

Praktikumsbericht - Blockpraktikum (SPS V) - Fach Informatik

Vorgelegt von:
Matrikelnummer:
Semester:

Schule:	Berufliches Schulzentrum 1 der Stadt Leipzig Wirtschaft und Verwaltung	
Anschrift:	Crednerstraße 1, 04289 Leipzig	
Telefonnummer:	0341 48 47 921	
Zeitraum (von - bis)	31 . 08 . 2020	01 . 10 . 2020
Mentor:	Herr C. Hinz	
Vertreter der Fachdidaktik:		

Universität Leipzig
Fakultät für Mathematik und Informatik
Professur für Didaktik der Informatik



30.10.20

Inhaltsverzeichnis

1. Analysen	1
1.1. Kurzinformation über die Praktikumsschule	1
1.2. Analyse der Klasse in der Prüfungsstunde	2
2. Planungsentwurf einer Unterrichtseinheit	4
3. Planungsentwurf einer Unterrichtsstunde	6
3.1. Vorbereitung	6
Verlaufsplanung	10
3.2. Auswertung und Reflexion	12
4. Besondere Aktivitäten außerhalb des eigenen Unterrichts	14
5. Wertung der Erfahrungen	15
6. Anhang	16
6.1. Unterrichtsentwürfe	16
6.2. Unterrichtsmittel	18
6.3. Hospitations-Protokolle	33
7. Literatur- und Quellenverzeichnis	38
Eigenständigkeitserklärung	39

1. Analysen

1.1. Kurzinformation über die Praktikumsschule

Das Berufsschulzentrum 1 der Stadt Leipzig Wirtschaft und Verwaltung (im Folgenden kurz BSZ1 genannt) ist das größte Berufsschulzentrums Sachsen und beherbergt jährlich über 1000 Schülerinnen und Schüler. Die Schule hat dabei zwei unterschiedliche Schulteile, den Schulteil Crednerstraße, an welchem das Praktikum absolviert wurde und welcher der nachfolgenden Arbeit zugrunde liegt und den Schulteil Dachsstraße. Da das BSZ1 Berufsschulunterricht für verschiedenste Berufsgruppen anbietet, wurden die Berufsgruppen auf die Schulteile verteilt. (Berufliches Schulzentrum 1 der Stadt Leipzig, 2020)

Interessant für die nachfolgende Arbeit ist vor allem die hohe Diversität der Lernenden am Schulteil Crednerstraße, in welchem hospitiert und auch unterrichtet wurde. An diesem Schulteil wird Unterricht für das Berufliche Gymnasium, die Fachoberschule und die Berufsschule erteilt. Alle drei Schulformen sind sehr unterschiedlich angelegt. Die Berufsschüler*innen werden nach Lernfeldern unterrichtet und haben keinen herkömmlichen Informatikunterricht. In den entsprechenden Lernfeldern wird eine informatische Sicht auf die jeweiligen Themen behandelt. Die Lernenden in dieser Schulform sind außerdem nur ein Drittel des Schuljahres an der Schule und lernen die anderen zwei Drittel des Jahres in ihren Ausbildungsbetrieben.

Die Fachoberschüler*innen benötigen für den Eintritt in die Fachoberschule (im Folgenden kurz FOS genannt) einen Realschulabschluss. Mit diesem können sie die zweijährige FOS absolvieren, welche die Schüler*innen mit dem Erreichen der Fachhochschulreife zum Studium an einer Fachhochschule qualifiziert. Schüler*innen, die zusätzlich zum Realschulabschluss einen Berufsabschluss besitzen, können die FOS in einem einzigen Jahr absolvieren. Da diese Möglichkeit jedoch verhältnismäßig selten genutzt wird, werden diese Schüler*innen am BSZ1 in die bestehenden zwölfsten Klassen der FOS integriert und nicht, wie an anderen Berufsschulzentren vorzufinden, in eigenen Klassen zusammengefasst

Als dritte Schulform gibt es noch das Berufliche Gymnasium. Hier kommt der Unterricht dem Informatik-Unterricht an einem normalen Gymnasium am nächsten, wobei die Schüler*innen am Beruflichen Gymnasium mit einem Realschulabschluss innerhalb von drei Jahren ein vollwertiges Abitur und damit die allgemeine Hochschulreife erlangen. In der Klassenstufe 11 wird dabei im Klassensystem unterrichtet, bevor die Klassenstufen zwölf und dreizehn dann in einem normalen Kurssystem durchlaufen werden, wie es auch in der Sekundarstufe II an sächsischen Gymnasien der Fall ist.

Wie bereits angesprochen handelt es sich beim BSZ1 um das größte Berufsschulzentrum Sachsens, wobei zu beachten ist, dass im Laufe des Schuljahres nur selten alle Schüler*innen in der Schule sind, da die Auszubildenden (im Folgenden kurz Azubis genannt) aus der Berufsschule nur geringe Teile des Jahres an der Schule sind. Der Schulteil Crednerstraße besitzt ein E-förmiges Gebäude, welches im mittleren Anbau eine Turnhalle beinhaltet. Das Gebäude besitzt in den äußeren Gebäudeteilen drei

Vollgeschosse, während im Verbindungsteil neben der Mensa vor allem Verwaltungszimmer untergebracht sind.

Fachlich ist an der Schule die Ausrichtung auf Wirtschaft und Verwaltung wesentlich, was man vor allem auch dem hospitierten Unterricht insofern anmerkte, als dass die gewählten Beispiele der Lehrkräfte häufig einen wirtschaftlichen Hintergrund besaßen.

Der Unterricht ist in Blockstunden gegliedert, wobei der erste Block morgens 08:00 Uhr beginnt und der fünfte Block 16:55 Uhr endet. Die längste Pause stellt die Mittagspause von 11:20 Uhr bis 12:00 Uhr dar. Da die Schüler*innen in der Regel volljährig sind ist es ihnen gestattet, das Schulgelände in den Pausen zu verlassen.

Das BSZ1 liegt nahe der Bahnhaltestelle Probstheida in der Nähe der Pragerstraße und ist damit gut durch den ÖPNV zu erreichen, besitzt aber auch einen großen Schülerparkplatz für die Schüler*innen, die mit dem Auto zur Schule kommen.

Die Schule verfügt über 4 Informatikkabinette, welche mit jeweils 16 Schüler*innen- Rechnern ausgestattet sind und einen Lehrercomputer beinhalten. Die Räume besitzen neben einem Höhenverstellbaren Whiteboard auch einen Beamer, welcher mit dem Lehrercomputer verbunden ist und in der Mitte eine Tischinsel, welche die Arbeit abseits der Computer erleichtert. Die Schüler*innen erhalten für den Unterricht einen Login zu Schuljahresbeginn, mit welchem sie sich auf dem Schulserver anmelden können. Dort können sie ihre Dateien abspeichern. Auf den Computern ist keine pädagogische Software vorhanden. Die Rechner selber werden jedoch von einem HD-Guard überwacht, welcher bestimmte Rechte beschränkt und die Rechner selber bei jedem Neustart auf einen definierten Zustand zurücksetzt. Damit geht einher, dass Dateien, welche Schüler*innen versehentlich auf der lokalen Festplatte statt auf dem Server speichern, beim Herunterfahren des Rechners verloren gehen. Zum Austauschen von Dateien zwischen Schüler*innen und Lehrer*innen steht ein Datenaustauschlaufwerk zur Verfügung, welches nach Klassenstufen und Kursen getrennt ist. Die Lehrkraft hat dabei Zugriff auf alle Ordner, während die Schüler*innen nur den zu ihrer Klasse gehörenden Ordner sehen können.

1.2. Analyse der Klasse in der Prüfungsstunde

Die im nachfolgenden ausgeführte Stunde wurde im Kurs inf2-13 gehalten. Es handelt sich dabei um einen von zwei bestehenden Informatik-Grundkursen in der Klassenstufe 13 des beruflichen Gymnasiums. Parallel zu diesem Kurs liegt der Kurs inf1-13, weshalb eine grobe inhaltliche Abstimmung zwischen beiden unterrichtenden Lehrpersonen besteht und eingehalten werden sollte.

Der Kurs wird von 16 Schüler*innen belegt und bietet daher im Fall einer 100 prozentigen Anwesenheit keine Ausweichmöglichkeit mehr, falls ein Rechner nicht funktionieren sollte. Die Lerngruppe besteht aus 10 Schülerinnen und 6 Schülern. Da Informatik in der Jahrgangsstufe 12 verpflichtend, in der Stufe 13 jedoch Wahlfach, ist, wurden die Kurse neu gemischt. Die Schüler*innen kennen sich jedoch größtenteils durch die Wechsel im

Kurssystem bereits seit dem vergangenen Schuljahr. Zu Jahresbeginn herrschte freie Sitzplatzwahl, weshalb sich bereits an der Sitzordnung Sympathiegruppen ablesen lassen.

