

- häufig mit großer Anstrengung - gelernt werden. Das Begreifen von Ornamenten zum Beispiel wird erkannt und auch beschrieben werden können, wenn die Beurteilung fundiert sein will. Das Durchdenken praktischer Erfahrung ist es, was unseren Unterricht von der Beschäftigungswelt abhebt. Erst die Theorischulung führt zu kritischem Bewusstsein und rationalem Urteil. Kein Aspekt des Faches ist deshalb ohne eine grundlegende Theoriedebatte denkbar.

Die Vorliebe vieler Schüler für die Textgestaltung, gleich ob Mädchen oder Jungen, liegt wohl in den Erfolgserlebnissen und der daraus erwachsenden Selbstsicherheit, die ihnen der Unterricht vermitteln kann. Die Entfaltung und Förderung der Individualität eines jeden Schülers steht in der Textgestaltung neben der Erziehung zu kritischem Bewusstsein und Urteilsvermögen.

R. Skudleky

INFORMATIK

1) Organisation des Versuchs

Mit Beginn des Schuljahres 1975/76 nimmt das Zeppelin-Gymnasium und durch die Kooperation in der reformierten Oberstufe auch das Geschwister-Scholl-Gymnasium am Schulversuch „Informatik“ in der gymnasialen Oberstufe in Nordrhein-Westfalen teil. Dieser Versuch gilt als Pilotprojekt des Bundesministeriums für Forschung und Technologie und wird von ihm über das Land NRW mitfinanziert. Es nehmen weitere 6 Gymnasien aus Bochum (4), Opladen (1), Kerpen-Türnich (1), die Kaufm. Berufsschulen (gymnasialer Zweig) Ahaus und die Gesamtschule Gelsenkirchen als Curriculum-Zentrale teil. Die wissenschaftliche Betreuung liegt wie bei allen Projekten „DV im Bildungswesen“ beim FEoLL (Forschungs- und Entwicklungszentrum für objektivierte Lehr- und Lernverfahren GmbH) in Paderborn. Das Projekt „INIS“ (Informatik in der Schule), wie es im internen Gebrauch heißt, wird dabei von der Zentrumsprojektgruppe unter Prof. Dr. M. Lansky von Dr. Sturm und seinen wissenschaftlichen Mitarbeitern betreut.

2) Ziel des Schulversuchs

Der Versuch soll bis zum Ende des Schuljahres 1977/78 untersuchen, ob „Informatik“ als Schulfach des Gymnasiums eingeführt werden soll. Dazu werden ein Curriculum entwickelt und die dabei entwickelten Kurse im Unterricht erprobt. Sollte das Ergebnis dieses Versuches negativ sein, bleibt es bei der bisher bereits an vielen Gymnasien geübten Praxis: Informatik wird als Teilgebiet in die Fächer Mathematik, Physik, Technik (neues Schulfach) oder Wirtschaftswissenschaft (Organisationslehre) integriert. Sollte das Schlußgutachten das Schulfach „Informatik“ zur Aufnahme in den Fächerkanon des Gymnasiums empfehlen, so gilt es zu prüfen, mit welcher Wertigkeit es eingeht: ob als Gk-Fach (Wahl) ohne Abiturberechtigung oder wie jetzt im Schulversuch als GK-Fach, das auch als 3. oder 4. Abiturfach in Frage kommt, oder schließlich als Leistungskursfach jeweils im Aufgabenfeld Naturwissenschaften.

3) Bisherige Aktivitäten

Bisher liefen in NRW nur an der Gesamtschule Gelsenkirchen (Lk und Gk) und am Lucas-Gymnasium Opladen (Gk) Versuche im Fach Informatik. Weitere Schulversuche mit Tischrechnern im CUFU (Computer-unterstützter Fachunterricht) wurden vom Kultusministerium vorher an 10 Schulen finanziell unterstützt. Unser Gymnasium konnte bei der Antragstellung zu diesem Schulversuch am 21.12.73 eine Vielzahl von Aktivitäten im Rahmen des CUFU nachweisen: Einsatz des Tischrechners P 101 (Olivetti) im Mathematik- und Physikunterricht der Oberstufe, Kurse Informatik im Differenzierungsbereich der Mittelstufe in den Fächern Mathematik und Physik, Wahlkurse Informatik (Gk) im Fach Mathematik und einige Jahre vorher die vom verstorbenen Kollegen Dr. Markwald initiierten AG's über „Kybernetik“ und „Sequentielle Logik“. Diese AG's bereiteten mit einfachen selbstgebaute Modellrechnern und Schaltelementen diese Entwicklung vor (s. auch Jahresberichte 66/67 und 71/72).

