

Experimentallehrplan

Informatik

Klasse 11

Ministerrat
der Deutschen Demokratischen Republik
Ministerium für Volksbildung

und Wissen Volkseigener Verlag Berlin

MINISTERRAT DER DEUTSCHEN DEMOKRATISCHEN REPUBLIK
MINISTERIUM FÜR VOLKSBIIDUNG

Experimentallehrplan
für den obligatorischen Unterricht in den Klassen 11
der erweiterten Oberschule

INFORMATIK

1. Auflage

Lizenz Nr. 203/1000/89 (30 10 90)

LSV 0670

Printed in the German Democratic Republic

Gesamtherstellung: VOB Nationales Druckhaus, Berlin

Verlagstitelnummer: 30 10 90-1

Der Experimentallehrplan für den obligatorischen Informatik-
unterricht in den Klassen 11 der erweiterten Oberschule tritt
am 1. September 1989 in Kraft.

Stellvertreter des Ministers
H. Drechsler

Ziele und Aufgaben

Der Informatikunterricht der Klasse 11 hat das Ziel, allen Schülern solide Kenntnisse über grundlegende Begriffe, Methoden und Verfahren der Informatik sowie sichere Grundfertigkeiten im Handhaben der an der Schule vorhandenen informationsverarbeitenden Technik zu vermitteln und sie auf dieser Basis zu befähigen, zielgerichtet und zunehmend selbständig Computer zum Bearbeiten und Lösen vielfältiger Aufgaben zu nutzen. Damit verbunden sollen die Schüler ein elementares Verständnis für Gegenstand und wesentliche Inhalte der Informatik sowie für die Leistungsfähigkeit und volkswirtschaftliche Bedeutung der informationsverarbeitenden Technik erwerben.

Der Unterricht im Fach Informatik muß über den Erwerb des genannten spezifischen Wissens und Könnens hinaus gewährleisten, daß die in der Beschäftigung mit dem fachlichen Gegenstand liegenden Potenzen für die Entwicklung der Persönlichkeit der Schüler als Ganzes zielstrebig und systematisch genutzt werden. Insbesondere soll durch die Aneignung von Denk- und Arbeitsweisen der Informatik sowie durch das Lösen von Aufgaben mittels des Computers zur Denk- und Sprachentwicklung der Schüler beigetragen, ihre Selbständigkeit und geistige Beweglichkeit, ihr Interesse und ihre Freude an angestrebter, disziplinierter und auch elementarschöpferischer geistiger Arbeit gefördert und weiter ausgebildet werden.

Damit das genannte Ziel des Informatikunterrichts erreicht wird, ist es erforderlich, die Arbeit an der Realisierung insbesondere folgender Teilziele in das Zentrum zu stellen:

- Die Schüler kennen den Begriff "Algorithmus" und wesentliche Eigenschaften von Algorithmen. Sie besitzen sichere Kenntnisse über Algorithmenstrukturen (in dem im Stoffteil angegebenen Umfang) und wesentliche Möglichkeiten ihrer Darstellung, über Datentypen und -strukturen und haben sich die für das Nutzen der zur Verfügung stehenden Computer benötigten Befehle fest eingeprägt.
- Die Schüler haben das Können erworben, numerische und nichtnu-

merische Aufgaben angemessener Schwierigkeit zu vielfältigen, ihnen aber prinzipiell vertraute Problemkreise zu analysieren, geeignete und ihnen im wesentlichen bekannte mathematische Modelle zum Lösen auszuwählen, entsprechende Algorithmen zu entwickeln und diese in einer problemorientierten Programmiersprache (z. B. BASIC) darzustellen. Die Schüler sind in der Lage, ausgewählte Aufgaben und Probleme zur Prozeßdatenverarbeitung sowie zur Text-, Grafik- oder Dateiverarbeitung mittels bereitgestellter Software zu lösen.

- Die Schüler können mit den in der Schule vorhandenen Computern und ihren peripheren Geräten sicher umgehen. Sie besitzen insbesondere die Kenntnisse, Fähigkeiten und Fertigkeiten, Programme in den Computer einzugeben, zu testen, zu korrigieren und zu dokumentieren.
- Im Prozeß der eigenen Arbeit sind die Schüler dazu erzogen worden, mit den hochwertigen Geräten der Mikroelektronik sorgsam umzugehen, Bedienungsanleitungen exakt einzuhalten, Ordnung und Übersichtlichkeit am Arbeitsplatz wie in den persönlichen Unterlagen zu wahren. Die Befähigung und Bereitschaft der Schüler zu kollektiver Arbeit bei individuellem Leistungsbedürfnis und Verantwortungsbewußtsein, zu Sorgfalt, Exaktheit, Zuverlässigkeit, Kritik und Selbstkritik - einschließlich einem kritischen Verhalten gegenüber den von der Informationsverarbeitungstechnik gelieferten Resultaten - sind weiterentwickelt worden.
- Die Schüler haben ein prinzipielles Verständnis für die gesellschaftliche Bedeutung der Informatik und informationsverarbeitenden Technik als Schlüsseltechnologie gewonnen und erkennen in diesem Zusammenhang die Notwendigkeit der Verbindung der Vorzüge des Sozialismus mit dem wissenschaftlich-technischen Fortschritt als ein wesentlicher Faktor für die gesellschaftliche Entwicklung insgesamt.
- Durch die Bearbeitung vielfältiger Aufgaben und Probleme sind die Schüler zu der Einsicht gelangt, daß die Nutzung von Informatik und informationsverarbeitender Technik es gestattet, viele Anforderungen schneller als mit anderen Hilfsmitteln zu erfüllen. Sie erkennen, daß für manche Probleme ein anderer

oder überhaupt erst ein Lösungsansatz möglich wird. Zugleich haben die Schüler die Grenzen der Einsatz- und Verwendungsmöglichkeiten von Informatik und informationsverarbeitender Technik erkannt.