Allgemein ist das Verhalten der Lerngruppe sehr sozial und dem Alter angemessen. Die Schüler*innen und Schüler verhalten sich im Unterricht ruhig, sind respektvoll sowohl untereinander als auch gegenüber der Lehrperson und arbeiten konzentriert mit. Von den 16 Schüler*innen haben sich neun für eine mündliche Abiturprüfung im Fach Informatik entschieden, weshalb die Lernmotivation bei den meisten Schüler*innen sehr hoch ist, da ihnen bewusst ist, dass der gesamte vermittelte Stoff relevant für die Prüfung sein könnte.

Prinzipiell ist ein Grundniveau von informatischem Verständnis und Interesse gegeben. Dies begründet sich zum einen dadurch, dass die Schüler*innen alle freiwillig den Informatikunterricht in Klasse 13 besuchen, zugleich haben sie seit mindestens zwei Jahren geregelten Informatikunterricht am BSZ1. Zum aktuellen Stoffgebiet der Algorithmierung und Programmierung unterscheidet sich das Vorwissen dennoch sehr stark. Einige Schüler*innen haben an der vorhergehenden Schule erste Einblicke beispielsweise mittels Kara erhalten und eine Schülerin hat auf freiwilliger Basis in den Ferien einen 2-wöchigen Python-Kurs belegt. Andere Schüler*innen haben hingegen noch nie programmiert. Folglich ist ein großer Leistungsunterschied in der Lerngruppe vorhanden, den es zu berücksichtigen gilt. Außerdem hat sich bei näherer Betrachtung im Verlaufe der Hospitation gezeigt, dass einige Schüler*innen sich nicht aktiv bemerkbar machen, wenn sie Probleme haben, erst auf Nachfrage kommunizieren sie diese. Auch das ist bei der Planung des Unterrichts und der Umsetzung zu beachten, ein beständiges Einholen von Bestätigung kann hier vorbeugend hilfreich sein.

Die Kompetenzen der Lerngruppe in Bezug auf unterschiedliche Sozialformen kann nur oberflächlich beurteilt werden, da der Kurs in dieser speziellen Zusammensetzung nur ein einziges Mal hospitiert werden konnte. In der Hospitationsstunde wurde ausschließlich frontal gearbeitet, da es sich um die Einführungsstunde im neuen Schuljahr handelte. Vereinzelt schienen Schüler*innen dabei Probleme zu haben, dem Tempo des Unterrichts zu folgen. Sie holten sich jedoch selbstständig Hilfe bei ihren Klassenkamerad*innen. Damit kann pauschal keine Aussage über Kompetenzen vor allem bezüglich Partner- und Gruppenarbeit getroffen werden, da die Schüler*innen jedoch hilfsbereit und freundlich auftraten und sich auch in den Pausen entsprechend verhielten ist davon auszugehen, dass sie, auch auf Grund ihrer Reife und Erfahrung, gut mit diesen Sozialformen umgehen können.

2. Planungsentwurf einer Unterrichtseinheit

Es soll im Folgenden die Planung der Unterrichtseinheit zu „Lernbereich 3: Algorithmen und Programme“ (Sächsisches Staatsministerium für Kultus, 2020, S.22) erläutert werden. Für diesen Lernbereich stehen am beruflichen Gymnasium laut Lehrplan in der Klassenstufe 12 oder 13 insgesamt 24 Unterrichtsstunden zur Verfügung. Vorausgesetzt wird bei der Behandlung dieses Lernbereichs lediglich Wissen über einfache Datentypen, welche den Schüler*innen bereits in „Lernbereich 2: Modellierung von Datenbanken“ (ebd., S. 21) bei der Arbeit mit MS Access erläutert wurde und nochmal aufgefrischt wird. Darüber hinaus müssen die Schüler*innen in der Lage sein, sich allgemein sicher in Computerprogrammen zurechtzufinden (Drag-and-Drop-Prinzip, Öffnen und Speichern von Dateien etc.). Dieser letzte Punkt ist selbstverständlich nach mindestens zwei Jahren Informatik-Unterricht am beruflichen Gymnasium gegeben. Da im Informatikunterricht der Oberschule Programmierung nicht vorkommt, kann hierzu kein weiterer Stoff vorausgesetzt werden. Es wurde lediglich das Prinzip „Klassen und Objekte“ laut Lehrplan behandelt, da aber am BSZ1 imperativ programmiert wird, wenngleich Delphi als objektorientierte Sprache angelegt ist, ist dieses Wissen nicht zwingend erforderlich. Dennoch kann ein Verständnis über Klassen, Objekte und Attribute natürlich im Verlauf dieser Unterrichtseinheit helfen, einzelne Code-Bestandteile besser zu verstehen.

Block (je 2 Stunden)	Datum	Inhalte	Organisationsform
1	01.09.	Einführungsstunde, Motivation, Definition Algorithmus	Frontalunterricht
2	08.09.	Grundlagen Algorithmierung	Gruppenarbeit
3	15.09.	Übungen Algorithmierung Überblick Einordnung der Programmiersprachen	Einzelarbeit, Schülervorträge
4	22.09.	PAP-Designer als Case-Tool, Einführung Formulardesign	Frontalunterricht
5	29.09.	Einführung Programmierung	Think-Pair-Share, Stationsarbeit
6	06.10.	LK Programmablaufplan/Formulardesign	Einzelarbeit
7	13.10.	Übungsstunde Verzweigung	Partnerarbeit
8	03.11.	Einführung Schleifen, Schleifenarten	Gruppenpuzzle
9	10.11.	Übung Algorithmierung, Grundbegriffe Programmiersprachen	Frontalunterricht

10	17.11.	Modularisierung, Programmierung	Übung	Stationsarbeit
11	24.11.	Grenzen der Algorithmierbarkeit		Schülervorträge
12	01.12.	Klausur Algorithmen und Programme		Einzelarbeit

3. Planungsentwurf einer Unterrichtsstunde

3.1. Vorbereitung

Einordnung in Lehrplan/Bildungsstandards Sek. II:

Die im folgenden betrachtete Unterrichtsstunde behandelt schwerpunktmäßig die Einführung der Programmierung. Die Schüler*innen haben sich im vorherigen Teil des Schuljahres bereits mit wesentlichen Inhalten des Lehrplans befasst. Vor allem wurde dabei auf die Grundstrukturen Sequenz und Entscheidung eingegangen und deren Visualisierung sowie das Problemlösen mit Hilfe des Programmablaufplans geübt. Außerdem fordert der Lehrplan die „Anwend[ung] von Komponenten der objektorientierten Programmierung“, dem wurde bereits in den vorhergehenden Stunden insofern Rechnung getragen, als dass ein Formular designed wurde. Dabei wurde auf die Klassen von Objekten sowie deren Eigenschaften eingegangen. Da die Problemlösestrategien in einer höheren Programmiersprache beherrscht werden sollen, soll nun in der betrachteten Stunde das vorhandene Wissen von Programmablaufplänen auf die Programmierung mit Delphi übertragen werden, womit ein Grundstein für diesen wesentlichen Teil des Lernbereichs 3 gelegt wird. (Sächsisches Staatsministerium für Kultus, 2020, S.22)

Bei den Prozessbereichen der Bildungsstandards sind vor allem zwei Bereiche besonders relevant für die Thematik des Programmierens. Sowohl Prozessbereich 1, das „Modellieren und Implementieren“, als auch der Bereich 5, das „Darstellen und Interpretieren“, sind besonders geeignet, um mit der Programmierung verknüpft zu werden. Das Implementieren einer Lösung stellt gerade den Wesenskern dieses Themenbereichs dar, während die Modellierung für die Problemlösung unerlässlich ist. Ebenso wichtig ist schließlich die Darstellung der Lösung. In der Praxis ist zudem relevant, andere Lösungen interpretieren zu können, weshalb auch dies bereits in der Schule geübt werden sollte. Dennoch sind natürlich auch die anderen drei Prozessbereiche nicht zu vernachlässigen und passen mit gewissen Schnittstellen gut zur Thematik der Programmierung.

