PEARL als erste Programmiersprache im Grundstudium der Studienrichtung Elektrotechnik (Informationsverarbeitung)

von Prof. K. Fasold, Bielefeld

Zusammenfassung

Im Studiengang Elektrotechnik, Studienrichtung Informationsverarbeitung, der Fachhochschule Bielefeld wurden im Grundstudium die Programmiersprachen FORTAN und PEARL eingesetzt. In einem analysierenden Bericht und anhand einiger der vorliegenden Erfahrungen wird gezeigt, daß für die genannte Studienrichtung die Programmiersprache einen leistungsfähigen Einstieg in die Programmierung technischer Prozesse bietet. Die Unterschiede in der Fähigkeit, komplexe Programme mittleren Umfangs auch nach kurzer Lernzeit konstruktiv zu beherrschen, sind beachtlich.

Schlüsselwörter: PEARL, Grundausbildung

1. Studienordnung Informationsverarbeitung

1.1. Fächer des Grund- und Hauptstudiums

Das Studium in der Studienrichtung Informationsverarbeitung gliedert sich gemäß der (auf der Grundlage der Prüfungsordnung vom 21.10.76 erlassenen) Studienordnung an der Fachhochschule Bielefeld in ein Grundstudium und ein Hauptstudium. Die Fächer des Grundstudiums sind klassische Fächer der Fachrichtung Elektrotechnik: Mathematik, Physik, Grundgebiete der Elektrotechnik, Meßtechnik, Grundlagen der ADV / Programmiersprachen, Bauelemente. Das Fachstudium ist dagegen sehr stark auf die Anwendung automatischer Datenverarbeitung ausgerichtet. Fächer des Hauptstudiums sind: technischer Aufbau von Datenverarbeitungsgeräten, Betriebssoftware von Da-

tenverarbeitungsgeräten, Datennetze und Datenfernübertragung, Steuerungs- und Regelungstechnik, Betriebswirtschaft und Operations Research u.a. Dazu kommen Wahlpflichtfächer, die aus einem Katalog von 20 Datenverarbeitungsanwendungsfächern wie Geometrische Datenverarbeitung, Dokumentation und Datenbanken u.a. auszuwählen sind.

1.2. Formen der Lehrveranstaltungen

Die Fachhochschule kennt im Lande Nordrhein-Westfalen drei Grundformen der Lehrveranstaltungen in den technischen Fachbereichen, nämlich Vorlesung, Seminar und Praktikum, wozu in unserer Studienordnung etwa folgendes ausgeführt ist:

Die **V o r l e s u n g** dient der zusammenhängenden Darstellung eines Lehrstoffes sowie der Vermittlung von Fakten und Methoden durch den Lehrenden. Der Lehrende trägt vor und beantwortet Fragen. Die Studenten verhalten sich überwiegend rezeptiv und stellen Informationsfragen.

Das **S e m i n a r** dient der Erarbeitung von Fakten, Erkenntnissen, komplexen Problemstellungen im Wechsel von Vortrag und Diskussion. Der Lehrende leitet die Veranstaltung und führt die Diskussion. Die Studenten erarbeiten Beiträge und diskutieren die Beiträge*.

* Die Wirklichkeit sieht allerdings häufig so aus, daß die Professoren Übungen veranstalten, in denen sie Aufgabenlösungen vortragen, die eine Anwendung des Vorlesungsstoffes auf die Praxis darstellen.

Tabelle 1. Lehrveranstaltungsform und -umfang einiger Fächer der Studienrichtung Informationsverarbeitung an der Fachhochschule Bielefeld

Bezeichnung des Faches	1. Sem.			2. Sem.			3. Sem.			4. Sem.			5. Sem.			6. Sem.			
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	
Grundlagen der ADV / Programmiersprachen				2	-	2	2	2	2										
Prozeßdatenverarbeitung									2	1	1		1	1	2				
Prozeßmeßtechnik									1	1	-		1	-	2				
Betriebswirtschaft und Operations Research																2	2	-	

Das Praktikum dient dem Erwerb und der Vertiefung von Kenntnissen durch Bearbeitung praktischer, experimenteller Aufgaben. Der Lehrende leitet die Studenten an und überwacht die Veranstaltung. Die Studenten führen praktische Arbeiten und Versuche durch.