4) Technische Voraussetzungen

Durch die Bereitstellung von finanziellen Mitteln im Haushaltsjahr 1974 durch den Rat der Stadt Lüdenscheid wurde die Grundlage für eine Teilnahme an dem Projekt gelegt. Nach schwierigen Verhandlungen mit den beteiligten Stellen (SK Münster, KM Düsseldorf und Stadt Lüdenscheid) konnte im November der Auswahl einer Maschine der mittleren Datentechnik (Minicomputer) nähergetreten werden. Das Kultusministerium, Abt IV Referat 3 (Medien), schränkte die Auswahl auf 3 ungefähr gleich-

Figur 1

Interdata 7/16

(16 bit - Wort)

X - Nr.
'01'

(X:Hexadecimal)

LU Nr.

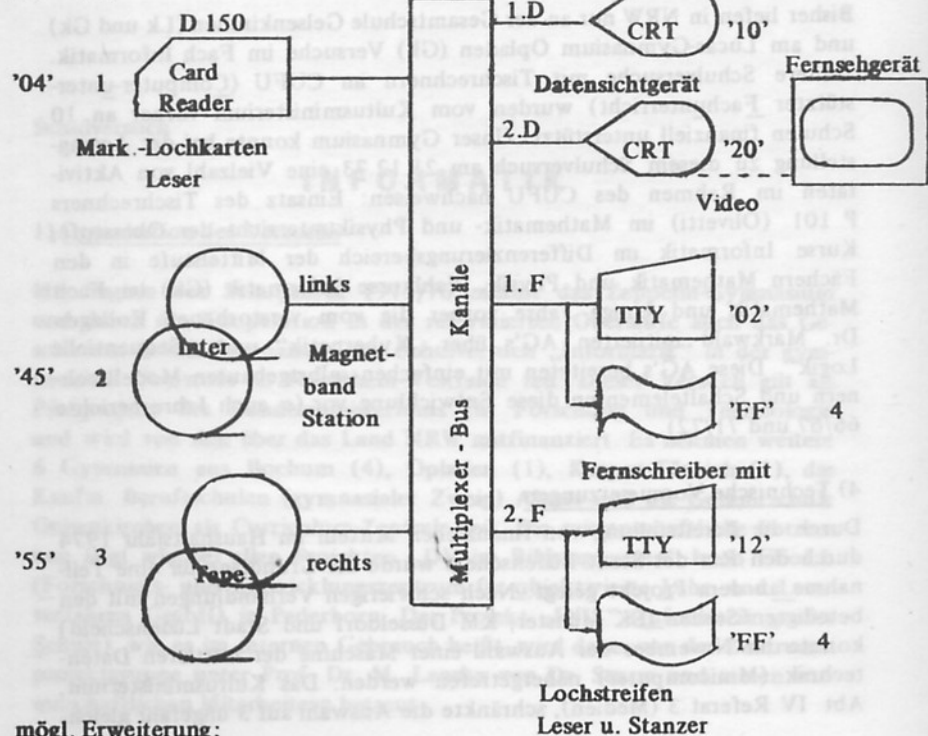
(Logical Unit)

PU Nr. 'Hexad.'

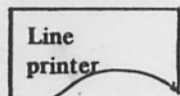
(Physical Unit)

PU LU

PU LU



mögl. Erweiterung:



Schnelldrucker
10



Platte

Disk.

$$'45' = 45_{16}$$

$$= 4 \cdot 16^1 + 5 \cdot 16^0$$

wertige Typen der Firmen Interdata, Digital-Equipment und Data General ein. Nach einer Checkliste des FEoLL über Eigenschaften und Preise von Hardware (CPU) und Peripherie wurde die Auswahl getroffen. Sie fiel auf den Prozeßrechner INTERDATA 7/16 in der Konfiguration, wie sie Fig. 1 zeigt: Zentraleinheit (CPU) mit der Peripherie von 4 Terminals (2 Fernschreiber (TTY) und 2 Datensichtgeräte (CRT), 1 Magnet-Cassettenstation (Intertape) mit 2 Einheiten und einem Kartenleser für IBM-Markierungs- oder Lochkarten). Beide Fernschreiber sind mit Lochstreifenleser und -stanzer ausgerüstet. Zur Gesamtausstattung gehörte dann noch der Modellrechner „Simulog“. Nur 3 der beteiligten Gymnasien wurden so ausgestattet, während bei den anderen auf vorhandene Rechner zurückgegriffen wurde bzw. bei den Gymnasien in Bochum u. a. eine Terminal-Lösung mit einem IBM-Großrechner gewählt wurde. Die vollständige Lieferung erfolgte im Februar 1975, und es dauerte bis zum Juni 1975, bis die ersten Programme in BASIC eingegeben werden konnten. Bis dahin und auch später zeigte sich in vielen Fehlern und Störungen bei Hard- und Software, daß zwischen einem praktisch wartungsfreien Tischrechner und dieser Rechnerkonfiguration ein großer Unterschied besteht. Dadurch entsteht eine erhebliche zeitliche Belastung der beteiligten Kollegen.

