Accepting an European Bachelor Degree for Admission to the Master Degree Program

Georg Wittenburg*

January 21, 2004

1 Introduction

29 European countries[1]are in the process of restructuring their higher education systems. Traditional national degrees are being replaced with standardized, internationally accepted degrees, such as the Bachelor and the Master degrees. The transition is planned to be completed by 2010. According to [1], the goals of this reform are to *"simplify the patchwork of higher education qualifications,"* to *"improve the mobility within Europe and attracting students from around the world,"* and to *"ensure high standards."* The reform is commonly referred to as the *"Bologna Process."*[2]

While only directly affecting European countries, the Bologna Process also has implications on educational institutions in other continents. As of the ever increasing mobility of students, it is desirable to take the changes in the European education sector into account when establishing related policies at non-European institutions. More specifically, non-European institutions need to consider in which way the degrees earned by visiting or transferring students from Europe can be integrated into their established systems of admission requirements.

Perhaps the most important single source of conflict arises from the fact that [2] states that *"access to second cycle* [Master degree studies] *shall require successful completion of first cycle studies* [Bachelor degree studies]*, lasting a minimum of three years."* The discrepancy lies in the fact that at Anglo-American institutions the first cycle of studies, the Bachelor degree, commonly takes four years to complete, rather than three years as proposed in the new European system. Due to this mismatch in duration it has been questioned whether these two degrees can be considered equivalent, especially in the context of admission to Master degree studies.

2 The Reasons

The following is a list of reasons on why Anglo-American institutions *should* accept the new European Bachelor degree as equivalent to their own Bachelor degree when considering applications of European students to their Master programs. Where applicable, exemplary background information from the case of the author is provided.

*The author has received his B.Sc. in Computer Science from Freie Universität Berlin, Germany and is seeking admission to the M.Sc. program at the University of British Columbia.

[1]Signatory countries are: Austria, Belgium (French community), Belgium (Flemish community), Bulgaria, Czech Republic, Denmark, Estonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovak Republic, Slovenia, Spain, Sweden, Swiss Confederation, and the United Kingdom

- **Equivalence of degrees is regulated by established international treaties.**

  Whether a degree is acceptable or not is already covered by established international treaties, such as the Washington Accord (`http://www.washingtonaccord.org/`). In [3] this treaty is described as:

  > *"[The Washington Accord] is an agreement between the bodies responsible for accrediting professional engineering degree programs in each of the signatory countries. It recognizes the substantial equivalency of programs accredited by those bodies, and recommends that graduates of accredited programs in any of the signatory countries be recognized by the other countries as having met the academic requirements for entry to the practice of engineering."*

  The accord was signed for Canada by the Canadian Engineering Accreditation Board of the Canadian Council of Professional Engineers (`http://www.ccpe.ca/`) in 1989. In Germany, ASIIN – Akkreditierungsagentur für Studiengänge der Ingenieurwissenschaften, der Informatik, der Naturwissenschaften und der Mathematik (`http://www.asiin.de/`) was the first accreditation agency to become a provisional member of the Washington Accord on June 11, 2003. ASIIN also was officially accredited by the German Accreditation Council (`http://www.accreditation-council.de/`) on December 12, 2002. The bachelor and master degree programs in computer science of Freie Universität Berlin were submitted to this agency in November 2003. The accreditation procedure is expected to be completed by May 2004.

- **Adequate background cannot be expressed in terms of duration of a degree program.**
  Organization and content of lectures vary between institutions. It is wrong to assume that a similar amount of material has been covered based on the duration of a program. In contrary, the comparison between degrees needs to be made entirely on the knowledge gained.

  At the Faculty of Computer Science of the University of British Columbia, the expected knowledge for admission to graduate studies is described in [8]. Appendix A shows how the B.Sc. in Computer Science of Freie Universität Berlin meets and exceeds these requirements.

- **Considering the Bachelor degree alone means ignoring differences earlier in the education.**
  For a comprehensive view of a students expertise, her performance in high-school needs to be taken into account. The knowledge gained in high-school will vary between educational systems and countries.