Durch einen Studienverlaufsplan werden den einzelnen Fächern und Studienhalbjahren Lehrveranstaltungen der Form Vorlesung, Seminar und/oder Praktikum zugeschrieben. Eine inhaltliche Festlegung der Lehrveranstaltungen ist durch "Stoffpläne" erfolgt, die jedoch glücklicherweise unverbindlich sind. Einige der "Studentenafeln" sind in Tabelle 1 gegeben.

2. Didaktik des Faches Grundlagen der ADV / Programmiersprachen

2.1. Eingangsniveau

Die Studenten des Fachhochschulstudienganges Informationsverarbeitung an der Fachhochschule Bielefeld sind nahezu ausschließlich Absolventen der Fachoberschule Technik - Fachrichtung Elektrotechnik - und nahezu ausschließlich männlich. Sie haben beim Eintritt in die Fachhochschule in der Mehrzahl noch keine Datenverarbeitungsausbildung gehabt. Allerdings hat ein merklicher Teil der Studenten an einem Hobby-computer Programmiererfahrung, häufig in BASIC, erworben. Dieser Anteil liegt im laufenden Studienjahr bei etwa 12,5% und scheint gegenüber den vergangenen Jahren zugenommen zu haben.

Insgesamt scheinen bei Beginn der Lehrveranstaltung das Interesse für praktische Datenverarbeitung und die Fähigkeit zur konkreten technischen Problemlösung sehr ausgeprägt zu sein, während die Fähigkeit zu Abstraktion und Systematisierung und Lösen komplexer Probleme sich noch entwickeln muß.

Bezüglich ihrer speziellen Vorkenntnisse, insbesondere ihrer Programmiererfahrung, sind die Studenten außerordentlich heterogen.

2.2. Lernsituation

Das propädeutische Fach Grundlagen der ADV / Programmiersprachen ist das einzige Datenverarbeitungsfach des Grundstudiums und zeitlich mit 11,5% an dessen Lehrangebot beteiligt. Es wird regelrecht erdrückt von den "großen" Fächern Mathematik, Physik und Elektrotechnik. Der Lehrende hat daher in seiner Veranstaltung zu berücksichtigen, daß die Teilnehmer in ihrer Mehrheit überhaupt noch nicht in der Lage sind, sich diesem Fach, das sie als Nebenfach empfinden, zuzuwenden.

Das Fach Grundlagen der ADV / Programmiersprachen wird an der Fachhochschule Bielefeld zweisemestrig dem 2. und 3. Fachsemester angeboten. Eine Besonderheit des Bielefelder Studienverlaufsplanes besteht darin, daß das erste Halbjahr ausschließlich mit Vorlesung und Praktikum bestritten wird. Es ist in diesem Kreis wohl nicht erforderlich zu begründen, daß die Veranstaltungsform Vorlesung ebensowenig geeignet ist, Grundfertigkeiten im Programmie-

ren zu vermitteln, wie diese Veranstaltungsform die Beherrschung des kleinen 1 x 1 vermitteln könnte. In Bielefeld ist es also der Veranstaltungsform Praktikum überlassen, die Grundfertigkeiten in der Konstruktion von Programmen zu entwickeln.

Die hier geschilderte Lernsituation an der Fachhochschule Bielefeld verlangt in ganz besonderem Maß eine Anfangsprogrammier-sprache und ein Programmiersystem, die den Lernenden mit geringem Zeitaufwand und mit wenig Anleitung zur sicheren Beherrschung einfacher Programmier-techniken verhelfen. Damit sind auch für das Anfängerselbststudium geeignete Unterlagen gemeint - wo zweifellos noch immer die größte Schwäche von PEARL liegt.