Das System 7/16 ist außerordentlich flexibel. Man kann auf mehreren Sprachebenen programmieren (ROM, 16-bit, ASSEMBLER, problemorientiert (Basic, Fortran)). Weiter ist der interne Speicher (32 kB) erweiterbar. Ebenso kann auch die Peripherie mit weiteren Terminals, Lineprinter, Magnetband- oder Disk-Speicher, Schneller Lochstreifenleser und -stanzer erweitert werden, theoretisch auf 256 Einheiten. Unser Rechner arbeitet im Multi-User-Betrieb mit einem BASIC-Interpreter, d. h. von jedem der 4 Terminals aus kann man in der Sprache BASIC programmieren und die Peripherie (Kartenleser (CR) und die Magnet-Cassettenstation) anwählen und bedienen. Die Verbindung zu den Lochstreifen-Leser und -Stanzer ist nur vom jeweiligen Fernschreiber möglich. - Ein einfaches Einschalter des Systems ist zunächst nicht möglich, man muß dazu das Betriebssystem über einen 16-bit Loader, dann über einen General-Loader (Assembler) in das Memory (int. Speicher) einladen. Das Betriebssystem (MUE) ist auf der Basis eines Disk-Operating-Systems (DOS) aufgebaut, so daß auch ein komfortables File-Handling (Datei-Organisation) mit einer Platte in BASIC möglich wird.

Es muß noch erwähnt werden, daß für diese Systemkonfiguration 2 Räume im 2. Stock gegenüber dem Sekretariat bereitgestellt wurden, die das städtische Hochbauamt in Zusammenarbeit mit dem Schulverwaltungsamt ausbaute und ausrüstete.

5) Projektarbeit

Die Arbeit des Schulversuchs Informatik spielt sich in der Hauptsache auf 2 Ebenen ab:

- a) Curriculumentwicklung (Entwurf und Planung) in Form von Kursen für die gymnasiale Oberstufe. Dazu treffen sich die Lehrer der beteiligten Schulen seit 1974 zweimal im Semester und beraten über die Entwürfe der Curriculum-Zentrale Gelsenkirchen. Die Schulkollegen und das Kultusministerium sind mit Fachreferenten vertreten und ebenfalls die Mitarbeiter des FEO LL. So sind schulfachliche Aufsicht und wissenschaftliche Begleitung sofort beim Entwurf dabei. Diese Gruppe ändert mit der Schulbasis auf Grund von bisherigen Erfahrungen und wissenschaftlichen Gutachten die Entwürfe ab.
- b) Erprobung an den Versuchsschulen in Form von Grundkursen. Fig. 2 zeigt den derzeitigen Stand der Kursthemen. Die Erweiterung (E) gilt für mögliche Leistungskurse.

Bisher wurden vom FEO LL 2 wissenschaftliche Gutachten angefordert:

Kerninformatik: Prof. Dr. A. Schmitt
Informatik-Inst. Universität Karlsruhe

Angew. Informatik: Prof. Dr. R. Reichl
Inst. f. Informatik u. Statistik, Universität Linz

Beide hatten erheblichen Einfluß auf eine Änderung der ersten Planung und auf die Weiterentwicklung. So wurde der zunächst starke Anteil der Ausbildung der Hardware (techn. Informatik) zurückgedrängt.

Nach dem derzeitigen Stand (15.11.76) ist folgende Kursfolge für den ersten Durchgang vorgesehen:

- | | | |
|------------|---|---|
| K 1 (11/1) | Einführungskurs | (Programmieren in einer problemorientierten Sprache) |
| K 2 (11/2) | Algorithmen I | (Strukturiertes Programmieren: Unterprogramme - 2 dimensionale Felder) |
| K 3 (12/1) | Aufbau und Funktionsweise einer ADVA | (Struktur einer digitalen DV-Anlage - Arbeit an stat. u. sequentiellen Bausteinen und am Modellrechner) |
| K 4 (12/2) | Datenstrukturen | (Datenorganisation und die Strukturierung) |
| K 5 (13/1) | Anwendung der Informatik | (z.B. in der Statistik, Physik, Wirtschaftswissenschaft u.a.) |
| K 6 (13/2) | Grundlagen der Informatik u. Wiederholung | (Vertiefung an Teilgebieten der Informationstheorie, Automatentheorie und formalen Sprachen) |

Figur 2

11. Jg. 1. Sem.	11. Jg. 2. Sem.	12. Jg. 1. Sem.	12. Jg. 2. Sem.	13. Jg. 1. Sem.	13. Jg. 2. Sem.
Kurs I Einführung in die Informatik	Kurs II Algorithmik I (Schwerpunkt Algorithmen)	Kurs III Aufbau und Funktionsweise von DVA	Kurs IV Algorithmik II (Schwerpunkt Datenstrukturen)	Kurs V Algorithmik III (Schwerpunkt Anwendung)	Kurs VI Ergänzungen und Prüfungs- vorbereitungen
	II E Vertiefung 1. Eigenschaften der verwendeten Programmiersprache 2. Algorithmen- begriff	III E Erweiterung 1. Analogrechner- prinzip 2. entweder masch. or. program- mieren oder Schalt- netze, Schaltwerke oder digitale Elektronik	IV E Vertiefung 1. Sprach- probleme 2. Elemente der Graphentheorie	V E Vertiefung durch Behandlung komplexerer Pro- bleme einschl. Erarbeitung dazu notwendiger theoretischer Grundlagen	VI E Ergänzungen und Prüfungs- vorbereitungen