  More specifically, education in Germany is organized in such a way that high-school includes a 13th school year. In this 13th grade topics, content and knowledge is covered that is equivalent to first year courses at university in some areas. To illustrate this point appendix B contains the questions of final exams at German high-school (*"Abitur"*). The topics covered in this example correspond to to first year physics and foreign language courses at the University of British Columbia. In Germany, a students who has passed exams like these is eligible to study at any German university.

- **Three year Bachelor degrees are already being accepted by Anglo-American institutions.**
  While information on the acceptance of the new European Bachelor degree in non-European countries is scarce due to the lack of a central institutional body and the recent introduction of the degree programs at different universities, there are reports in the media about students who in fact were accepted into graduate studies based on this degree.

One of these is [11], which tells of a student from the city of Osnabrück, Germany who was accepted into the graduate program of Loughborough University, United Kingdom (`http://www.lboro.ac.uk/`) based on her three year Bachelor degree.

- **The degree is not a comprehensive indication of an individual students knowledge.**
  In fact, the degree only describes how well a student did in one particular program, while she may indeed have attended additional courses that are not part of the degree.

  For example, while earning his B.Sc. in Computer Science degree at Freie Universität Berlin, the author attended lectures that correspond to CPSC 411 Introduction to Compiler Construction, CPSC 415 Advanced Operating Systems, and CPSC 417 Computer Communications. These lectures are not part of the degree program. Furthermore, the author worked as teaching assistant for courses equivalent to CPSC 312 Functional and Logic Programming, and CPSC 319 Software Engineering.

- **Not accepting European Bachelor degrees is harmful for the exchange with European universities.**
  In case their Bachelor degrees are considered insufficient for admission to a graduate program, a continuation of their studies at an Anglo-American institution may not be considered as an option by European students.

  Time considerations play a crucial role in this argument: A student can earn a M.Sc. in Computer Science at Freie Universität Berlin in a five year timeframe. This is the same time as it would take her to earn a Hauptdiplom, the most widely accepted university degree in Germany nowadays. However, at [9] a Hauptdiplom is listed as a requirement for admission to graduate studies.[2] Any student coming from Germany wanting to earn his Master degree a the University of British Columbia would have to study for seven years, while she could earn the same degree in her homecountry in five years. This fact may negatively affect considerations of doing a Master degree at the University of British Columbia at all.

- **Problems can be addressed by the means of conditional admission.**
  It is possible to overcome problems or doubts as far as the qualification or knowledge of an applying student is concerned locally by the means of a conditional admission. A student with a non-standard degree could be required to pass a certain amount of courses whose topics depend of her current qualification in a certain timeframe. This way the institution is reassured that a student is indeed qualified for a program, while the student is given the chance to prove herself.

  In the case of the University of British Columbia, the current options are described in [7].

3 Conclusion

Several reasons were presented on why an European Bachelor degree as proposed in the Bologna Process should be accepted by non-European institutions as valid for admission into their graduate degree programs. These reasons include general observations on the current situation as well as examples based on the author's case.

It is hoped that this paper helps to understand why a reaction to the results of the Bologna Process at non-European institutions is required, and that eventually the European Bachelor degree will be accepted for admission to graduate progams at Anglo-American institutions.

[2] [9] has no provisions for the new European degrees, so this may very well a transitional problem that is solved if updates are made.

APPENDIX

A Adequate Background in Computer Science

In the following it is demonstrated that the B.Sc. in Computer Science degree as offered at Freie Universität Berlin [4] provides a student with an adequate background for entering graduate studies at the University of British Columbia. First, a summary of the requirements as stated in [8] is given, then follow the descriptions of the relevant courses based on [6] and for each of them the courses from Freie Universität Berlin are listed that cover the same topics, and finally descriptions of these courses based on [5] are provided.

The requirements as stated in [8] are as follows:

- **Required:**
  - A data structures course corresponding to CPSC 221, requiring a substantial amount of programming.
  - An analysis of algorithms course corresponding to CPSC 320.
- **Required:** (at least four courses chosen from at least three of the following areas)
  - **Artificial Intelligence:** CPSC 322, 422, 430, 435
  - **Computer Architecture:** CPSC 318, 418
  - **Databases:** CPSC 204, 404
  - **Graphics:** CPSC 414, 424
  - **HCI:** CPSC 444
  - **Operating Systems:** CPSC 313, 315, 415, 416, 417
  - **Programming Languages:** CPSC 311, 312, 411
  - **Software Engineering:** CPSC 310, 319, 352, 410
  - **Theoretical Computing:** CPSC 302, 303, 402, 403, 405, 420, 421

The equivalent courses from the B.Sc. in Computer Science program at Freie Universität Berlin are shown in the following list (UBC course descriptions are taken from [6]). Courses that are not shown are not covered by the B.Sc. program.