Bei den Inhaltsbereichen behandelt der Lernbereich laut Lehrplan vor allem Themen, die in die Bereiche „Algorithmen“ sowie „Informatiksysteme“ einzuordnen sind. Die Einordnung in den Bereich Algorithmen liegt mit der Behandlung von Grundstrukturen und Problemlöseverfahren auf der Hand. Aber auch Informatiksysteme, vor allem die Klassen-Objekt-Hierarchie, spielen bei der Behandlung der Programmierung eine Rolle, wobei hier vor allem die Software-Seite des Informatiksystems betrachtet wird. (Gesellschaft für Informatik e.V., 2016)

Am BSZ1 wird die Programmierung im RAD Studio von embarcadero unterrichtet. Die Software unterstützt sowohl die Programmierung mit Delphi als auch mit C++ in der gleichen Entwicklungsumgebung (IDE). Es steht für das Programm eine Schullizenz zur Verfügung. Programmiert wird jedoch nur in Delphi, weshalb nur darauf im Weiteren eingegangen werden soll.

Die Programmierung ist nur ein kleiner Aspekt der Softwareentwicklung. Üblicherweise müssen in der Praxis zunächst die Anforderungen an eine Problemlösung sowie das genaue Problem aufgenommen werden. Anschließend kann ein Konzept zur Umsetzung erarbeitet werden. Erst wenn dies geschehen ist kommt es schließlich zur tatsächlichen Programmierung, bevor schließlich eine Testphase die Entwicklung abschließt. (vgl. Meyer, 2018, S.6).

Zur Planung einer Programmlösung sind verschiedene Methoden verbreitet. Eine häufig genutzte Möglichkeit ist das Nassi-Shneiderman-Diagramm (auch Struktogramm genannt),

welches in der DIN 66261 genauer spezifiziert wird. (IGPM, 2015) In einem Struktogramm werden, wie der Name bereits sagt, vorrangig die Programmstrukturen erfasst. Dabei gibt es durch die Norm eine genaue Festlegung, welche Struktur symbolisch wie darzustellen ist. So werden Verarbeitungen durch Rechtecke dargestellt und Sequenzen durch die Aneinanderreihung dieser Rechtecke. Bei Alternativen werden die möglichen Anweisungen unter dem Entscheidungsblock durch nebeneinander platzierte Anweisungen symbolisiert. Da viel mit rechteckigen Strukturen gearbeitet wird, welche sich Kanten teilen und dadurch direkt ineinander übergehen, ist ein Struktogramm jedoch häufig für Anfänger schwer zu lesen.

Eine weitere Möglichkeit zur Visualisierung einer Problemlösung stellt der Programmablaufplan dar. Auch sein Aussehen ist in einer Norm, der DIN 66001 festgehalten. Der Programmablaufplan sticht dabei vor allem durch verschiedene Formen heraus. So ist eine allgemeine Verarbeitung weiterhin als Rechteck dargestellt, aber bereits Ein- und Ausgabe werden durch ein Parallelogramm davon abgehoben. Auch die Verzweigung hat mit der Raute ein eigenes Symbol. Die einzelnen Bestandteile werden schließlich durch Linien miteinander verbunden. (Institut für elektronische Datenverarbeitung Zürich (Hrsg.), 1975)

Ist die theoretische Betrachtung der Problemlösung abgeschlossen, kann der fertige Programmablaufplan genutzt werden, um den Quellcode zu erzeugen.

Ein einacher Quellcode lässt sich dabei nach dem EVA-Prinzip zergliedern. EVA steht dabei für die drei Teilschritte „Eingabe“, „Verarbeitung“ sowie „Ausgabe“. Es werden also zunächst alle benötigten Daten zur Problemlösung vom Nutzer erfragt oder anderweitig eingelesen, anschließend werden alle notwendigen Verarbeitungsschritte auf den Daten ausgeführt und schließlich wird die Lösung ausgegeben. Die eigentliche Programmierung erfolgt in einer beliebigen Programmiersprache. Weit verbreitet an Schulen ist die Programmierung mit Delphi. Delphi ist dabei jedoch, anders als häufig falsch angenommen, nicht der Name der Programmiersprache, sondern der Name der von Borland entwickelten Entwicklungsumgebung, in welcher mit Object Pascal programmiert wird. Diese Sprache ist eine Weiterentwicklung der ursprünglich als Lehrsprache entwickelten Sprache Pascal welche sich vor allem dadurch auszeichnet, dass sie das objektorientierte Programmierparadigma unterstützt. (Rauch, Augsten, 2019) Damit gehört die Sprache zu den imperativen Programmiersprachen, bei welchen der Weg zur Lösung eines Problems beschrieben wird. Dem entgegen würden die deklarativen Sprachen wie beispielsweise Common LISP stehen, welche die Ausgangslage und die gesuchte Lösung, nicht aber den Lösungsweg bei der Programmierung fokussieren.

Delphi nimmt dem Programmierer beim Anlegen einer Formularanwendung bereits viel Arbeit ab, da im Hintergrund bereits ein funktionierendes Klassengerüst erstellt wird. Der Nutzer muss nun nur noch eigene Prozeduren anlegen und seine Variablen deklarieren, bevor die Programmierung der Prozeduren auch schon beginnen kann. Das Oberflächendesign einer Formularanwendung ist mit Delphi ebenfalls sehr einfach, da eine Oberfläche zur Verfügung gestellt wird, in welcher mittels Drag-and-Drop das gewünschte Layout des Fensters mit der Maus „zusammengeklickt“ werden kann.

Für die Vermittlung in der Schule ist vor allem relevant, die grundlegenden Prinzipien dieses dargestellten Konstrukts der Softwareentwicklung zu verstehen. Daher sollten die Schüler*innen eine Form der Visualisierung eines Algorithmus beherrschen. In diesem Fall wurde dazu vom Lehrer der Programmablaufplan ausgewählt. Außerdem sollten sie einfache Programme, die nach dem EVA-Prinzip verlaufen, erstellen können.

Aus lernpsychologischer Sicht ist es sehr förderlich für die Motivation der Lernenden, wenn sie ein neues Thema mit einem Erfolgserlebnis beginnen können. In den vorausgegangen

Stunden haben die Schüler*innen das Algorithmieren mittels Programmablaufplan bereits ausführlich geübt. Ebenso wurde mit Delphi bereits eine Oberfläche für eine Formularanwendung erstellt. Der inhaltliche Fokus liegt also darauf, nun den Transfer zwischen Programmablaufplan, fertigem Formular und Quellcode herzustellen, wobei die gesamte Syntax des Quellcodes noch zu vermitteln ist.

Um ein Erfolgserlebnis möglichst wahrscheinlich zu machen, wurde für die Stunde ein sehr einfaches Programm gewählt, welches zu zwei gegebenen Seitenlängen eines Rechtecks den Oberflächeninhalt und den Umfang berechnet. Dafür wurde bereits der Programmablaufplan als Übung erstellt und es wurde auch der Aufbau des Formulars bereits besprochen und umgesetzt. Da die Schüler*innen jedoch bisher noch keinen Quellcode gesehen haben, ist hier sehr viel Stoff auf einmal zu vermitteln. Dafür soll eine Methode des entdeckenden Lernens gewählt werden. Die Schüler*innen werden dazu mit einem fertigen Code zu einem ähnlichen Programm konfrontiert. Da die Befehle in Delphi dem englischen Wortschatz entstammen und die Arithmetik auf der klassischen Mathematik beruht, sollten die Schüler*innen in der Lage sein, sich große Teile des Codes, basierend auf ihrem Vorwissen über Programmablaufpläne, Wertebereiche von Variablen und den Zuweisungsoperator „:=“, welcher bereits bei Programmablaufplänen behandelt wurde, selbst zu erschließen. An diesem Code können dann die wichtigsten Merkmale besprochen werden und es kann eine Adaption auf das eigene Programm erfolgen. Zeitgleich wird so ermöglicht, dass nicht bereits in der ersten Stunde alle Teile einer Prozedur vollständig verstanden werden müssen, um lauffähigen Code zu erzeugen. Da die Quelltextinterpretation sehr anspruchsvoll ist, soll hier nicht ausschließlich Einzelarbeit erfolgen. Um die tatsächliche Lernzeit, die nach Meyer (2011) ein Merkmal guten Unterrichts ist, dennoch möglichst hoch zu halten, soll hier auf die Methodik des Think-Pair-Share zurückgegriffen werden. So können sich die Schüler*innen zunächst selbstständig und anschließend in Partnerarbeit Gedanken machen, bevor im Schüler-Lehrer-Gespräch nur noch ein/e Schüler*in zeitgleich zu Wort kommen kann. Ist der Programmcode schließlich angepasst, sollte eine weniger belastende Phase folgen, da es für die Schüler*innen unmöglich ist, 90 Minuten hochkonzentriert mit maximalem Erfolg zu arbeiten. Daher kann nun anhand einfacher Werte die Funktionalität des Programms getestet werden und zugleich lernen die Schüler*innen implizit ein weiteres wesentliches Element der Softwareentwicklung kennen.