2.3. Lernziele

Das Fach Grundlagen der ADV / Programmiersprachen ist Grundlage nahezu aller Fächer des Hauptstudiums. Daher müßte es Hardware und Software behandeln. Es hat sich jedoch bewährt, den Inhalt des Faches auf die Softwaregrundlagen zu beschränken. Die Lernziele liegen dann allgemein gesprochen in Dingen wie der Entwicklung eines guten Programmierstils, im Erlernen einiger grundlegender Algorithmen, in Fertigkeiten in der Handhabung des Programmiersystems, in der Fähigkeit zu ingenieurmäßiger Konstruktion von Programmen, und vor allem in der Entwicklung der Fähigkeit, "datenverarbeitungsmäßig zu denken".

Aber auch dies zum Stichwort Lernziele: Ein außerordentlich wichtiges Lehrziel besteht darin, von Anfang an Augenmaß für die Komplexität einer Automatisierungsaufgabe zu wecken und Software als technischen Gegenstand zu begreifen, der nach professionellen Gesichtspunkten zu konstruieren ist.

3. Instrumentelle Möglichkeiten

3.1. Datenverarbeitung im Fachbereich Elektrotechnik der Fachhochschule Bielefeld

Der Fachbereich Elektrotechnik der Fachhochschule Bielefeld ist aus der Staatlichen Ingenieurschule für Maschinenwesen Bielefeld hervorgegangen. Bereits im Jahre 1965 wurde dort - damals in einer Ingenieurschule ganz ungewöhnlich - eine Datenverarbeitungsanlage Zuse Z 25 mit 6 Programmierplätzen und mit on-line betrieblichem Zeichentisch Zuse Z 64 in Betrieb genommen. Es folgte dann nach jeweils vier Jahren etwa ein neues System: Siemens 302 mit Lochstreifenumgebung, Siemens 305 mit Platte und Prozeßelement, KAE Doppelrechnersystem. Parallel dazu erschlossen wir dem Fachbereich die Rechenkapazität des Großrechners in der Universität, wobei es mir allerdings noch immer nicht gelungen ist, eine RJE-Station ins Haus zu bekommen. Die Zahl der Datenverarbeitungs-kollegen wuchs seit 1972 um fünf Professoren und drei Ingenieure. Daneben erhielten einige Laboratorien Kleinrechner für begrenzte Automatisierungsaufgaben und Mikrorechnerprogrammierung. Insgesamt steht heute dem Fachbereich Elektrotechnik eine ganz beachtliche Ausbildungs- und Forschungskapazität auf dem Gebiet Datenverarbeitung zur Verfügung.

3.2. Erfahrungen mit FORTRAN IV / Siemens 305

Es war von Beginn an unser Ziel, im Fachbereich selbst nur eine kleinere Datenverarbeitungsanlage für die Laborautomatisierung zu installieren und daneben für die "große" Datenverarbeitung der übrigen Anwender das Hochschulrechenzentrum der Universität verfügbar zu machen. Dies war keineswegs eine vorgeschobene Absicht, um einen unabhängigen Universalrechner örtlich verfügbar zu machen, sondern entsprach unseren Vorstellungen von den Schwerpunkten der Lehre in einer Studienrichtung Informationsverarbeitung. Im Ergebnis des demokratischen Beschaffungsprozesses für

unseren ersten echten Prozeßrechner war es dann allerdings so, daß wir einen relativ leistungsfähigen Universalrechner mit sehr magerer Prozeßperipherie erwarben.

Dem Stand der damaligen Softwaretechnologie entsprechend habe ich vorzugsweise in der Assemblersprache PROSA 300 programmieren lassen. Das Betriebssystem war für unsere einfachen, im Grundstudium eingesetzten Prozeßexperimente (Ampelsteuerung, Heizungsmischregelung, Impulzzähler, einfache Kennlinien u.s.w.) erfreulich schlicht, und in dieser Periode entstanden einige sehr gute Arbeiten, zumal der Programmieraufwand in PROSA durch die Vereinbarung von Makrobefehlen recht gering gehalten werden konnte.