Stand:
Juni 1976

KURSPLAN INFORMATIK

für die gymnasiale Oberstufe

Grundkursfolge:
Leistungskursfolge:

II - VI
II - VI

+ Ergänzungen E

Gesamtschule Berger Feld
der Stadt Gelsenkirchen
Fachbereich Datenver-
arbeitung und Informatik

6) Informatik als Wissenschaft und Schulfach

Dieser Kursaufbau erfaßt alle Teilgebiete der Informatik

Praktische Informatik: K 1, K 2, K 4

Technische Informatik: K 3

Theoretische Informatik: K 6, K 4

Angewandte Informatik: K 4, K 5

Davon rechnet man die ersten drei zu den Gebieten der Kerninformatik. Informatik, aus dem französischen ‚Informatique‘ entstanden, englisch ‚Computer Science‘, ist die Wissenschaft von der „Information“ und ihrer Verarbeitung mit digitalen und analogen Maschinen.

Diese auch an den Hochschulen noch junge Wissenschaft entstand aus mehreren Teilgebieten der bestehenden Wissenschaften: Elektronik, angewandte Mathematik, angewandte Physik und Kybernetik. Sie ist heute an vielen wissenschaftlichen Hochschulen durch Institute mit Lehrstühlen vertreten und hat nach C. F. v. Weizsäcker⁵⁾ den Charakter einer Strukturwissenschaft.

Neben der Materie und der Energie, die die Grundlagen für den ersten technischen Entwicklungsprozeß abgaben, tritt jetzt bewußt die „Information“ und leitet mit ihrer vielseitigen Verarbeitung einen weitergehenden Prozeß ein, den man auch die 2. technische Revolution nennt.

Prof. Dr. A. Schmitt begründet in seinem Gutachten vom 3.3.76 die Aufgabe des Schulfachs Informatik wie folgt:

„..... folgende Bildungswerte sollen vermittelt werden:

- 1) kunstsprachliche Darstellung von Informationen,
- 2) Darstellung von Algorithmen durch umgangssprachliche, formalsprachliche und grafische Beschreibung,
- 3) Abwicklung von Algorithmen, speziell an DV-Anlagen,
- 4) Eigenschaften, Möglichkeiten und Grenzen von Algorithmen.“

Im Mittelpunkt steht deshalb bei diesem Ansatz das Programmieren im erweiterten Sinne.

Dabei lernt der Schüler

- „1) Informationen zu analysieren und formal darzustellen.
- 2) ein für ihn durchsichtiges Problem präzise in eine algorithmische Lösung umzusetzen. Dabei werden sprachliche Darstellung (Programmiersprache) und die Datenstruktur je nach Vorgabe variiert,

- 3) exemplarisch die Abwicklung von selbsterstellten Algorithmen auf einer konkreten DV-Anlage und
- 4) die Komplexität und Qualität von Algorithmen zu beurteilen, selbständig Programme zu verbessern und evtl neue Lösungen zu entwickeln“ (A. Schmitt)

Zweitrangig sind dabei die Ziele: organisatorisch-technische Konzepte der DV-Anlage, Bedienungsfertigkeiten und technische Detailkenntnisse. Das schließt nicht aus, daß diese im einzelnen zur Motivation und zum Einstieg in bestimmte Probleme eingesetzt werden.

Im Gegensatz zur Hochschul-Informatik werden „mäßiger Abstraktionsgrad“, „Betonung der Anschaulichkeit“, keine „komplexen Gegenstände wie Compiler, Hardware-Strukturen, physikalisch-technologische Grundlagen“ gefordert.

7) Algorithmen

Damit steht für die Schul-Informatik der Begriff „Algorithmus“ im Mittelpunkt. Die beiden anderen Leitbegriffe „Datenstruktur“ und „Automat“ treten dagegen zurück.

Algorithmus, aus „algebra“ (arab.) und „arithmos“ (griech.) entstanden, bedeutet nach C.H. A. Koster ³⁾

„Algorithmus ist eine genaue, endliche Beschreibung eines Prozesses (zur Lösung einer Klasse von Aufgaben) (d. Verf.).“

Diese Beschreibung erfolgt in irgendeinem Formalismus, in Termen von andern Algorithmen und letztlich in elementaren Algorithmen.

Der Algorithmus muß ausführbar sein (von Mensch oder Maschine) und erlaubt Aussagen über den Prozeß und sein Ergebnis. Bei der Ausführung eines Algorithmus werden Objekte manipuliert, insbesondere finden Eingabe und Ausgabe von Objekten statt.“

Ein Beispiel aus dem Einführungskurs (11/1) soll diese Definition erläutern.