Anschließend soll dann noch einmal ausführlichen zu allen bisher behandelten Themen, Schwerpunktmäßig dabei dem Programmablaufplan, geübt werden, um einen anstehenden Test vorzubereiten. Um dabei das größtmögliche Maß an Differenzierung zu ermöglichen, soll eine digitale Stationsarbeit durchgeführt werden. Passend zur Corona-Pandemie bleiben die Schüler*innen dabei an ihrem Platz und bearbeiten die virtuellen Stationen in beliebiger Reihenfolge. Dabei gibt es einen Pflicht-Teil der Stationen, welcher ausreichend auf den Test vorbereiten soll, darüberhinausgehende Stationen wurden unter Wahlpflicht eingeordnet. Zur Orientierung erhalten die Schüler*innen einen Laufzettel. Um die Einheit motivierend abzuschließen, soll mittels Kahoot in einem Quiz noch einmal ein breites Spektrum an Fragen abgedeckt werden. Die Schüler*innen erhalten hier die Möglichkeit, sich mit anderen zu vergleichen und können sehen, welche Themenbereiche ihnen noch Probleme bereiten. So können die letzten Minuten genutzt werden, um Fragen für den Test zu besprechen.

Damit lassen sich für die Stunde die folgenden Lernziele ableiten:

- Die SuS interpretieren einfache Quelltextzeilen korrekt und zeigen dies, indem sie vorgegebenen Quelltext auf ihr eigenes Projekt anpassen.

- Die SuS vergleichen Quelltext und Programmablaufplan und können mindestens drei Gemeinsamkeiten benennen.
- Die SuS benennen sowohl die einzelnen Fenster der Programmoberfläche „RAD-Studio“ als auch ausgewählte Komponenten auf dem Formular korrekt und geben ihren Zweck korrekt wieder.
- Die SuS entwickeln selbstständig zu einem gegebenen Problem einen entsprechenden Programmablaufplan.

Verlaufsplanung

Tabellarische Auflistung der Unterrichtsschritte in der geplanten Reihenfolge mit den verbindlichen Angaben (siehe Tabellenkopf)

Uhrzeit	Unterrichtsschritt	Lehrer- / Schülerhandlung	Methode	Sozialform ¹	Lernmittel, Bemerkungen
- 08:00		SuS starten bereits vor Unterrichtsbeginn ihre Rechner		EA	Computer
08:00 – 08:03	Begrüßung/ Motivation	L benennt Stundenziel (Programmierung abschließen, PAP für Test wiederholen) S nennt zuletzt designtes Programm, L zeigt fertige Musterlösung	S.-L.-Gespräch	FA	Musterlösung Rechteck-Programm
08:03-08:05	Wiederholung	SuS sollten fertige Oberfläche haben, wenn nicht: Musteroberfläche kopieren Alle anderen: PAP zu Rechteckprogramm öffnen, still eindenken	EA	EA	Leere Oberfläche für SuS die nicht fertig sind
08:05 – 08:10	Erarbeitung	PAP wird an Beamer besprochen, SuS benennen EVA-Phasen, Bezug zwischen PAP und Formular wird von SuS mdl- hergestellt	Angeleitetes Lehrgespräch	FA	PPP
08:10 – 08:20	Erarbeitung	Code für Berechnung eines Dreiecks wird erst von SuS still analysiert (2 min), dann 2 min in Partnerarbeit besprochen, anschließend an Beamer durchgegangen und für Rechteck angepasst, SuS machen Notizen auf AB	Think-Pair-Share	EA/ PA/ FA	Programmcode für Dreieck, AB

1

vgl. Schubert/Schwill: FA...Frontalarbeit, GA...Gruppenarbeit, PA...Partnerarbeit, EA...Einzelarbeit

08:20 – 08:35	Erarbeitung	SuS sollen erarbeiteten Code auf eigenes Program anpassen, Probleme ggf. mit Nachbarn besprechen	Arbeit mit interakt. Medium	EA/ PA	PPP mit Aufgabenstellung
08:35 – 08:40	Erarbeitung	Anhand des Programm-Lifecycle (auf AB von Jahresbeginn) wird im Gespräch erarbeitet, dass nach Programmierung Test folgt	Gelenktes Lehrgespräch	FA	AB von Jahresbeginn
08:40 – 08:45	Übung	SuS testen eigenes Programm anhand gegebener und eigener Werte, erstellen Wertetabelle		EA	Wertetabelle in PPP
08:45 – 08:50	Motivation	Hinweis auf und durchgehen der Zusammenfassungen der letzten Stunde auf Lernsax, Motivation einer Stationsarbeit und Erklärung der Funktionsweise	L.-S.-Gespräch	FA	Laufzettel
08:50 – 09:10	Übung	SuS üben an verschiedenen Aufgaben Pflicht/Wahlpflichtteil	Stationsarbeit	EA	Stationsmaterial (digital)
09:10 – 09:15	Sicherung	Ergebnisse werden selbstständig verglichen, Resultate in Word zusammengefasst SuS sollen Dokument mit Resultaten ausdrucken und sich nebenbei bei Kahoot anmelden		EA	
09:15 – 09:20	Übung	Quiz mit Kahoot, Ziel: Fragen aufdecken	Arbeit mit interaktivem Medium	EA	Kahoot-Quiz
09:20 – 09:28	Sicherung	Zeit für Fragen der SuS, sonst Fragen der Lehrkraft	L.-S.-Gespräch	FA	Vorbereitete Verständnisfragen
09:28 – 09:30	Desinfektion	Während SuS Arbeitsplätze desinfizieren Ausblick auf nächste Stunde geben			

3.2. Auswertung und Reflexion

Die Stunde verlief deutlich abweichend vom angegebenen Plan. Bereits die Motivationsphase zum Anfang sowie das Eindenken in den Programmablaufplan benötigten beinahe 15 Minuten, weshalb hier schon zu Beginn ein deutlicher zeitlicher Rückstand aufgebaut wurde. Dies ist jedoch allein damit zu begründen, dass zur entsprechenden Stunde ein Warnstreik der LVB stattfand. Zu Unterrichtsbeginn waren daher von 16 erwarteten Schüler*innen nur 7 anwesend. Innerhalb der ersten 10 Minuten nach Beginn erschienen die weiteren 9 Schüler*innen. Daher wurden dann Fragen geklärt und die Schüler*innen wurden mit dem Eindenken beauftragt, da jedoch beinahe im Minutentakt weitere Schüler*innen ankamen wurde dann gewartet, bis sich alle eingerichtet hatten. Von den anwesenden Schüler*innen wurde dies sehr gut toleriert und mit viel Verständnis aufgenommen. Alternativ wäre es möglich gewesen, die Stationsarbeit vorzuziehen, so dass die Schüler*innen differenziert arbeiten können. Die Programmierung stand jedoch am Stundenanfang, um Zeitprobleme ausgleichen zu können, weshalb diese Reihenfolge bewusst nicht getauscht wurde.

Anschließend dauerte die Besprechung des Quelltextes fast 25 Minuten und damit deutlich länger, als kalkuliert. Da durch das sehr konkrete Nachfragen einzelner Schüler*innen jedoch immer mehr Verständnis auch bei den Mitschüler*innen erreicht wurde, wurde die Zeit hier gut investiert. Dadurch war die gesamte Programmierung jedoch erst nach einer sehr reichlichen Stunde abgeschlossen. Es wurde deshalb auf das Testen des Programms als allgemeiner Unterrichtsbestandteil verzichtet und die schnelleren Schüler*innen konnten dies vorarbeiten, so dass auch für sie kein Leerlauf entstand.

Da nun nur noch ca. 20 Minuten des Unterrichts verblieben war es unmöglich, noch sowohl Kahoot als auch die Stationsarbeit im Unterricht unterzubringen. Da das Stationsmaterial vollständig digital vorlag, wurde entschieden, noch das Quiz durchzuführen, um Fragen aufzudecken, und den Schüler*innen das Stationsmaterial auf Lernsax hochzuladen, um damit selbstständig zuhause zu üben. Das Quiz kam bei den Schüler*innen sehr gut an.

Der Stundenschluss war leider etwas chaotisch, da die Schüler*innen abgelenkt waren und keine wirklichen offenen Fragen gestellt wurden. Hier sollte noch einmal mehr darauf geachtet werden, qualitativ hochwertige Fragen vorzubereiten und auch eine Zusammenfassung der Stunde durchzuführen.