Der Lehrgang ist in folgender Weise gegliedert:

Im 1. Teil (identisch mit dem Stoffgebiet 1) soll hinsichtlich des elementaren Wissens und Könnens auf dem Gebiet der Informatik und der Computernutzung eine tragfähige Ausgangsbasis aller Schüler für erfolgreiches Lernen im weiteren Informatikunterricht geschaffen werden. Dabei sind die unter Umständen sehr differenzierten Voraussetzungen genau zu beachten, die die einzelnen Schüler in ihrem bisherigen schulischen und individuellen Bildungsgang erworben haben.

Das Anliegen des 2. Lehrgangsteils (Stoffgebiete 2 und 3) besteht darin, die Schüler durch das zunehmend selbständige Lösen von Aufgaben unter Nutzung von Computern mit der Vorgehensweise beim Bearbeiten sowohl numerischer als auch nichtnumerischer Problemkreise unter Verwendung von vorgefertigter bzw. selbstentwickelter Software vertraut zu machen und entsprechendes elementares Können zu entwickeln.

Hinweise zur methodischen und organisatorischen Gestaltung des Unterrichts

Geleitet von einem auf solides Wissen und Können orientierten Informatiklehrgang, muß die methodische Gestaltung des Unterrichts von dem Grundprinzip gekennzeichnet sein, die Vermittlung und Aneignung von Kenntnissen, Fähigkeiten und Fertigkeiten, von Einstellungen und Verhaltensweisen stets durch ein entsprechend gestaltetes Arbeiten mit Aufgaben unter möglichst hoher Selbständigkeit der Schüler zu realisieren. Insbesondere ist der Erwerb der theoretischen Kenntnisse stets mit der praktischen Arbeit an Problemen oder der Tätigkeit am Computer zu verbinden. Dadurch soll zugleich die Aneignung eines Grundverständnisses für den Gegenstandsbereich der Informatik sowie für die Möglichkeiten und Grenzen des Einsatzes dieser Wissenschaft unterstützt werden.

Den durchgängigen inhaltlichen Hauptakzent bildet das algorithmische Arbeiten. Die erforderlichen Algorithmenelemente

werden in den Darstellungsformen Programmablaufplan und Struktogramm erarbeitet, wobei an die den Schülern bereits aus dem Mathematikunterricht bekannten Rechenablaufpläne angeknüpft werden sollte.

Es empfiehlt sich, für die Gesamtdarstellung von Algorithmen Struktogramme zu bevorzugen, da sie das anzustrebende strukturierte Programmieren fördern. Die Vermittlung der für die Darstellung der Algorithmen in einer problemorientierten Programmiersprache (z. B. BASIC) erforderlichen Sprachelemente sollte dann stets in Verbindung mit den Erfordernissen der jeweilig verwendeten Algorithmenstrukturen erfolgen.

Beim Bearbeiten von Aufgaben ist ein ausgewogenes Verhältnis zwischen Dialog-Arbeit am Computer und vorhergehender, also "computerferner" Programmentwicklung anzustreben. Insbesondere bei komplizierten Aufgaben ist darauf zu achten, daß die Schüler nicht sofort unüberlegt mit dem Programmieren beginnen, sondern die unmittelbare Tätigkeit am Computer zunächst durch gründliche Vorüberlegungen zum Programm vorbereiten. Durch eine solche Vorgehensweise sollen die Schüler zugleich zu der Einsicht geführt werden, daß Methoden wie problemorientierte Datenaufbereitung, strukturiertes und systematisches Programmieren (insbesondere durch konsequente Verwendung der algorithmischen Grundstrukturen und modulares Arbeiten) den Problemlöseprozeß wesentlich unterstützen. Beim Arbeiten mit BASIC empfiehlt es sich, konsequent von BASIC-Ersatzstrukturen Gebrauch zu machen, wenn ein direktes Übertragen der jeweiligen Algorithmenstruktur nicht möglich ist. Bei all diesen vorbereitenden und begleitenden Tätigkeiten darf nicht außer Betracht geraten, daß die gestellten Aufgaben im allgemeinen von den Schülern auch mit dem Computer wirklich zu lösen sind.

Die Wahl der im Informatikunterricht zu bearbeitenden Aufgaben muß so erfolgen, daß die hierbei von den Schülern zu erbringenden Leistungen in erster Linie im Auswählen bzw. Aufstellen geeigneter Algorithmen und deren Darstellen in der gewählten Programmiersprache sowie ihrem Abarbeiten am Computer bestehen. Aus diesem Grunde sollten im allgemeinen keine solchen Aufgaben gestellt werden, wo bereits für das Vertrautmachen mit dem jeweiligen Sachverhalt umfangreiche und zusätzliche Erkennt-

nisse erforderlich sind. Durch die Aufgabenauswahl muß außerdem die Verwendung von Algorithmenstrukturen in natürlicher Weise motiviert werden.