Problem. Aufsuchen der kleinsten Zahl M (Minimum) aus einer vorgegebenen Menge von N Zahlen (Z 1, Z 2, Z 3, ..., Z N)

Kurzname: SUCH-MIN

Analyse: 1) Wenn die Anzahl N = 1 ist, d. h. nur die Zahl Z 1 vorliegt, ist trivialerweise M = Z 1.

- 2) Wenn mehr als ein Element in der Menge enthalten ist ($N > 1$), dann nehme ich zunächst irgendeine Zahl der Menge als die kleinste Zahl M an. Dann überprüfe ich nacheinander alle andern Zahlen, ob eine davon kleiner ist als die zuerst gewählte. Ist das der Fall, so wird diese dann das neue Minimum M . Damit man weiß, wieviele Zahlen überprüft wurden, führt man einen Zähler (I) mit, der zunächst auf 1 gesetzt wird und dann bei jedem Vergleich mit einer weiteren Zahl um 1 erhöht wird ($I + 1$). Durch einen Blick auf den Zähler (Abfrage: Ist I größer als N) kontrolliert man, ob man alle Zahlen überprüft hat.

Damit kann ich den Algorithmus wie folgt beschreiben und formalisieren:

	Speicher
	← wird belegt mit
1) <u>Eingabe:</u>	Anzahl N , Zahlen Z_1, Z_2, \dots, Z_n
2)	Zähler wird auf 1 gesetzt ($I \leftarrow 1$)
3)	Die erste kleinste Zahl ist Z_1 ($M \leftarrow Z_1$)
4)	Zähler (I) wird um 1 erhöht ($I \leftarrow I + 1$)
5) <u>Kontrolle:</u>	Zähler (I) größer N ? ($I > N$)?
	ja: dann gehe nach 7) (Überprüfung fertig)
	nein: dann gehe nach 6) (Fortsetzung d. Überprüfung)
6) <u>Überprüfung:</u>	Z_i (i -te Zahl) kleiner als das bisherige
	Minimum M ($Z_i < M$)?
	ja: dann ist Z_i neues M ($M \leftarrow Z_i$)
	und gehe nach 4)
	nein: gehe sofort nach 4)
7) <u>Ausgabe:</u>	„Das Minimum der N Zahlen ist M “

Das ist eine genaue, endliche Beschreibung eines Such-Prozesses. Der Formalismus ist verbal und in Klammern auch bereits formal erfaßt.

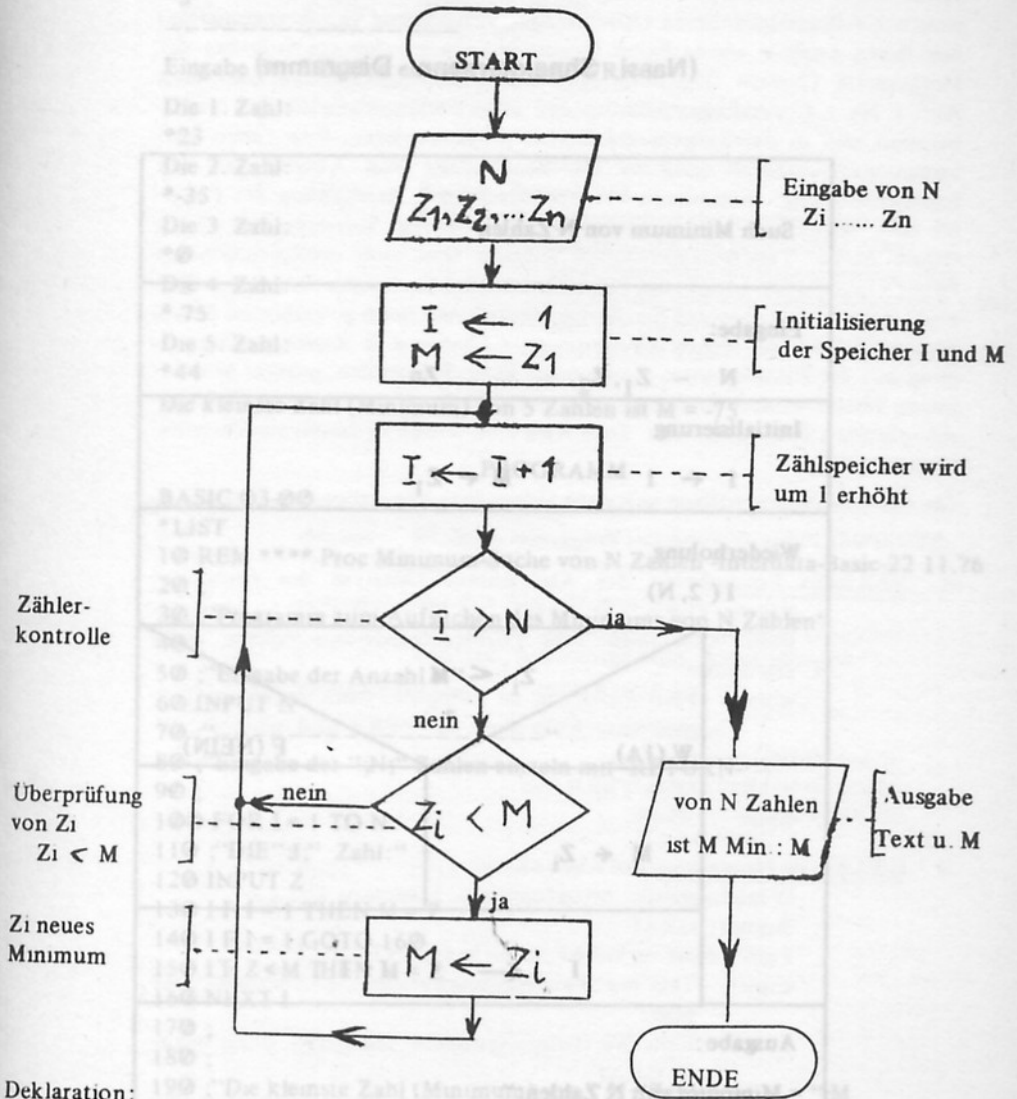
„Zähler erhöhen“, „Zähler kontrollieren“, „Minimum überprüfen“, „Gehe zu“ sind andere Algorithmen, die sich in elementare Schritte zerlegen lassen. Die Ausführung läßt sich an einem Beispiel (Speicherbelegungsplan) testen. Es werden hier Zahlen und Aussagen ($I > N$) manipuliert, und es findet eine Eingabe und Ausgabe von Objekten statt (numerische Daten: N, Z_1, Z_2, \dots, M , Zeichenfolge (String) „Minimum von N Zahlen.“)