Der Schluss war auch der einzige beanstandete Punkt durch den Mentor und den Vertreter der Didaktik, neben der Zeitplanung. Somit ist also neben dem eben gesagten für die Überarbeitung dieser Stunde vor allem zu nennen, dass man sich von vornherein mehr Zeit nehmen sollte, um den Quellcode zu besprechen. Es kann dann im Unterricht noch eine erste Station angefangen werden und ansonsten kann die Stationsarbeit von vornherein darauf ausgelegt werden, den Schüler*innen zur individuellen Vorbereitung zuhause zur Verfügung zu stehen.

Für die generelle weitere Lehrtätigkeit habe ich aus der Stunde vor allem Positives mitgenommen. Ich hatte während des gesamten Praktikums Spaß am Unterrichten und mir wurde widerspiegelt, dass ich das sehr gut transportieren könne. Arbeiten muss ich

vor allem noch an meinem Stundenabschluss und an meiner Zeitplanung, die oft sehr knapp ausfällt.

4. Besondere Aktivitäten außerhalb des eigenen Unterrichts

Ich habe gemeinsam mit meinem Mentor den Aufbau des Verwaltungsnetzes sowie der Notendatenbank besprochen und die Eingabesoftware ausführlicher besprochen. Das war ein sehr interessanter Einblick, da hier bei einer so großen Schule wie dem BSZ1 ein enormer Aufwand betrieben werden muss, welcher von meinem Mentor administriert wurde.

5. Wertung der Erfahrungen

In diesem Praktikum konnte ich noch einmal viele Erfahrungen sammeln, die mir auch trotz des bereits absolvierten Blockpraktikums in Mathe neu waren. Besonders bedeutsam war für mich dabei vor allem noch einmal der Alltag eines Lehrers, der im Studium trotz aller Bemühungen aus meiner Sicht nicht vollständig wiedergespiegelt werden kann. Besonders beeindruckend fand ich dabei die Beobachtung, wie viel administrativer Aufwand von den Informatiklehrern neben ihrer eigentlichen Unterrichtsverpflichtung verlangt bzw. geleistet wird. Darunter vielen auch viele Dinge, für die ein Infollehrer eigentlich nicht zuständig sein sollte, wie die Hilfe bei vergessenen Lernsaxpasswörtern, private Nachhilfe für Kolleg*innen zu Lernsax in Freistunden oder auch mitten im Unterricht von Kolleg*innen gestört zu werden, bei denen der Laptop abgestürzt ist. Das waren Erfahrungen, die in der Fachdidaktik zwar in Randbemerkungen teilweise genannt wurden, aber die mir so nicht präsent waren. Sicher ist dabei aber auch zu beachten, dass die Handhabung solcher Fälle sich von Schule zu Schule unterscheiden kann. Darüber hinaus habe ich viel neues über die Schulform der Berufsschule gelernt, hier würde ich mir mehr Einblicke in alle Schulformen im Bildungswissenschaftlichen Bereich erhoffen, da man doch sehr unterschiedliche Erfahrungen sammeln kann, die auch im eigenen Berufsleben am Gymnasium nützlich sein können.

Auf den Unterricht als solchen habe ich mich durch das Didaktikstudium gut vorbereitet gefühlt. Besonders auch die große Auswahl an Tools, die wir kennengelernt haben, kam mir hier zu Gute und auch mein Mentor konnte davon noch profitieren.

6. Anhang

6.1. Unterrichtsentwürfe

1. Stunde: Grundkurs Informatik, Klasse 12, zweigeteilt: Formatierung einer Belegarbeit, Abfragen in MS Access

Lernziele:

- Die SuS sind in der Lage, Grafiken und Tabellen korrekt in eine Facharbeit einzubinden und entsprechende Verzeichnisse zu erstellen.
- Die SuS können sicher auch komplexe Abfragen mit Hilfe von Access erstellen.
- Die SuS sind in der Lage, sich selbstständig in einer neuen DB zu orientieren.

Uhrzeit	Unterrichtsschritt	Lehrer- / Schülerhandlung	Methode	Sozialform ²	Lernmittel, Bemerkungen
09:50 – 10:35	Übung	Gemeinsam wird erarbeitet, wie Grafikobjekte in Belegarbeit eingebunden werden, SuS machen am Rechner mit Zeit für andere Formatierungsfragen	L.-S.-Gespräch	FA	Material Festplattentext
10:35 – 10:40	Wiederholung	SuS wiederholen mdl, was eine Datenbank ist und grenzen DBMS, DB und DBS voneinander ab	L.-S.-Gespräch	FA	PPP mit Aufgabenstellung
10:40- 11:15	Übung	SuS arbeiten selbstständig an Abfragen in Datenbanken, zwischendurch werden Knackpunkte besprochen	Arbeit mit interakt. Medium	EA	DB Farben, Aufgabenblatt mit Abfragen

2

vgl. Schubert/Schwill: FA...Frontalarbeit, GA...Gruppenarbeit, PA...Partnerarbeit, EA...Einzelarbeit

11:15 – 11:20	Sicherung	Während die SuS die Plätze desinfizieren werden wichtigste Methoden noch einmal zusammengefasst		FA	Ergebnisse in Lernsax festhalten
------------------	-----------	---	--	----	----------------------------------

Kurze Reflexion:

Die Stunde war vor allem wegen des ersten Teils problematisch. Die SuS hatten die Gelegenheit, alle beliebigen Fragen zu stellen, daher war umfassendes Wissen, vor allem über die von der Schule genutzte Handreichung und ihre Bestimmungen, notwendig. Es lief aber alles problemlos. Außerdem wurde das Niveau der Kenntnisse über Abfragen falsch eingeschätzt, da die vorhergehende Stunde nicht hospitiert werden konnte. Dies konnte jedoch durch spontane Intervention und kurze Erarbeitung am Beamer gelöst werden.

2. Stunde: Grundkurs Informatik, Klasse 12, Einführung Datenbankmodellierung

Lernziele:

- Die SuS kennen den Aufbau eines ERD.
- Die SuS erstellen ein einfaches ERD zu einem gegebenen Problem.

Uhrzeit	Unterrichts-Schritt	Lehrer- / Schülerhandlung	Methode	Sozial form ³	Lernmittel, Bemerkungen
09:50 – 10:00	Wiederholung	Kahoot zu Datenbanken, letzte Frage zu Redundanz (bisher unbekannt)	Kahoot-Quiz	FA	Kahoot-Quiz
10:00 – 10:20	Erarbeitung	Erarbeitung der Begriffe „Redundanz“, „Konsistenz“ und „Integrität“ mit Arbeitsblatt und am Beispiel DB Student. SuS sollen klären, weshalb Tabelle ungünstig ist	Arbeit mit AB	PA	AB
10:10 – 10:50	Erarbeitung	Kurze PPP zu Aufbau von ERD	L.-S.-Gespräch	FA	PPP

3

vgl. Schubert/Schwill: FA...Frontalarbeit, GA...Gruppenarbeit, PA...Partnerarbeit, EA...Einzelarbeit

		Anschließend gemeinsame Erarbeitung ETD Studenten, SuS schreiben im Hefter mit			
10:50 – 11:10	Übung	SuS entwerfen ERD zu gegebenem Problem	Arbeit mit AB	EA	AB
11:10 – 11:20	Schluss	Rückgabe und kurze Auswertung Test zur Belegarbeits-Formatierung, Desinfektion		FA	Tests

Kurze Reflexion:

Obwohl bereits viel Zeit für die Erarbeitung des ERD eingeplant war, war der Plan immer noch zu voll. Bereits das Kahoot-Quiz zu Beginn dauerte länger als gedacht und auch die gemeinsame Erarbeitung eines Beispiel-ERD dauerte zu lang, so dass die letzte Übung gestrichen werden musste. Aus der guten Mitarbeit und den vielen guten Vorschlägen bei der gemeinsamen ERD-Erstellung lässt sich jedoch annehmen, dass die SuS bereits ein gutes Verständnis für das Modell entwickelt haben.