Große Bedeutung kommt im Informatikunterricht einem sinnvoll differenzierten Vorgehen zu, das die unterschiedlichen Interessen und Begabungen, die unter Umständen sehr unterschiedlichen allgemeinen und fachlich-spezifischen Voraussetzungen der einzelnen Schüler optimal nutzt und zugleich gewährleistet, daß jeder Schüler die Ziele des im Informatikunterricht zu leistenden Beitrags zur Allgemeinbildung auch erreicht.

Eine gründliche Analyse und durchgängige Beachtung des Entwicklungsstandes der einzelnen Schüler ist deshalb besonders wichtig, weil in Abhängigkeit von den im "Grundkurs Informatik" des ESP-Unterrichts der Klasse 9 erreichten Ergebnissen, unterstützt durch die Teilnahme von Schülern an fakultativen Lehrgängen, anderen Arbeitsgemeinschaften und Kursen sowie durch die interessenbestimmte individuelle Tätigkeit das Ausgangsniveau unter Umständen recht verschieden ausgeprägt ist. Außerdem werden sich auch im nachfolgenden Unterricht beträchtliche Differenzierungen im Aneignungstempo, im Festigungsbedarf, in der Selbständigkeit der Schüler zeigen, deren Berücksichtigung wesentlichen Einfluß auf den gesamten Bildungs- und Erziehungserfolg ausübt.

Hauptansatzpunkte für Differenzierungsmaßnahmen im Informatikunterricht stellen neben einer geeigneten Aufgabenauswahl vor allem unterschiedliche Ansprüche an den beim Lösen zu erreichenden Selbständigkeitsgrad, an die von dem einzelnen Schüler beim Arbeiten zu erreichende programmtechnische Qualität oder auch an die Art seiner Einführung in den kollektiven Lernprozeß (z. B. als Leiter oder Mitglied einer Arbeitsgruppe, als verantwortlicher Bearbeiter eines Teilproblems oder "Organisator" der Synthese von Einzel- zur Gesamtlösung, als Diskussionsleiter oder Vortragender zu einem Lösungsansatz u. ä.) dar. Hinweise auf weitere inhaltliche Differenzierungsmöglichkeiten geben ferner die im Stoffteil enthaltenen eingerückten Passagen: Sie kennzeichnen Stoffe, die nicht verbindlicher Aneignungsgegenstand sind, sondern vorrangig für die Ergänzung oder Erweiterung des Wissens

und Könnens derjenigen Schüler genutzt werden sollten, die die einheitlichen Ziele bereits erreicht haben.

Für das Vertrautmachen der Schüler mit der Verwendung vorgefertigter Software für spezielle Anwendungen stehen insgesamt 14 Stunden zur Verfügung. Davon sind 6 für den Themenkreis "Prozeßautomatisierung" zu nutzen. Aus den Themen "Dateiarbeit", "Textbearbeitung" und "Grafiksysteme" sind vom Lehrer entsprechend den konkreten pädagogischen und materiell-technischen Bedingungen ein oder zwei Einsatzgebiete auszuwählen, für die der Plan dann insgesamt 8 Stunden ausweist.

Es ist dem Lehrer außerdem freigestellt, das 3. Stoffgebiet entweder in der durch die Stoffübersicht angegebenen Weise in den Gesamtlehrgang einzufügen oder im Interesse einer engeren Verbindung der Arbeit mit selbständig erarbeiteter und vorgefertigter Software so zu verteilen, daß die Themen "Textbearbeitung" oder "Dateiarbeit" im Anschluß an den Stoffabschnitt 1.6. behandelt werden bzw. das Thema "Grafiksysteme" an 3.3. angeschlossen wird. Damit würde sich dann der Stoffabschnitt "3.2. Prozeßautomatisierung" unmittelbar an den Abschnitt 2.4. anschließen.

1

STOFFÜBERSICHT

1.	Grundlagen der Computernutzung zum Lösen von Aufgaben; Daten; Algorithmenstrukturen	28 Stunden
1.1.	Grundbegriffe der Informatik; Verwenden des Computers beim Lösen von Aufgaben; Datentypen; lineare Algorithmen (Folgen)	(6 Stunden)
1.2.	Algorithmen mit gezählter Wiederholung	(4 Stunden)
1.3.	Algorithmen mit Verzweigungen	(6 Stunden)
1.4.	Algorithmen mit Wiederholungen; Datenstruktur Feld	(6 Stunden)
1.5.	Unteralgorithmen	(4 Stunden)
1.6.	Systematisierung	(2 Stunden)
2.	Lösen vielfältiger Aufgaben mit dem Computer durch selbständiges Erstellen von Programmen	18 Stunden
2.1.	Schritte beim Aufgabenlösen	(2 Stunden)
2.2.	Lösen numerischer Aufgaben	(6 Stunden)
2.3.	Lösen von Aufgaben mit grafischen Darstellungen	(6 Stunden)
2.4.	Übung und Anwendung durch das Lösen einer komplexen Aufgabe	(4 Stunden)
3.	Arbeit mit vorgefertigter Software und Benutzersystemen	14 Stunden
3.1.	Textbearbeitung/Dateiarbeit/Grafiksysteme	(8 Stunden)
3.2.	Prozeßautomatisierung	(6 Stunden)