CPSC 221 Basic Algorithms and Data Structures. Design and analysis of basic algorithms and data structures; algorithm analysis methods, searching and sorting algorithms, basic data structures, graphs and concurrency.

Covered by: Algorithmen und Programmierung III, Entwurf und Analyse von Algorithmen

CPSC 304 Introduction to Relational Databases. Overview of database systems, ER models, logical database design and normalization, formal relational query languages, transaction processing, concurrency control and recovery.

Covered by: Datenbanksysteme

CPSC 310 Introduction to Software Engineering. Specification, design, implementation and maintenance of large, multi-module software systems. Principles, techniques, methodologies and tools for computer aided software engineering (CASE); human-computer interface, reactive systems, hardware-software interfaces and distributed applications.

Covered by: Software Technik

CPSC 311 Definition of Programming Languages. Comparative study of advanced programming language features. Statement types, data types, variable binding, parameter passing mechanisms. Methods for syntactic and semantic description of programming languages.

Covered by: Algorithmen und Programmierung II

CPSC 312 Functional and Logic Programming. Principles of symbolic computing, using languages based upon first-order logic and the lambda calculus. Algorithms for implementing such languages. Applications to artificial intelligence and knowledge representation.

Covered by: Algorithmen und Programmierung I[3]

CPSC 313 Computer Hardware and Operating Systems. Instruction sets, pipelining, code optimization, caching, virtual memory management, dynamically linked libraries, exception processing, exectution time of programs.

Covered by: Rechnerarchitektur

CPSC 318 Machine Structures. Machine organization and classification. Instruction formats and addressing. Input/Output including bus protocols, memory-mapped I/O, direct memory access, and interrupts. Processor architectures including instruction classes, instruction cycle, micro-programming, representation of numeric and non-numeric data. Memory organization. Advanced computer architectures.

Covered by: Rechnerstrukturen, Rechnerorganisation

CPSC 319 Software Engineering. The design, implementation, and test of a large software system, using a team approach.

Covered by: Software-Praktikum

CPSC 320 Intermediate Algorithm Design and Analysis. Systematic study of basic concepts and techniques in the design and analysis of algorithms, illustrated from various problem areas. Topics include: models of computation; choice of data structures; graph-theoretic, algebraic, and text processing algorithms.

Covered by: Entwurf und Analyse von Algorithmen

CPSC 352 Introduction to Software Engineering. Specification, design, implementation and maintenance of large multi-module software systems. Principles, techniques, methodologies and tools for software development.

Covered by: Software Technik

CPSC 404 Advanced Relational Databases. Physical database design, file organization, indexing and hashing, multimedia issues, relational query processing and optimization.

Covered by: Datenbanksysteme

CPSC 418 Advanced Computer Architectures. Introduction to advanced processor architectures and taxonomical views; performance considerations. Introduction to parallel machine designs. Examination of pipeline organizations; pipelined ALU and control units; representative architectures. Exploratory non-Von Neumann architectural models including: object-oriented, tagged, capability, dataflow and RISC designs.

Covered by: Rechnerarchitektur[4]

[3] This course does not cover logic programming, but concentrates entirely on functional concepts such lambda calculus, μ-recursion, and Haskell.

[4] The exploratory non-Von Neumann architectures discussed in this course are quantum computing and DNA computers.

CPSC 421 Introduction to Theory of Computing. Characterizations of computability (using machines, languages and functions). Universality, equivalence and Church's thesis. Unsolvable problems. Restricted models of computation. Finite automata, grammars and formal languages.

Covered by: Grundlagen der Theoretischen Informatik

The German courses referenced above are described as follows:[5]

Algorithmen und Programmierung I - Algorithms and Programming I: The development and description of algorithms is central to the area of algorithms and programming. This includes the theory of caputability, verification, and complexity, as well as practical programming. Specification and implementation of algorithms and data structures, and fundamental principles of programming languages and methodology are part of the course. While in the second semester programming is taught in an imperative language, this course uses functions for the description of algorithms. Functional programming is introduced with the Haskell programming language.