6.2. Unterrichtsmittel

Da es bereits sehr umfangreich ist, wird hier nur das Material zur beschriebenen hospitierten Stunde aufgeführt:

Programmierbeispiel Dreieck

```
01 procedure TForm1.Button1Click(Sender: TObject);
02 var basis, hoehe, flaeche, umfang: real;
03 begin
04
05   basis := StrToFloat(Edit1.Text);
06   hoehe := StrToFloat(Edit2.Text);
07
08   flaeche := 0.5*basis*hoehe;
09   umfang := basis+2*sqrt(0.25*basis*basis+hoehe*hoehe);
10
11   Edit3.Text := FloatToStr(flaeche);
12   Edit4.Text := FloatToStr(umfang);
13
14 end;
```

Notizen:

Abbildung 1- Arbeitsblatt Vorderseite

Programmierbeispiel Rechteck

```
procedure TForm1.Button1Click(Sender: TObject);
var _____, _____, flaeche, umfang: real;
begin
_____ := StrToFloat(Edit1.Text);
_____ := StrToFloat(Edit2.Text);

flaeche := _____;
umfang := _____;

Edit3.Text := FloatToStr(_____);
Edit4.Text := FloatToStr(_____);

end;
```

Übersicht Stationsarbeit:

Bearbeiten Sie die Stationen. Schätzen Sie nach der Bearbeitung ein, wie leicht oder schwer Ihnen eine Aufgabe fiel und vergleichen Sie die Aufgabe mit dem Lösungsdokument. Passen Sie Ihre Lösung gegebenenfalls an. Fassen Sie Ihre Lösungen der Pflichtstationen in einem Word-Dokument zusammen und drucken Sie sie aus.

Pflicht

Nr.	Thema	Einschätzung
Station 1	PAP erstellen Es werden ein Einkaufs- und ein erwarteter Verkaufspreis eingegeben. Liegt der Verkaufspreis mindestens 5% über dem Einkaufspreis, so soll die Gewinnsumme ausgegeben werden, andernfalls soll eine Fehlermeldung ausgegeben werden.	
Station 2	PAP interpretieren Interpretieren Sie den PAP aus dem Lehrbuch auf Seite 32 unten rechts. Erstellen Sie eine kleine Wertetabelle mit den Spalten „a“, „b“, „d“, „d<0?“, „Ausgabe“.	
Station 3	Quelltext interpretieren Interpretieren Sie den Quelltext aus dem Dokument „Station 3“. Vereinfachen Sie Rechnungen falls möglich. (Terme zusammenfassen).	

Wahlpflicht

Nr.	Thema	Einschätzung
Station 4	PAP erweitern Erweitern Sie ihren PAP aus Aufgabe 1 so, dass die Fehlermeldung danach unterschieden wird, ob der erwartete Verkaufspreis überhaupt größer als der Einkaufspreis ist.	
Station 5	PAP korrigieren Korrigieren Sie den PAP aus dem Dokument „Station 5“ dermaßen, dass Sie schlecht gewählte Bezeichnungen durch geeignetere ersetzen. Interpretieren Sie den PAP anschließend.	
Station 6	Zusammenhang zwischen Quelltext und PAP Bearbeiten Sie die folgende LearningApp: https://learningapps.org/watch?v=na24wonk520	
Station 7	Programmoberfläche beschreiben und interpretieren Beschreiben Sie die Komponenten, die Sie auf der Programmoberfläche im Dokument „Station 7“ sehen. Überlegen Sie, was dieses Programm tut.	

Zusatz:

Entwerfen Sie zu der Programmoberfläche aus Station 7 einen passenden PAP.

Form1

Datei

Rechteck

Seite a:

Seite b:

Umfang:

Flächeninhalt:

Abbildung 3- Formular Rechteck

Rechteck-Programm und PAP-Wiederholung

E. SPRÖTE

Das Rechteck-Programm (PAP)



Das Rechteck-Programm

PROGRAMMABLAUFPLAN



FORMULAR

Programmtest

Legen Sie auf Ihrem Arbeitsblatt eine Wertetabelle wie sie hier sichtbar ist an. Übernehmen Sie die zwei Beispielzeilen und ergänzen Sie mindestens zwei weitere.

Seite a	Seite b	Erwarteter Umfang	Umfang	Erwarteter Flächeninhalt	Flächeninhalt
3	3	12		9	
0	2	4		0	

Abbildung 4- Folien PPP

```
Station3.txt
1 procedure TForm1.Button1Click(Sender: TObject);
2   var a, b, c prod, sum: real;
3   begin
4
5     a := StrToFloat(Edit1.Text);
6     b := StrToFloat(Edit2.Text);
7     c := 2*a;
8
9     prod := a*b*c;
10    sum := b+c;
11
12    Edit3.Text := FloatToStr(prod);
13    Edit4.Text := FloatToStr(sum);
14
15  end;
```

Abbildung 5- Material Station 3

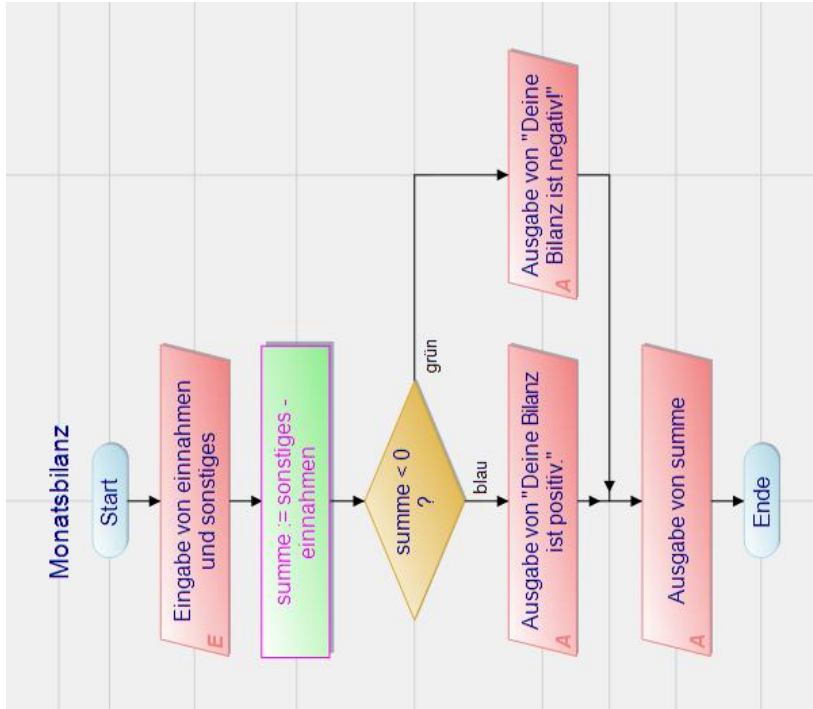


Abbildung 6- Material Station 5

The image shows a screenshot of a Windows application window titled "Form1". The window has a standard Windows title bar with minimize, maximize, and close buttons. The main content area has a light gray background and is titled "Quadrat" in a large, bold, black font. Below the title, there are three input fields and one button. The first input field is labeled "Flächeninhalt:" and is empty. Below it is a button labeled "Berechnen". Below the button are two more input fields: the first is labeled "Seitenlänge:" and the second is labeled "Umfang:". All input fields are empty.

Abbildung 7- Material Station 7

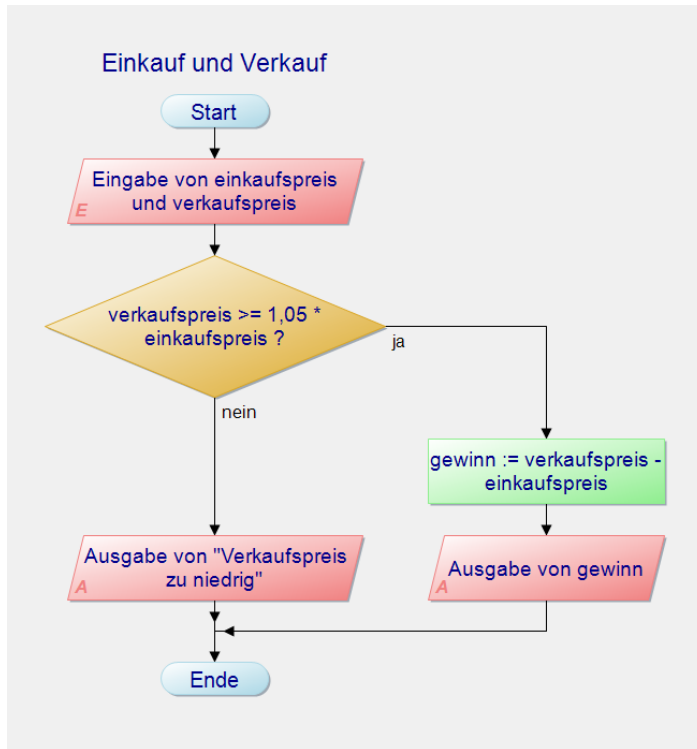


Abbildung 8- Lösung Station 1

Interpretation:

Es werden zunächst zwei Zahlen a und b eingegeben.