insgesamt: 60 Stunden

1. Grundlagen der Computernutzung zum Lösen von Aufgaben; Daten;
Algorithmenstrukturen 28 Stunden

Das Hauptziel des ersten Stoffgebiets besteht darin, bei allen Schülern sichere Kenntnisse und Fertigkeiten im Hinblick auf den Umgang mit dem Computer und vor allem Können im bewußten Nutzen algorithmischer Grundstrukturen zum Lösen vielfältiger Aufgaben herauszubilden bzw. zu vertiefen sowie zu erweitern.

Im Stoffabschnitt 1. 1. ist durch das Lösen entsprechender Aufgaben und durch unmittelbare Arbeit am Computer zu erreichen, daß alle Schüler über solide Kenntnisse zu solchen wesentlichen Bestandteilen einer Informatik-Bildung wie einem "naiven" Algorithmus-Begriff und Eigenschaften von Algorithmen, Datentypen, Kommando, Anweisung und Programm verfügen, daß sie mit dem prinzipiellen Aufbau eines Computers vertraut sind und diesen zuverlässig bedienen können. Dabei sind die dazu benötigten Elemente der verwendeten Programmiersprache (z. B. BASIC) einzuführen bzw. zu wiederholen. An das Wissen und Können der Schüler aus dem Mathematikunterricht (Algorithmusbegriff¹, Arbeiten mit Variablen, Nutzen von Ablaufplänen für den Taschenrechner, Fallunterscheidung) und aus dem ESP-Unterricht, speziell dem "Grundkurs Informatik" in Klasse 9 (Fertigkeiten im sachgemäßen Umgang mit einem Computer, Kenntnisse über Schritte der Programm-entwicklung, Wissen und Können im Erarbeiten und Nutzen einfacher kleiner Programme) ist bewußt anzuknüpfen. Im Zusammenhang mit historischen Betrachtungen zur Entwicklung der Rechentechnik sollte auf Möglichkeiten und Grenzen des Einsatzes informationsverarbeitender Technik hingewiesen werden.

In den Stoffabschnitten 1. 2., 1. 3., 1. 4. und 1. 5. vertiefen und erweitern die Schüler durch das Bearbeiten geeignet ausgewählter Beispielaufgaben ihre Kenntnisse über algorithmische Grundstrukturen. Sie lernen die

¹ Auf Probleme, die mit einem exakten mathematischen Algorithmusbegriff im Zusammenhang stehen, ist hier nicht einzugehen. Auf solche können interessierte Schüler gegebenenfalls im Verlaufe des Lehrgangs hingewiesen werden.

jeweils neuen Grundstrukturen in Verbindung mit den bereits vorher behandelten zum zunehmend selbständigen Lösen entsprechender (numerischer und nichtnumerischer) Aufgaben zu verwenden. Dabei wird die Sicherheit im Darstellen dieser Strukturen in der verwendeten Programmiersprache und damit zugleich das Beherrschen der erforderlichen Sprachelemente erhöht.

Anhand konkreter Aufgaben sind die Schüler außerdem mit dem Gebrauch der logischen Operationen Negation, Konjunktion und Alternative auf elementarer Stufe (und ohne gesonderte Reflexion über Wahrheitswerte) vertraut zu machen. Im Zusammenhang mit der Behandlung von Algorithmen mit Wiederholung lernen die Schüler auch die Datenstruktur Feld kennen. Im Stoffabschnitt 1.5. (Unteralgorithmen) sammeln die Schüler auch erste Erfahrungen beim Arbeiten mit dem Zufallszahlengenerator. Umfang und Schwierigkeitsgrad von Aufgaben, zu deren Lösen Felder bzw. der Zufallszahlengenerator Verwendung finden, sollten in Abhängigkeit von Leistungsvermögen und Interessenlage der Schüler differenziert gestaltet werden. Die Aufgabenauswahl sollte ferner gewährleisten, daß die Schüler Verständnis für die vielfältigen Anwendungsmöglichkeiten eines Computers (z. B. Verarbeiten von Zeichenketten oder Erzeugen von Grafiken) aber auch erste Einsichten über Grenzen der Computernutzung (z. B. durch numerische Ungenauigkeiten aber auch durch nichtalgorithmierbare Probleme) gewinnen. Die Aufgabenauswahl und der Prozeß des Lösens sind so zu gestalten, daß alle Schüler hinsichtlich ihrer Einstellungen und Haltungen sowie ihrer Charakter- und Willenseigenschaften angemessen gefordert und entwickelt werden.

Im Stoffabschnitt 1. 6. steht eine Systematisierung des Wissens der Schüler über Daten und Algorithmenstrukturen im Zentrum.