Fig. 3 und 4 zeigt die grafische Darstellung des Algorithmus SUCH-MIN

Fig. 5 Testlauf und Programm-Listing

Figur 3
FLUßDIAGRAMM

Ablaufplan



Deklaration:

N Anzahl variable

Z_1, Z_2, \dots, Z_n

Zahlvariable

I Zählervariable

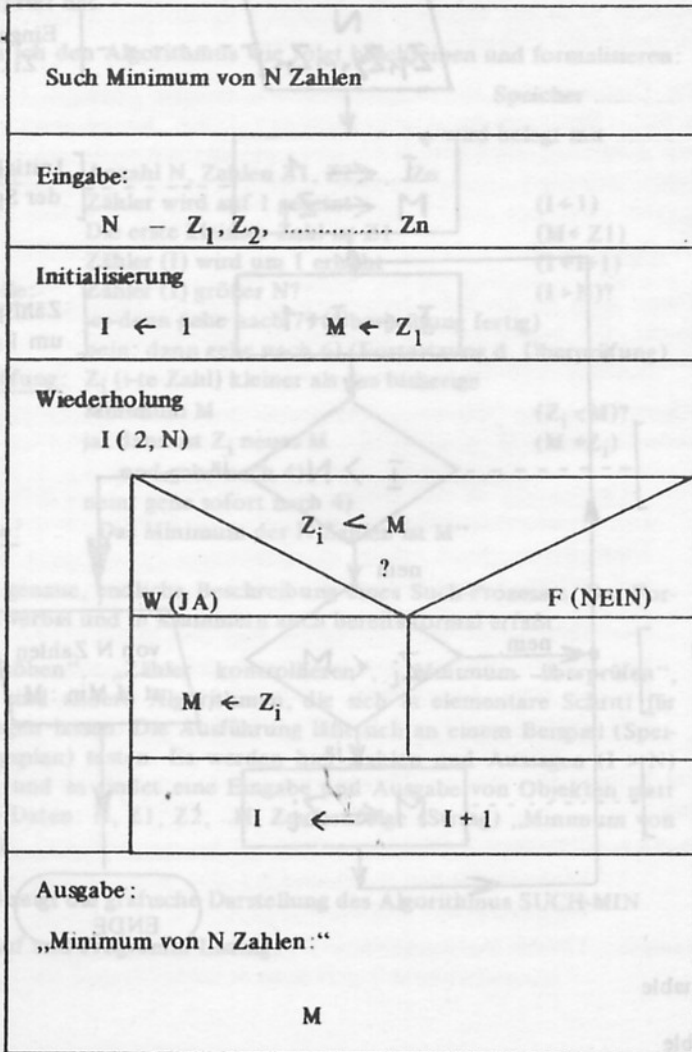
M Min variable

„von N“ String-Konstante

Figur 4

STRUKTOGRAMM

(Naasi - Shneidermann - Diagramm)



Figur 4

TESTLAUF

Eingabe der Anzahl N

*5

Eingabe der 5 Zahlen einzeln mit -RETURN-!

