Mengen) abgesetzt werden kann.

Viele dieser Firmen können der großen Konkurrenz nicht standhalten und müssen ihre Pforten schließen.

DI Simonic, zweiter Vortragender, ist am Institut für medizinische In-

formatik im LKH tätig, wo er vor allem für das Analysieren und Archivieren von Daten zuständig ist. (Eine Datenmenge von 3 Gigabyte pro Jahr muß für 20 Jahre aufbewahrt und verwaltet werden.) Er betonte vor allem, daß es sehr wichtig ist, im Team arbeiten zu können. Anscheinend ist es außerordentlich schwierig, ein Team zu bilden, in dem kein Mitwirkender die absolute Führungsrolle übernimmt.

Der dritte Vortragende war Dr. Matzl, der als Softwareentwickler arbeitet. Er meinte auf die Frage, was ein Student in die Arbeitswelt mitbringen soll, "abilities". Zusätzlich ist es von besonderer Wichtigkeit, sich verständlich machen zu können. Denn wenn man als Programmierer mit den verschiedensten Wissenschaften konfrontiert wird, sollte es keine großen Schwierigkeiten machen, mit Leuten, die nichts vom Computer verstehen, zusammenzuarbeiten.

DI Simonic ging in der Gesprächsrunde noch auf das Problem der hohen Ausfallrate bei InformatikerINNEN ein. Es gibt nämlich einen bedeutenden Unterschied zwischen dem Profi und einem Freak. Ein "in die EDV Vernarrter" erhöht zwar zumeist die Qualität der Software, hat aber oft zuviele Emotionen im Spiel. Das wirkt sich im Laufe der Zeit negativ auf seine Leistung aus.

Während ein Profi sich auch mal ein schönes Wochenende gönnt. (Er selbst hat noch nicht gelernt abzuschalten!) Apropos Wochenende. Es scheint üblich zu sein, Tag und Nacht und Wochenende vor dem Computer zu sitzen. Deshalb auch der Tip von DI Simonic, so schnell als möglich mit seinem Studium fertig zu werden, denn mit 25 ist man meist noch flexibler und mobiler, was Arbeitsort und -zeit angeht. Empfehlenswert ist auch ein Auslandsjahr in einem fremdsprachigen Land, das in jedem Fall den geistigen Horizont erweitert.

Der einzige Vorteil eines Techn. Mathematikers ist seine Fähigkeit, analytisch zu denken. Alles andere muß er sich wohl neben seinem Studium beibringen. So gesehen ist das Studium nur eine schwache Vorbereitung für die Berufswelt.

Nach den insgesamt beinahe 2 Stunden dauernden Vorträgen gab es "natürlich" auch noch die Möglichkeit mit den Referenten persönlich ins Gespräch zu kommen (bei ausgezeichneten Brötchen!).

Vielen Dank für den glatten Ablauf an die Organisatoren Hellfried Sabathy, Hannes Pintaritsch, Andi Terler und Michi Samastur.

(bd)

"Well, gentlemen, we've got a stunning new logo and a marvellous publicity campaign ready. We just need to come up with a product..."