1.1. Grundbegriffe der Informatik; Verwenden des Computers beim Lösen von Aufgaben; Datentypen; lineare Algorithmen (Folgen)
(6 Stunden)

Wiederholen und Vertiefen des Begriffes "Algorithmus" und der Eigenschaften von Algorithmen, Darstellungsformen und Beispiele für Algorithmen; Daten (numerisch/nichtnumerisch) und entsprechende Variable, Funktionseinheiten eines Computers (Zentraleinheit,

Arbeitsspeicher, Tastatur, Bildschirm, externe Speicher, Drucker/Plotter).

Kommando- und Programmmodus eines Computers anhand von Beispielen (beschränkt auf unverzweigte Programme).

Kommando und Anweisung (anhand von Beispielen in der gewählten Programmiersprache); Programm als Folge von Anweisungen; Laden von Programmen vom externen Speicher; Starten; Auflisten, Editieren von Programmen; Wertzuweisung; Aufstellen einfacher Programme; Eingeben, Verarbeiten und Ausgeben von Daten (jeweils unter Verwendung der entsprechenden Elemente der verwendeten Programmiersprache).

Lösen einfacher Aufgaben; Übersicht über die historische Entwicklung der Rechentechnik; Hinweise auf Möglichkeiten und Grenzen der Nutzung dieser Technik.

1.2. Algorithmen mit gezählter Wiederholung (4 Stunden)

Beispiele für Algorithmen mit einer festgelegten Zahl von Wiederholungen (Zählschleifen); Darstellen solcher Algorithmen in der gewählten Programmiersprache (gegebenenfalls vorher auch als Struktogramm); Übungen im Aufstellen, Testen und Abarbeiten von Programmen, die Zählschleifen enthalten.

Grafische Darstellungen im Zusammenhang mit dem Verwenden von Zählschleifen, Nutzen der Symbol- bzw. Punktgrafik des Computers (je nach Computertyp); Anwenden der Möglichkeiten des Computers zur grafischen Darstellung von Sachverhalten aus unterschiedlichen Bereichen.

Unterschiedliche Möglichkeiten der Bildschirmgestaltung. Lösen vielfältiger numerischer und nichtnumerischer Aufgaben.

Schachtelung von Zählschleifen, Betrachtungen zum Aufwand entsprechender Programme.

1.3. Algorithmen mit Verzweigungen (6 Stunden)

Wiederholung von Fallunterscheidungen aus dem Mathematikunterricht. Beispiele für Algorithmen mit Verzweigung in Abhängigkeit von einer Auswahlbedingung (zweiseitige Auswahl, einseitige Auswahl als Spezialfall); Darstellung derartiger Algorithmen in der gewählten Programmiersprache (gegebenenfalls vorher auch als Struktogramm); Übungen im Aufstellen, Testen und Abarbeiten der-

artiger Programme. Logische Operationen "Negation", "Konjunktion" und "Alternative", entsprechende logische Operationen in der gewählten Programmiersprache. Einführen der Algorithmenstruktur Fallauswahl; Darstellen derartiger Algorithmen in der gewählten Programmiersprache (gegebenenfalls vorher auch als Struktogramm). Übungen im Aufstellen, Testen und Abarbeiten von Programmen mit Verzweigungen, deren Bedingungen auch logische Operationen enthalten.

1.4. Algorithmen mit Wiederholungen; Datenstruktur Feld

(6 Stunden)

Beispiele für Algorithmen mit Wiederholungen (Schleifen), dabei Unterscheiden von Wiederholungen mit nachgestelltem Test (Wiederholerschleife) und Wiederholungen mit vorangestelltem Test (Solangeschleife), Darstellen beider Arten der Algorithmen mit Wiederholungen als Programmablaufplan, als Struktogramm und in der gewählten Programmiersprache; Übungen im Aufstellen, Testen und Abarbeiten von Programmen, die Schleifen enthalten. Lösen von Aufgaben, bei denen zweckmäßigerweise entweder die Wiederholerschleife oder die Solangeschleife (in Abhängigkeit vom Computertyp) Anwendung findet.

Einführen der Datenstruktur Feld an Beispielen für ein- und zweidimensionale Felder (auch aus dem nichtnumerischen Bereich); Einführen von "Feldelement", "Index", "Dimension"; Verwenden von Feldern zur Erfassung von Sachverhalten aus der Numerik, Grafik oder Textbearbeitung; Übungen im Aufstellen, Testen und Abarbeiten von Programmen mit Feldern.

1.5. Unteralgorithmen

(4 Stunden)

Unteralgorithmen und ihre Darstellung in der gewählten Programmiersprache; Vorteile der Unterprogrammtechnik; Übungen im Aufstellen und Testen von Programmen, die Unterprogramme enthalten.

Verschachtelungsmöglichkeiten von Unterprogrammen; Problem der Parametervermittlung und der Rekursion in BASIC.

Zufallszahlengenerator; Gewinnung von Pseudozufallszahlen durch Verwendung einer Standardfunktion; Lösen von Aufgaben mit Hilfe des Zufallszahlengenerators.

1.6. Systematisierung

(2 Stunden)

Übersicht über die behandelten Algorithmenstrukturen (Folge; einseitige, zweiseitige und mehrseitige Auswahl; Zähl-, Wiederhol- und Solangeschleife, Unteralgorithmus), ihre Darstellung als Programmablaufplan oder Struktogramm und in der gewählten Programmiersprache; Typische Beispiele für das Nutzen dieser Algorithmenstrukturen.