Die 1. Zahl:

*23

Die 2. Zahl:

*-35

Die 3. Zahl:

*0

Die 4. Zahl:

*-75

Die 5. Zahl:

*44

Die kleinste Zahl (Minimum) von 5 Zahlen ist M = -75

PROGRAMM

BASIC 03-00

*LIST

10 REM **** Proc Minimum-Suche von N Zahlen -Interdata-Basic-22 11.76

20 ;

30 ;'Programm zum Aufsuchen des Minimums von N Zahlen''

40 ;

50 ;'Eingabe der Anzahl N''

60 INPUT N

70 ;'-----''

80 ;'Eingabe der ''N'' Zahlen einzeln mit -RETURN-''

90 ;

100 FOR I = 1 TO N

110 ;'DIE ''I'' Zahl:''

120 INPUT Z

130 IF I = 1 THEN M = Z

140 IF I = 1 GOTO 160

150 IF Z < M THEN M = Z

160 NEXT I

170 ;

180 ;

190 ;'Die kleinste Zahl (Minimum) von ''N'' Zahlen ist M = ''M

200 ;

210 ;

220 END

8) Informatik-Unterricht am Zeppelin-Gymnasium

Im ersten Jahr des Schulversuchs (1975/76) nahmen in Jahrgangsstufe 11/1 42 Schülerinnen und Schüler an 2 Einführungskursen teil, die von Herrn OStR Hille und mir durchgeführt wurden. Nach einer ersten Diskussion von Alltagsalgorithmen (Kochrezept, Verordnung von Arzneimitteln) und deren Analyse wurde damit begonnen, bisher bekannte Aufgaben der Mathematik (lineare Gleichung $Ax + B = 0$, quadratische Gleichung $Ax^2 + Bx + C = 0$) algorithmisch mit allen Fallunterscheidungen aufzubereiten und in der Sprache BASIC zu programmieren. Von diesen Verzweigungsalgorithmen ging es zur Bearbeitung von Algorithmen mit Schleifenbildung: Summieren von Quadratzahlen, Berechnung der Fakultät ($n!$), Sortieralgorithmen (Min., Max-Sortierung, Sortieren (num.) der Größe nach). Tabellenalgorithmen leiteten über zum eindimensionalen Feld (Vektor oder Liste), das bei Einlese Schleifen, Ausdrucken der k Zahl eines Feldes, Verbesserung der Sortieralgorithmen beim gewichteten Mittel angewendet wurde. Neben Anwendungsproblemen z. B. Zinseszins-Berechnung bot die Zahlentheorie (Teiler einer Zahl, Primzahlen, größter gemeinsamer Teiler) weitere Probleme zum Aufstellen von Algorithmen zum anschließenden Programmieren. Zum Abschluß wurde in die String-Verarbeitung eingeführt.

Die einzelne Arbeit vollzog sich nach folgendem Algorithmus:

„SYSTPROG“ (systematisches Programmieren) ⁴⁾

A VERBales Formulieren des Algorithmus (Analyse des Problems)

Beginn VERB

Falls Kenntnisse zum Problemverständnis fehlen, dann erarbeiten

sonst: Falls Probleme zu komplex, dann zerlege in Teilprobleme formuliere sie VERB und setze sie zusammen,

sonst: setze VERB fort

Ende VERB

B GRAFische Darstellung des Algorithmus

(Flußdiagramm, Struktogramm, Ljapunow-Darst.)

Beginn GRAF

Falls Fehler im GRAF, dann beginn VERB

sonst: Test mit Wertbelegungstabelle (Speicherbelegungsplan)

Falls Wertbelegungstabelle fehlerhaft, dann beginne GRAF,

sonst: codiere in BASIC und bereite Testdaten vor

Ende GRAF

C TESTlauf am Rechner (Testen des Programms)

Beginn TEST
Beginn Eingabe
Falls Eingabefehler auftreten, dann berichtige sie
sonst: weitere Eingabe
Ende Eingabe
Beginn Testlauf
Falls Fehler auftreten, dann
Falls Codierungsfehler,
dann berichtige sie
sonst: beginne GRAF
sonst: Beginn Überprüfung
Falls Programmänderung notwendig, dann
Minimierung des Programms oder
Strukturierung (REM-Statement) oder
Einführung von Benutzeranweisungen oder
sonstige Änderungen
und Beginn TEST
sonst: Ende Überprüfung
Ende Testlauf
Ende TEST

D DOKumentation des Algorithmus

Beginn DOK
Daten-Deklaration (Variablenliste)
Aussagen über den Algorithmus
Art (linear, verzweigt, ...). Aufwand (Speicher)
Laufzeit, Schwierigkeiten, Anwendungen
Dokumentation: VERB-, GRAF-
TEST-: Testlauf
Programm (Befehlsliste):
Listing und Lochstreifen
od Markierungskarte od.
Magnetband)
Ende DOK

Zum Kurs in 11/2, Thema: Algorithmen I, meldeten sich 19 Schülerinnen und Schüler, von denen dann 16 bis zum Schluß durchhielten. Das Lernziel des Kurses: Strukturiertes Programmieren wurde am Gegenstand der Prozedur-Technik (Unterprogramm-T.) und der zweidimensionalen Felder (Matrizen und String-Felder) an zahlreichen Problemen eingeübt. Dabei wurden erstmals von Gruppen einzelne Projekte bewältigt. In dem laufenden Schuljahr setzte sich der Grundkurs Informatik in 12/1 mit „Aufbau und Struktur einer ADV-Anlage“ fort (Kursleiter: OstR Hille, 10 Schüler).