Als unsere Prozeßrechenstelle die Bausteine der Prozeßprogrammierung auf FORTRAN-Bibliotheksunterprogramme übernahm, ließ ich von da an vorzugsweise in FORTRAN programmieren, konnte mich aber mit den Ergebnissen nicht in gleicher Weise wie mit den PROSA-Programmen zufrieden geben. Die den Studenten erteilte Auflage, FORTRAN-Programme in der üblichen Weise "wohlstrukturiert" zu gestalten, brachte nur mäßigen Erfolg. Regelmäßig sind Lösungen vorgelegt worden, die erst in FORTRAN aufgeschrieben und dann mit einem Struktogramm beschrieben wurden, das dann um eine geeignete Symbolik zur Darstellung von Sprüngen erweitert worden war.

Zu unserem weiteren Unglück installierte uns Siemens einen sehr komfortablen FORTRAN-IV-Compiler, und einige der Benutzer aus unserem Fachbereich beglückten uns damit, daß sie Programmpakete wie CSMP und ECAP auf unsere Anlage brachten, statt mit ihren Projekten den Unirechner zu benutzen. Das brachte uns eine gewaltige Überlastung der Anlage, und ich kann mich erinnern, daß ich einige Male den Rechnerraum mit körperlicher Gewalt geräumt habe, um meinen Studenten zu ermöglichen, ihre Laboraufgaben mit den einfachen Prozeßexperimenten zu fahren.

Dies veranlaßte offensichtlich den Fachbereich, einen größeren Prozeßrechner zu beantragen, der in Parallelarbeit als Vielzweckgerät genutzt werden sollte: Laborautomatisierung, Programmentwicklung mit time sharing und Stapelverarbeitung. Eigentlich sollte das Gerät auch noch RJE-Station zum Unirechner werden. Jedoch ist dieser Plan auf unbestimmte Zeit zurückgestellt. Im übrigen aber haben wir das System nach einem Beschaffungsprozeß voller Dramatik schließlich zum größten Teil bekommen.

3.3. Möglichkeiten mit PEARL

Mit dem neuen System, das zunächst neben der Siemens 305 lief und nach und nach eine gewisse Zuverlässigkeit erreicht hat, stand uns nun mit PEARL zum ersten Mal eine blockorientierte Sprache im Hause zur Verfügung. Da die Fächer Prozeßdatenverarbeitung und Betriebssysteme und weitere Fächer des Hauptstudiums Echtzeitverarbeitung beinhalten, liegt es nahe, bereits im Grundstudium die Elemente der Programmierung nebenläufiger Prozesse zu lehren. Damit ergibt sich eigentlich zwangsläufig die Entscheidung, allen Bedenken zum Trotz PEARL als erste Programmiersprache im Grundstudium der Informationsverarbeitung einzuführen. Der Kurs Sommersemester 80 - Wintersemester 80/81 lief erstmals in dieser Form.

4. Erreichbarkeit der Lernziele im Vergleich FORTRAN / PEARL

4.1. Ausgangslage

Der im Sommersemester 1979 begonnene Jahreskurs im Fach Grundlagen der ADV / Programmiersprachen baute noch voll auf der Siemens 305 und auf FORTRAN auf. Lediglich ein Praktikumstermin wurde am KAE-System, das zu diesem Zeitpunkt noch sehr instabil und auch unzulänglich dokumentiert war, abgehalten. Zum Prüfungsumfang gehörten die

Programmiersprachen FORTRAN und PROSA. Bei der Prüfung im Frühjahr 1980 lag die Bestandenquote bei 50 %.