Übersicht über die behandelten Datentypen und -strukturen. Übersicht über die bisher verwendeten Operationen, Relationen und Standardfunktionen für die Datentypen Zahl, Zeichen, Zeichenkette.

2. Lösen vielfältiger Aufgaben mit dem Computer durch selbständiges Erstellen von Programmen

18 Stunden

Im Zentrum dieses Stoffgebietes steht die Befähigung der Schüler zum selbständigen Erarbeiten von Programmen, wobei das im ersten Stoffgebiet erworbene Wissen und Können bezüglich Algorithmenstrukturen und Datentypen anzuwenden ist.

Es stehen solche Aufgaben im Mittelpunkt, mit denen den Schülern einerseits die hohe Arbeitsleistung von Computern verdeutlicht werden kann und andererseits gezeigt wird, daß Computerergebnisse stets einer kritischen Wertung bedürfen. Gleichzeitig erhalten die Schüler einen ersten Einblick in verschiedene Anwendungsgebiete informationsverarbeitender Technik.

Im **S t o f f a b s c h n i t t** A. 1. ist anhand eines Beispiels der Prozeß des AufgabenlöSENS mit Hilfe des Computers durch die Schüler inhaltlich so zu erschließen, daß eine entsprechende Schrittfolge für jeden Schüler als Orientierungsgrundlage für das LöSEN weiterer Aufgaben zur Verfügung steht. Hierbei ist besonderes Gewicht auf die Etappen der Algorithmierung und Programmierung (einschließlich Test und Korrektur) zu legen.

Schwerpunkt des **S t o f f a b s c h n i t t e s** 2. 2. ist das selbständige LöSEN von Aufgaben mit numerischen Inhalten. Die Schüler sollen erkennen, daß Algorithmen, die rekursive und iterative Verfahren beschreiben, sehr geeignet sind, vielfältige

mathematische Aufgaben vorteilhaft mit dem Computer zu lösen. Dabei lernen sie die Verfahren Rekursion und Iteration an konkreten Beispielen inhaltlich kennen. In diesem Zusammenhang lernen die Schüler benutzerdefinierte Funktionen kennen und anzuwenden. Des Weiteren werden die Schüler mit der Methode der vollständigen Durchmusterung eines großen (aber endlichen) Variablengrundbereichs hinsichtlich der Erfüllung einer gegebenen Bedingung durch den Computer vertraut gemacht. Damit erwerben die Schüler die Einsicht, daß auch mathematisches Beweisen durch den Computer unterstützt werden kann.

Im **S t o f f a b s c h n i t t 2. 3.** sind die Schüler mit den Grafikmöglichkeiten der zur Verfügung stehenden Computer (Punkt- bzw. Symbolgrafik) vertraut zu machen. Sie lernen diese Kenntnisse beim Lösen von Aufgaben mit grafischen Darstellungen anzuwenden. Insbesondere sind Funktionen grafisch darzustellen, geeignete Koordinatensysteme zu zeichnen und Bewegungen sowie durch wechselseitige Lagebeziehungen von Punkten und Kurven in der Ebene entstehende Sachverhalte mit dem Computer zu veranschaulichen. Die Schüler sind zu befähigen, für Meßwertdarstellungen und statistische Auswertungen geeignete grafische Veranschaulichungen (z. B. Häufigkeitsverteilung als Linienzug, Histogramm oder Streckendiagramm) selbständig zu entwickeln und zu programmieren. Beim Aufgabenlösen ist auf die Verwendung von Feldern ebenso zu orientieren wie auf die Bearbeitung der einzelnen Teilaufgaben als in sich geschlossene Programmteile (Moduln), die auch in anderen Programmen verwendet werden können.

Gestützt auf das in den Stoffabschnitten 2.1. - 2.3. angeeignete Wissen und Können, sollen die Schüler im **S t o f f a b s c h n i t t 2. 4.** eine ihrem Leistungsvermögen angepaßte komplexe Aufgabe selbständig lösen. Dabei sollen sie zeigen, daß sie die Etappen des Aufgabenlöseprozesses schöpferisch auf die konkrete Aufgabenstellung anwenden können. Die durch die einzelnen Schüler zu lösenden Aufgaben sind so auszuwählen, daß jeweils sowohl numerische als auch nichtnumerische Inhalte (einschließlich Fragen der Bildschirmgestaltung) enthalten sind.

Das Lösen solcher komplexer Aufgaben kann für besonders leistungsfähige Schüler verbunden werden mit einer Vertiefung ihres Könnens im heuristischen Arbeiten mit dem Ziel, selbständig weitere Lösungswege zu finden, die für die Bearbeitung mit dem Computer geeignet sind.

2.1. Schritte beim Aufgabenlösen

(2 Stunden)

Verdeutlichen der Schrittfolge beim Aufgabenlösen mit dem Computer durch Verallgemeinerung der Bearbeitung von Beispielaufgaben:

- Analysieren der Aufgabe,
- Wählen eines geeigneten mathematischen Modells und
- Definieren entsprechender Daten,
- Algorithmmieren,
- Programmieren (einschließlich Testen und Korrigieren),
- Bearbeiten der Aufgabe mit dem Computer,
- Interpretieren der Resultate,
- Dokumentieren und Speichern des Programms.