Weiter begannen in diesem Schuljahr 2 neue Grundkurse in 11/1, (Kursleiter: StD' Franzen (GSG), StD Giedinghagen - 34 Schüler und Schülerinnen).

Die 3 Gk-Wochenstunden werden im allgemeinen so aufgeteilt, daß in 2 Stunden Probleme zum Programmieren vorbereitet werden und 1 Stunde am Terminal des Rechners INTERDATA 7/16 gearbeitet wird. Dabei wird für jeden Schüler diese Unt.-Stunde zu einer Doppelstunde zusammengefaßt, die dann 14-tägig im Wechsel nachmittags stattfindet. Dabei bildet man Gruppen von ungefähr 8 Schülern, die dann zeitlich wechselnd im Dialog mit der Maschine vorbereitete Programme oder auch eigene eingeben und testen.

Trotz zahlreicher Schwierigkeiten (technische, organisatorische, didaktische und methodische) konnten, wenn auch zum Teil eingeschränkt, die gesteckten Kurslernziele erreicht werden. Der starke Rückgang der Teilnehmer des ersten Versuchsjahrgangs hat, wie eine Analyse zeigte, verschiedene Ursachen. Neben technischen (Std. Plan-Block, Nachmittags-Unterricht, nur 2 Gk auf die Naturw. Pflichtbindung anrechenbar, zu geringe Übungszeit am Terminal) überwogen die Angaben, die dieses Fach als schwierig und zeitaufwendig einstufen.

9) Abschließende Betrachtungen

Vergleicht man das Fach Informatik mit den andern des mathematisch-naturwissenschaftlichen Aufgabenfeldes, so liegt es der Physik am nächsten. Auch hier liegt eine enge Wechselwirkung zwischen Theorie und Praxis vor. Dem Aufstellen von Theorien (Modellen) über Begriffe und Gesetze in der Physik entsprechen hier Aufstellen von Algorithmen aus Teil- und Elementaralgorithmen, die Daten in verschiedenster Form verarbeiten. Wie in der Physik jede Theorie verworfen wird, die im Widerspruch steht zur experimentellen Überprüfung, so hat auch hier kein Algorithmus Bestand, der sich nicht dem Test an der DV-Maschine unterzieht und das bringt, was er vorgibt zu lösen. Dem Arbeiten am Experi-

ment bei Schülerübungen entspricht das Arbeiten am Terminal. Nur daß hier durch den Dialog mit der DV-Maschine der Schüler selbst seine Leistung schnell überprüfen kann. Der Lehrer tritt bei dieser Arbeitsweise wie auch bei Schülerübungen in den Hintergrund. Er fungiert als Berater.

Wie selbständige Problemlösungen von Schülern zeigen, die neben dem Kanon der Pflicht-Algorithmen entstanden, fördert das Arbeiten mit Algorithmen am Computer in hohem Maße die Kreativität. Darüber hinaus erzieht die Arbeit in diesem Fach zum sorgfältigen Strukturieren von Problemen und zum präzisen Arbeiten bei der praktischen Überprüfung der Lösungen. Darüber hinaus sollen mit der gesamten Arbeit in der reformierten gymnasialen Oberstufe Grenzen und Möglichkeiten der Informationsverarbeitung einschließlich ihrer Folgen deutlich gemacht werden. Damit leistet das Fach Informatik einen wichtigen Beitrag zum Verständnis der Probleme der heutigen Gesellschaft und bereitet mit auf das Studium vor, das heute im allgemeinen ohne die Anwendung der Informatik (Datenverarbeitung) nicht mehr denkbar ist.

Zum Schluß darf ich allen beteiligten Kolleginnen und Kollegen und den mit der Planung und Durchführung beauftragten Dienststellen sehr herzlich für die bisher geleistete Arbeit danken, besonders aber unserem Schulleiter, Herrn Dr. H. Bartmann, der für die zum Teil nicht geringen Probleme und Sorgen immer ein offenes Ohr hatte und mithalf, sie zu lösen.

Literatur:

- 1) H. Balzert, Informatik 1, München 1976
- 2) V. Claus, Einführung in die Informatik, 1975
- 3) C. H. A. Koster, Algorithmen I
Vorlesungs-Nachschrift WS 73/74,
FB 20, TU Berlin
- 4) Rahmenlehrplan-NRW, Informatik in der gymnasialen Oberstufe,
Gelsenkirchen Juni 1976
- 5) C. F. v. Weizsäcker, Die Einheit der Natur, München 1971

F. - W. Giedinghagen