Anschließend wird die Differenz der beiden Zahlen berechnet. Ist die Differenz kleiner als Null, so wird ihr Vorzeichen umgekehrt. Anschließend wird in beiden Fällen die Differenz ausgegeben.

Mit Hilfe des beschriebenen Algorithmus wird folglich der Abstand der beiden Zahlen, also der Betrag ihrer Differenz, ermittelt und ausgegeben.

Wertetabelle: (beispielhaft)

a	b	d	d < 0?	Ausgabe
3	7	-4	ja	4
5	2	3	nein	3
6	6	0	nein	0

Gut gewählte Wertepaare:

1x $a > b$

1x $a < b$

1x $a = b$

Abbildung 9- Lösung Station 2

Interpretation:

In Zeile 2 werden die Variablen alle als Fließkommazahlen deklariert. In den Zeilen 5 und 6 werden die Variablen a und b aus Textfeldern eingelesen, in Zeile 7 wird die Variable c auf das Doppelte von a gesetzt.

In Zeile 9 wird das Produkt aus a, b und c berechnet, da $c=2*a$ ist wird also prod=2*a²*b berechnet.

In Zeile 10 wird die Summe $b+2*a$ berechnet.

Beide Ergebnisse (prod und sum) werden in zwei Textfeldern wieder ausgegeben (Zeile 12 und 13).

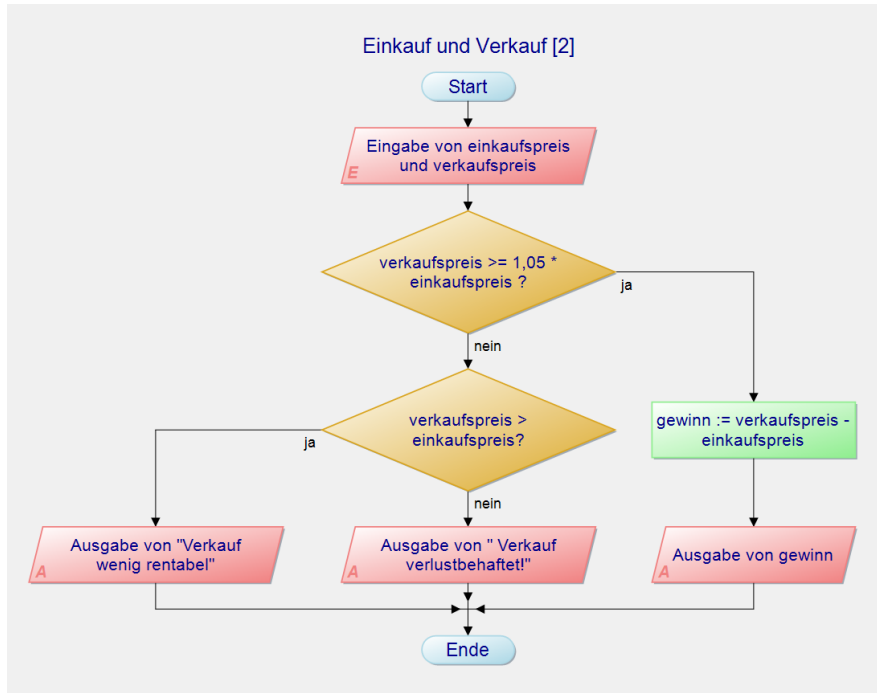
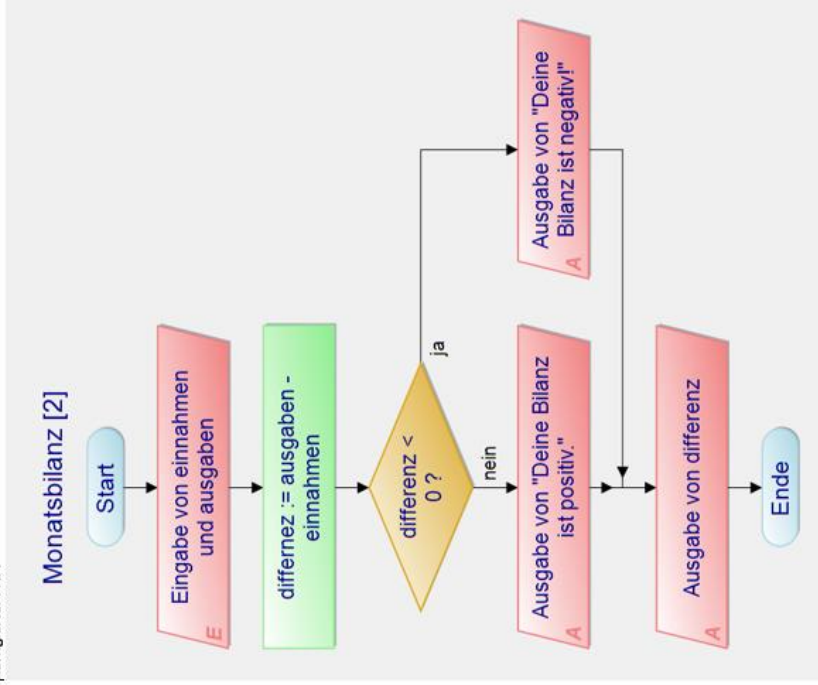


Abbildung 11- Lösung Station 4

Korrigierter PAP:



Interpretation:

Der Programmablaufplan bestimmt die Differenz aus Ausgaben und Einnahmen und gibt danach aus, ob die Bilanz positiv oder negativ ist. Anschließend wird in jedem Fall die Bilanz ausgegeben.

Abbildung 12- Lösung Station 5

Man sieht 4 Label mit den Inhalten

- Quadrat
- Flächeninhalt
- Seitenlänge
- Umfang

Man sieht einen Button mit der Aufschrift „Berechnung“.

Man sieht 3 leere Edit-Felder.

Offenbar wird der Flächeninhalt eines Quadrates eingegeben, aus welchem das Programm die Seitenlänge und den Umfang berechnet.

Abbildung 13- Lösung Station 7

Quadrat

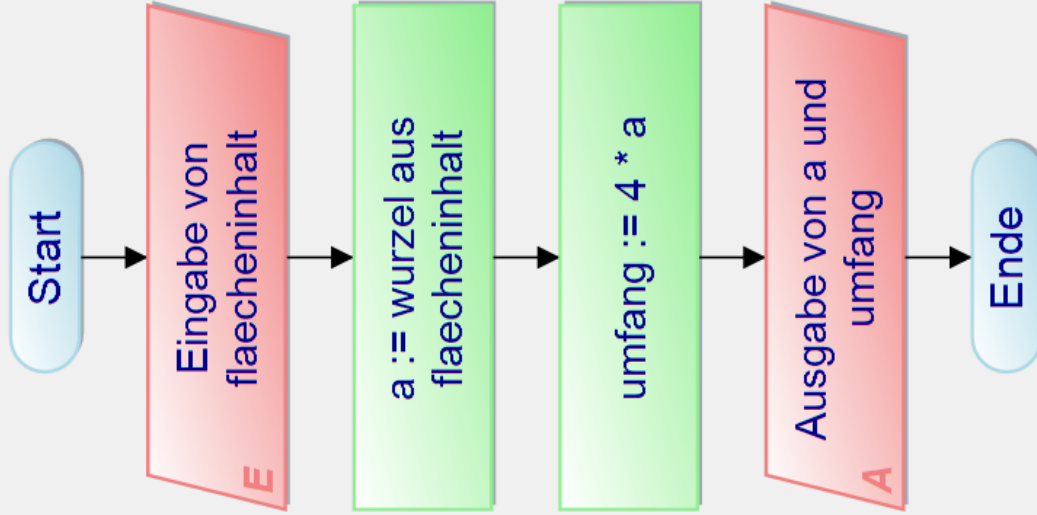


Abbildung 14- Lösung Zusatz

6.3. Hospitations-Protokolle

Berufliches Gymnasium, 11. Klasse, 02.09.2020: Steckbriefe in PowerPoint

Uhrzeit	Unt. Schritt	Lehrerhandlung	Schülerhandlung	Sozialform	Methode	Arbeitsmittel	Besondere Beobachtungen
12:00	Einstieg	L begrüßt SuS, geht Raumbelehrung durch	SuS füllen Sitzplan aus	FA	L.-S.-Gespräch	leerer Sitzplan	
12:10		L erklärt Aufbau Infohefter und Loginvergabe	SuS machen Notizen	FA	Lehrervortrag		
12:15	Erarbeitung	L stellt Stoffverteilungsplan vor	SuS schreiben als erste Hefterseite mit	FA	Lehrervortrag	Stoffverteilungsplan	
12:20	Erarbeitung	L schaltet WächterGuard aus	SuS melden sich an den Rechnern an, personalisieren Hintergrundbild	EA	Arbeit mit Rechner	Rechner	
12:30	Erarbeitung	L leitet wichtige Einstellungen in Word an	SuS machen mit	FA	Arbeit mit Rechner		
12:50	Erarbeitung	L skizziert an Tafel Aufbau des Schulnetzes	SuS vollziehen am Rechner nach	FA	Lehrervortrag	Tafel	
13:05	Übung	L gibt Hilfestellungen	SuS erstellen Steckbrief über sich als PowerPoint	EA	Arbeit mit PowerPoint		Keine Endzeit angesagt
13:27		L beendet Stunde	SuS desinfizieren Arbeitsplätze				