2.2. Lösen numerischer Aufgaben

(6 Stunden)

Selbständiges Lösen einfacher numerischer Aufgaben (z. B. Aufstellen von Wertetabellen, Mittelwertberechnungen). Bekanntmachen mit elementaren Rekursions- und Iterationsverfahren durch selbständiges Bearbeiten von Aufgaben (z. B. Ermittlung des g.g.T nach Euklid, näherungsweise Berechnen von Nullstellen durch schrittweises Einschachteln über Wertetabellen, Untersuchung von Zahlenfolgen, Partialsummen, Partialsummenfolgen).

Verfahren der näherungsweisen Berechnung von \sqrt{x} und e ; allgemeines Iterationsverfahren (ohne Nachweis der Konvergenz).

Bekanntmachen mit dem Verfahren der vollständigen Durchmusterung einer vorgegebenen endlichen Datenmenge zum Lösen einer Aufgabe (z. B. beim Lösen entsprechender diophantischer Gleichungen). Einführen von Darstellung, Vereinbarung und Aufruf benutzerdefinierter Funktionen.

2.3. Lösen von Aufgaben mit grafischen Darstellungen (6 Stunden)

Verwendung bereitgestellter Grafikwerkzeuge (Anweisungen; Unterprogramme) zur Abbildung ebener Figuren im jeweiligen Bildschirmraster-system; Erarbeiten der Koordinatentransformation für Parallelverschiebung oder Streckung.

Weitgehend selbständige Entwicklung von Programmen für die Darstellung endlicher Datenmengen in Form von Säulen-, Balken-, Kreisdiagrammen bzw. Histogrammen.

Grafische Darstellung von Funktionen; prinzipielle Darstellungsmöglichkeiten und Konsequenzen (Auflösung der Grafik, Koordinatensystem, optimale Nutzung des Bildschirms).

Aufstellen von Menübildern, Nutzen der Unterprogrammtechnik.

2.4. Übung und Anwendung durch das Lösen einer komplexen Aufgabe (4 Stunden)

Übung im selbständigen Lösen einer komplexen Aufgabe in der vollständigen Schrittfolge vom Analysieren der Aufgabe bis hin zum Interpretieren des Resultats (z. B. Sortieren von Datenmengen; empirisches Ermitteln von Extremwerten; Darstellen von statistischen Angaben in geeigneten Diagrammen).

Kennenlernen verschiedener Möglichkeiten zur systematischen Lösungssuche; kombinatorische Variantenvielfalt und Lösungsbaum, Suche in der Tiefe und in der Breite, Abbruch und Rücksprung.

3. Arbeit mit vorgefertigter Software und Benutzersystemen

14 Stunden

Aufgabe dieses Stoffgebietes ist es, die Schüler an Beispielen mit wichtigen Grundprinzipien der Arbeit mit vorgefertigter Software vertraut zu machen, ihnen erste Einblicke in die Arbeit mit Benutzersystemen (z. B. zur Textbearbeitung, Dateiarbeit, Grafikerzeugung) zu vermitteln und sie zu befähigen, entsprechende Aufgaben durch Nutzen vorgefertigter Software zu lösen. Außerdem ist den Schülern ein Überblick über einige wesentliche Einsatzmöglichkeiten des Computers als programmierbarer digitaler Steuerungsautomat im Rahmen der Prozeßautomatisierung zu vermitteln.

Im Stoffabschnitt 3. 1. sollen die Schüler das Arbeiten mit vorgefertigter Software kennenlernen.

Dies kann geschehen, indem

- die Schüler an einem einfachen Beispiel Prinzipien der D a - t e i a r b e i t kennenlernen (Anlegen, Verwalten und Verarbeiten einer Datei und die Ausgabe von Daten). Durch Nutzen von Such- und Sortierprozessen in der angegebenen Datei machen sich die Schüler mit der Leistungsfähigkeit eines Datenverarbeitungssystems vertraut. Sie erkennen die Bedeutung solcher Prozesse für die Volkswirtschaft.
- ausgehend von einer einfachen Aufgabenstellung zur T e x t - b e a r b e i t u n g einige Leistungen eines Textbearbeitungsprogramms behandelt werden. Die Schüler erwerben dabei erste Fähigkeiten im Umgang mit Menüs, programmierten Hilfsseiten und schriftlichen Nutzeranleitungen. Dabei ist ihnen auch zu verdeutlichen, daß der Einsatz von Textverarbeitungssystemen auf die Entlastung von monotoner Routinearbeit gerichtet ist und eine Verkürzung der Herstellungszeiten von Schriftstücken bewirken kann.
- anhand eines konkreten (auf den verwendeten Computertyp zugeschnittenen) G r a f i k s y s t e m s die Schüler mit typischen Arbeitsweisen beim Nutzen von Grafiksystemen vertraut gemacht werden. Dabei sind volkswirtschaftlich bedeutsame Anwendungsfälle in die Darstellung einzubeziehen, wobei deutlich werden muß, daß mit vorgefertigter Software immer nur Teilziele und spezielle Leistungen erreicht werden können.