Der im Sommersemester 1980 begonnene Jahreskurs wurde voll auf die KAE-Rechner abgestellt. Als Programmiersprachen standen FORTRAN 77 und PEARL zur Verfügung. Ich entschied mich für den Einsatz von PEARL. Eine weitere Programmiersprache wurde mit Rücksicht auf die außerordentlich schwierige Praktikumsituation nicht hinzugenommen: Im Sommer 1980 konnte keine Praktikumsitzung ordnungsgemäß beendet werden, da das System noch zu häufigen Ausfällen neigte. Ein besonders gravierender Zwischenfall muß hier noch berichtet werden: In der Sommerpause 1980 hatte ich auf der Grundlage der mit von der Prozeßrechenstelle der Fachhochschule Bielefeld dafür zur Verfügung gestellten KAE-PEARL-Dokumentation und unter Verwendung der deutschen Begriffe nach Werum und Windauer [1] eine Syntaxdokumentation in Form von Syntaxdiagrammen erarbeitet und war dabei, ein kleines Lehrbuch unter Verwendung der Begriffe und der frisch gezeichneten Diagramme zusammenzustellen, als ich erfahren mußte, daß das zugrunde gelegte KAE-Dokument völlig veraltet war, so daß meine Arbeit in den Papierkorb wandern mußte. Leider hat sich die Situation bis heute nicht gebessert. KAE hat eine Anfrage wegen Informationen bis heute nicht beantwortet. Der PEARL-Verein hat noch keine Liste deutscher Begriffe herausgegeben. Es bleibt festzuhalten, daß ich seitdem die PEARL-Kurse mit äußerst dürftigen Arbeitsunterlagen durchführe. Es spricht für die Eignung von PEARL als Anfangsprogrammiersprache, daß bei der Prüfung im Frühjahr dieses Jahres, die auch die Echtzeitelemente der Sprache umfaßte, die Bestandenquote 98 % betrug, obwohl ich die Leistungsanforderungen keineswegs herabgesetzt hatte.

4.2. Strukturiertes Entwurf

Ebenso wie in den Kursen unter Verwendung von FORTRAN verlange ich in den PEARL-

Kursen von den Teilnehmern wohlstrukturierte Entwürfe. Allerdings verzichte ich bei PEARL auf die Vorlage von Struktogrammen und verlange nur eine durch Einrücken sorgfältig hierarchisch gegliederte Niederschrift des Programms. Anders als bei FORTRAN haben wir bei PEARL vollen Erfolg. Bild 1 zeigt dagegen ein FORTRAN-Beispiel, wie es heute auf meinen Tisch kam.

4.3. Beschreiben von Algorithmen

In den FORTRAN-orientierten Kursen habe ich zur Darstellung von Algorithmen gewöhnlich eine Pascal-Fassung entwickelt, die dann nach gewissen formalen Regeln auf FORTRAN umgeschrieben wurde. Selbstverständlich können wir in einer PEARL-Umgebung den Umweg über Pascal entbehren.

Es braucht in diesem Kreise wohl nicht unterstrichen zu werden, daß PEARL ganz vorzüglich zur Beschreibung von Algorithmen geeignet ist. Fraglich ist eigentlich nur, ob wir durch Einsatz von PEARL eine Hemmschwelle aufbauen, die unsere Studenten hindert, mit Pascal oder Algol formulierte Algorithmen zu übernehmen.

Diese Frage dürfte sich bereits in den nächsten Monaten beantworten, wenn wir als weitere Programmiersprachen FORTRAN, Algol und Pascal in den Kurs Grundlagen der ADV / Programmiersprachen aufnehmen werden.