Wichtige Erkenntnisse:

Diese Stunde kann als exemplarischer Platzhalter für alle Stunden der ersten Schulwoche gesehen werden. Es war sehr eindrucksvoll, festzustellen, dass in der ersten Woche in 9 Klassen die ersten 45 Minuten nahezu identisch abliefen und der gesamte Unterricht sehr Lehrerzentriert ablief. Ich könnte mir für meine eigene Tätigkeit vorstellen, die Raumordnung beispielsweise als Puzzle zu gestalten, um diesen Teil interaktiver anzulegen. Besonders gut fand ich das gleichzeitige Skizzieren des Schulnetzes an der Tafel mit dem Nachvollziehen am Rechner, das hat den Schüler*innen sehr geholfen, die einzelnen Laufwerke korrekt ihrer Bedeutung und ihrem Sinn zuzuordnen.

Fachoberschule, 12. Klasse, 11.09.2020: Formatieren einer Facharbeit

Uhrzeit	Unt. Schritt	Lehrerhandlung	Schülerhandlung	Sozialform	Methode	Arbeitsmittel	Besondere Beobachtungen
09:50	Motivation	L begrüßt SuS, erläutert, wieso Formatierung der Facharbeit abseits des Lehrplans ausführlich behandelt wird	SuS hören zu	FA	Lehrervortrag		
09:55	Erarbeitung	L zeigt als Einschub, wie Signatur bei Lernsax erstellt wird	SuS machen mit	FA	Arbeit mit Rechner	Lernsax	Wurde zu Stundenbeginn vergessen
10:05	Erarbeitung	L zeigt verschiedene Formatierungen zur Facharbeit ausführlich, sucht Abschnite in Handreichung raus und macht vor, bezieht Schüler-Anmerkungen ein	SuS machen alle Formatierungen mit und machen Notizen, was wo in Handreichung zu finden ist	FA	L.-S.-Gespräch, Arbeit mit Word	Unformatierter Text, Handreichung	

		und lässt Dinge von SuS erklären Inhalt: Inhaltsverzeichnis Überschriften Quellenverzeichnis Zitation					
11:15		L beendet Stunde	SuS desinfizieren Arbeitsplätze				

Wichtige Erkenntnisse:

Auch wenn das Resultat der später erfolgten LK dem Lehrer in seiner Methodik Recht gibt, wurde die Stunde bei der Hospitation als sehr eintönig empfunden. Dies wird bereits daran deutlich, dass sich die wesentliche Phase zu 65 Minuten ohne Methoden- oder Sozialformwechsel zusammenfassen lässt. Hier wurde ein Thema nach dem anderen in einer Art „Klickschulung“ abgearbeitet. Später würde ich selbst versuchen, solche Teile abwechslungsreicher zu gestalten. Dazu könnte man beispielsweise, als einfachste mögliche Abwandlung, die Schüler*innen versuchen lassen, einzelne Dinge erst selbst herauszufinden. Als Sicherung kann man es dann noch einmal vormachen.

Auch der Einschub mit der Lernsax-Signatur wirkte etwas gezwungen und unterbrach den Unterrichtsfluss. Man könnte diese Phase beim nächsten Mal ans Stundenende setzen.

Berufliches Gymnasium, 11. Klasse, 21.09.2020: Hyperlinks in PowerPoint

Uhrzeit	Unt. Schritt	Lehrerhandlung	Schülerhandlung	Sozialform	Methode	Arbeitsmittel	Besondere Beobachtungen
12:00	Einstieg, Übung	L begrüßt SuS	SuS ergänzen ihre Steckbriefe	EA	Arbeit mit PowerPoint	PPP aus letzter Stunde	

12:07	Erarbeitung	L zeigt nebenbei für einige SuS, wie Signatur bei Lernsax eingerichtet wurde		FA	Lehrervortrag	Lernsax	
12:10	Übung	L macht sich Notizen zu den Schüler*innen	S präsentieren nacheinander ihre Steckbriefe	FA	Schülervortrag	PPP	
13:00	Motivation	L stellt Erwartungsbild zu „Opas Büro“ vor		FA	Lehrervortrag	Erwartungsbild	Opas Büro: PowerPoint mit Hyperlinks, die zwischen Gegenständen des Büros verlinken
13:07	Erarbeitung	L zeigt, wie man Hyperlinks in PowerPoint zwischen Folien erstellt	SuS machen mit	FA	Lehrervortrag	PowerPoint	
13:20	Übung		SuS arbeiten selbstständig für die letzten Minuten an „Opas Büro“	EA	Arbeit mit PowerPoint	Aufgabenstellung	
13:29		L beendet Stunde	SuS desinfizieren Arbeitsplätze				

Wichtige Erkenntnisse:

Diese Stunde fand ich vor allem inhaltlich sehr wertvoll. Die Idee mit den Steckbriefen und deren Präsentation hilft sehr, sich ein Bild von Schüler*innen zu machen, die man ja nur einmal pro Woche sieht. Man lernt hier sehr viel über die einzelnen Schüler*innen. Auch das Projekt „Opas Büro“ hat mir gut gefallen. Die Schüler*innen lernen hier viel über die grundlegende Funktion von Hyperlinks und damit implizit etwas über

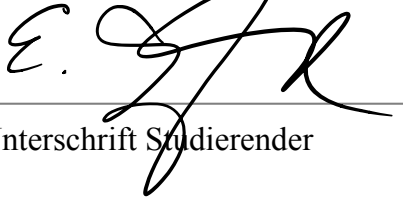
den Aufbau des Internets. Außerdem erweitern sie ihre Fähigkeiten im Umgang mit PowerPoint und erweitern im ersten Stundenteil ihre Präsentationskenntnisse.

7. Literatur- und Quellenverzeichnis

- Berufliches Schulzentrum 1 der Stadt Leipzig Wirtschaft und Verwaltung (2020): *Berufliches Schulzentrum 1 Leipzig Wirtschaft und Verwaltung*. Zuletzt abgerufen am 28.10.2020 von <https://www.bsz1leipzig.de/>
- Gesellschaft für Informatik e.V. (2016): *Bildungsstandards Informatik für die Sekundarstufe II*. Zuletzt abgerufen am 26.10.2020 von https://informatikstandards.de/fileadmin/GI/Projekte/Informatikstandards/Dokumente/Bildungsstandards_SII.pdf
- IGPM (Institut für Geometrie und praktische Informatik, RWTH Aachen) (2015): *Programmwurf mit Struktogrammen*. Zuletzt abgerufen am 29.10.2020 von <https://www.igpm.rwth-aachen.de/Download/ss15/cpp/nassi.pdf>
- Institut für elektronische Datenverarbeitung Zürich (Hrsg.) (1975): *IDV-Lernprogramm. Moderne Programmablaufplanung*. Berlin/New York: DE GRUYTER.
- Meyer, Albin (2018): *Softwareentwicklung. Ein Kompass für die Praxis*. Oldenburg: DE GRUYTER.
- Meyer, H. (2011): *Was ist guter Unterricht?*. Berlin: Cornelsen Verlag.
- Rauch, G.; Augsten, S. (2019): *Was ist Delphi?*. Zuletzt abgerufen am 20.10.2020 von <https://www.dev-insider.de/was-ist-delphi-a-843097/>
- Sächsisches Staatsministerium für Kultus (2020): *Lehrplan Berufliches Gymnasium. Informatik*. Zuletzt abgerufen am 26.10.2020 von https://www.schule.sachsen.de/lpdb/web/downloads/2232_lp_bgy_informatik_2020.pdf?v2

Eigenständigkeitserklärung

Hiermit erkläre ich, den vorliegenden Praktikumsbericht eigenständig und ausschließlich unter Verwendung der im Quellenverzeichnis angegebenen Literatur- und sonstigen Informationsquellen verfasst zu haben.



A handwritten signature in black ink, appearing to be 'E. Spröte', written over a horizontal line.

Unterschrift Studierender