In allen Fällen geht es um die Vermittlung allgemeingültiger Kenntnisse, die zum Einstieg in die Arbeit mit einem beliebigen Softwaresystem unbedingt erforderlich sind. Die Ausprägung von Fertigkeiten in der Nutzung solcher Systeme ist nicht das Ziel dieses Stoffabschnittes.

Im Stoffabschnitt 3. 2. erwerben bzw. vertiefen die Schüler erste Einsichten über die Nutzung des Computers zur Prozeßautomatisierung. An geeigneten Beispielen erfahren sie, daß in direkter Prozeßkopplung

- analoge und digitale Werte physikalischer Größen über Meßfühler erfaßt und gegebenenfalls durch Wandler in digitale umgeformt vom Computer eingelesen, verarbeitet, gespeichert und dargestellt werden können (Prozeßdatenerfassung);
- vom Computer ausgegebene Daten in Form von digitalen oder (gewandelten) analogen Signalen zur Steuerung von Prozessen verwendet werden können (offene automatische Steuerung);
- mittels Computer aus dem Prozeß gewonnene Daten - mit vorgegebenen Daten (Sollwerten) verknüpft - zur Prozeßsteuerung verwendet werden können (geschlossene automatische Steuerung).

Der Unterricht ist so zu gestalten, daß alle Schüler ein Grundverständnis über die Nutzung des Computers als programmierbarer digitaler Steuerungsautomat erwerben bzw. vertiefen. Die für die computergeführte Prozeßautomatisierung erforderliche Hardware ist nur in einer einfachen Blockbilddarstellung zu vermitteln.

Die PIO-, CTC- und Wandlerprogrammierung wird nicht im Detail behandelt. Es wird lediglich auf die in der vorgefertigten Software enthaltenen Unterprogramme (Programmmodule), die diese Funktion erfüllen, verwiesen. Interessierte Schüler sind auf den fakultativen Lehrgang "Mikrorechentchnik in der sozialistischen Produktion" in Klasse 12 zu orientieren.

Bei der Behandlung sowohl der im Stoffabschnitt 3.1. ausgewählten Anwendungsbereiche als auch des Stoffabschnittes 3.2. ist bei den Schülern die Einsicht von der Rolle der Informatik und informationsverarbeitenden Technik als Schlüsseltechnologie für die weitere Ausgestaltung unserer sozialistischen Gesellschaft zu vertiefen. Insbesondere sind ihnen - ausgehend von den konkret zu lösenden Aufgaben - Wesen und Leistungsfähigkeit von CAD/CAM-Arbeitsplätzen weiter zu verdeutlichen.

3.1. Textbearbeitung/Dateiarbeit/Grafiksysteme (8 Stunden)

Dateiarbeit

Einführen wesentlicher Begriffe: Datenbank, Datei, Datensatz (als problemorientierte Datenstruktur), Datenfeld.

Nutzen eines (dem verwendeter Computertyp entsprechenden) Dateiprogramms, dabei insbesondere Anlegen, Verwalten und Verarbeiten einer Datei (Anfügen, Löschen von Datensätzen); Suchen und Sortieren in einer Datei.

Textbearbeitung

Nutzen eines (dem verwendeten Computertyp entsprechenden) Textbearbeitungssystems; Verwenden dieses Systems zur Eingabe und Änderung von Texten, dabei insbesondere Arbeiten mit dem zugehörigen Hauptmenü und den einzelnen Teilmenüs, Editieren (Einfügen bzw. Korrigieren von Text auf dem Bildschirm), Formatieren von Text auf dem Bildschirm, Speichern und Drucken von Texten.

Übungen zum Schreiben, Korrigieren, Gestalten, Speichern, Ausgeben und Vervielfältigen eines Textes (z. B. Schreiben einer Studienbewerbung).

Grafiksysteme

Nutzen eines (dem verwendeten Computertyp entsprechenden) Grafiksystems zum Entwurf und zur Modifikation grafischer Darstellungen. Dabei Verwenden verschiedener Formen der Steuerung solcher Systeme (z. B. Menüsteuerung, parametrisierte Kommandos, interaktive Grafikeingabe über die zur Verfügung stehenden Eingabemöglichkeiten). Verschiedene Möglichkeiten zur Erzeugung einer Grafik (koordinatengebunden oder koordinatenfrei), deren Vor- und Nachteile in unterschiedlichen Aufgabenstellungen. Hinweisen auf effiziente Formen der Bildspeicherung auch mittels spezieller Medien (Bildplatten).

3.2. Prozeßautomatisierung

(6 Stunden)

Prozeßdatenerfassung, -auswertung und -darstellung; modularartige Konfiguration der Hard- und vorgefertigten Software.

Darstellung von Meßwerten (z. B. bei Auf- und Entladung eines Kondensators, bei optoelektronischer Drehzahlmessung, bei Zeitmessung mit Rangfolgeordnung);

Offene automatische Steuerung eines Prozesses nach vorgegebenen Daten (z. B. Steuerung einer Lichterkette);

Geschlossene automatische Steuerung eines Prozesses (z. B. programmierbare Temperaturregelung, Waschmaschinenprogramm).

Bei noch nicht erfolgter Geräteausstattung werden die Prozesse simuliert.