4.4. Lesbarkeit der Programme

Das schon vorgestellte Bild 1 stellt einen Ausschnitt aus einer Arbeit eines Studenten im 6. Fachsemester dar, der eine ganz beachtliche Leistung vollbracht und sich selbstständig die Benutzung einiger Programmpakete des Unirechners erarbeitet hat. Allerdings hatte er bereits BASIC-Vorkenntnisse, als er zu uns kam.

```

      INTEGER H,SB,DB,HZ(60),SBD(60),DBD(60),HU,HO,SBO,SBU,DBO,DBU
      INTEGER DTH,DTB
      COMMON NH,HU,HO,DTH,HZ(60),NB,DEU,DBO,DTB,DBD(60),SBU,SBO,SBD(60)
      READ(10,20)NH,HU,HO,DTH,NB,DEU,DBO,SBU,SBO,DTB
20    FORMAT(1X,10I3)
      K=0
11    K=K+1
      READ(10,21,END=30,ERR=39)H,DB,SB
21    FORMAT(1X,3I3)
      HZ(K)=H
      SBD(K)=SB
      DBD(K)=DB
      GOTO11
39    K=K-1
      BACKSPACE 10
40    K=K+1
41    READ(10,50,END=30,ERR=60)H
50    FORMAT(1X,I3)
      HZ(K)=H
      GOTO40
60    READ(10,70,END=30)DB,SB
70    FORMAT(4X,2I3)
      SBD(K)=SB
      DBD(K)=DB
      K=K+1
      GOTO60
30    M=K-1
      CALL ZEITK(TH,TD,TS)
      STOP
      END

```

Bild 1. Teil eines studentischen FORTRAN-Programms

Es handelt sich eigentlich um ein ganz schlichtes Eingabeprogramm. Sie werden kaum länger als eine Minute brauchen, um es in seinem prinzipiellen Ablauf zu enträtseln. Dennoch - ich bin froh, daß mir solche trickreichen Programme in Zukunft seltener vorgelegt werden dürften.

4.5. Fehlerhäufigkeit

Zur Frage der Fehlerhäufigkeit können keine qualifizierten Angaben gemacht werden. Im allgemeinen dürfte aber die Lösung von Praktikumsaufgaben mit PEARL besser gelingen. Jedenfalls scheint es bei PEARL sehr viel

seltener vorzukommen, daß ein Student wegen eines von ihm eingebauten Programmfehlers einen Assistenten zu Hilfe ruft.

4.6. Konstruieren oder Basteln

Zu dem Thema Konstrukteur u. Bastler sind schon ganze Bücher geschrieben worden. Hier wäre nur zu berichten, daß es uns nicht gut gelingt, die routinierten BASIC-Programmierer zur gegliederten und leserorientierten Gestaltung ihrer Entwürfe und zur Zerlegung in für sich verifizierbare Programmteile zu überreden.

Als außerordentlich erfreulich kann ich berichten, daß es mir mit PEARL anders als mit FORTRAN mühelos gelungen ist, eine komplexe Aufgabe mit verketteten Datenstrukturen und mehrfachen Rekursionen im Praktikum mit zum Teil ganz beachtlichen Ergebnissen bearbeiten zu lassen.

4.7. Weiterführende Fächer

Zum Schluß möchte ich noch erwähnen, daß die Echtzeitsprachelemente in PEARL sehr gut bereits in der Grundlagenveranstaltung eingeführt werden können. Das bringt ohne Zweifel eine ganz außerordentliche Vereinfachung für die weiterführenden Fächer Prozeßdatenverarbeitung und Betriebssysteme.

Am anderen Ende des Spektrums der softwareorientierten Fächer liegt das Fach Betriebswirtschaft und Operations Research. In ihm werden Programmpakete des Unirechenzentrums benutzt. Dabei spielt die Anfangsprogrammiersprache keine große Rolle.

5. Schlußbemerkung

Nach dem bisherigen Erfahrungsstand werden wir wohl bei PEARL als erster Programmiersprache für die Studienrichtung Informationsverarbeitung bleiben. Ein neuer, über Basis-PEARL hinausgehender Compiler ist angekündigt. Mit ihm sollten alle wesentlichen Sprachmittel einer modernen, universellen Programmiersprache verfügbar werden.

Literatur

- [1] Werum, W.; Windauer, H.: Process and experiment automation realtime language: PEARL; Braunschweig 1978

Anschrift des Autors:

Prof. Klaus Fasold
Wildhagen 48, 4800 Bielefeld 1
Tel. (0521) 103579