Algorithmen und Programmierung II - Algorithms and Programming II: This course continues the lecture series on algorithms and programming with an introduction to imperative and object-oriented programming. Key words: state, side-effects of instructions, iteration, classes and objects, pointers, type systems, parameter passing, graphical program representation, formal methods for the verification of imperative programs, stepwise correct code development, methods for testing, methodology of programming, imperative programming and computability, analysis of runtime and memory requirements, searching and sorting algorithms, I/O streams. Programming is done in Java and in Haskell.

Algorithmen und Programmierung III - Algorithms and Programming III: In the third semester of the lecture series algorithms and programming the course concentrates on data and progam structures. Starting point is the information hiding principle and its importance for the structure of programs and for the construction of data objects with modules and classes. A central role in the modeling of data plays the concept of data abstraction in the context of the difference between specification and implementation of abstract data objects and types. Sets, relations, lists, trees, graphs and other abstract data types are introduced. Efficient representations of these types are discussed and their algorithms are examined for their complexity. The methods used for representation and data structures include hash transformations, binary search trees, and suffix trees. In object oriented programming, additionally to data abstraction inheritance and polymorphism play a major role. This is why abstract data types will be specified and implemented by the means of inheritance. Programming is done in Java for the imperative side, and in Haskell for the functional side.

Datenbanksysteme - Database Systems: The area of database systems is interconnected with many other areas of computer science. The development of a database application starts with the DB design. Related (semi)formal tools are covered in the beginning of the course. The relational data model is of central importance. After this, aspects of implementation, the transaction concept, and methods to improve error tolerance are discussed. Web-based interaction with databases and distributed data storage are covered. Goal of the course is to provide a systematical overview of the subject and to introduce to the operational aspects of databases. We will use Oracle9i, one of most popular commercial systems.

Entwurf und Analyse von Algorithmen - Design and Analysis of Algorithms: The design of algorithms is one of the core competencies of computer science. This lecture is an introductory course into algorithmic theory and basis for all advanced-level courses in theoretical

[5] Translated by the author based on [5].

computer science. It deals with design and analysis of algorithms and data structures for many fundamental problems such as searching, sorting, graph problems, arithmetic, geometrical problems, etc.

Grundlagen der Theoretischen Informatik - Fundamentals of Theoretical Computer Science: This course gives an overview of the most important concepts of theoretical computer science. It concentrates on automata, formal languages, grammars and Chomsky-hierarchy, Turing machines, computability, and complexity of problems.

Software-Praktikum - Software Lab: Team-based development of a major software system. This course extends on the knowledge gained in the algorithms and programming lectures. It leads to the development of a major software system and is organized as a project. Students are split into several groups in the first lecture. The assignment is to develop a web-based business system (webshop). Modules to be implemented include business-to-costumer, data management, and decision support modules. The project is organized according to the EASE model. Form-based methods are used for system modeling.

Software Technik - Software Engineering: Software engineering is the science of the construction of software, the fundamental course for the methodology. Software enginering tries to answer questions such as: How do you establish, which properties a piece of software should have? How do you describe these properties? How do you structure a piece of software for it to be easily constructed and adapted? How do you adapt software that does not have this kind of structure or the structure is no longer recognizable? How do you discovered deficiencies of a piece of software? How do you organize the work in a software company or department in order to produce cost-efficient and high-quality products on a regular basis?

Rechnerarchitektur - Computer Architecture: Introduction: definition of terms, taxonomy, basics of computer system design, hardware description languages, performance measurement/benchmarking. Architecture and Implementation of Microprocessors: von-Neumann-architecture, microcomputer, microprocessor, today's PCs, from RISC to super scalar, pipelining, pipeline hazards and solutions, SIMD and multimedia, VLIW and EPIC. Memory Organization: registers, register assignment and register window, memory hierarchy, virtual memory management and cache memory. Processor Examples: Pentium 4 and Athlon. Further Developments: super speculative, trace-cache, single-chip-multiprocessor, multi threaded processors, data flow. Multi-Processor Systems: general basics, interconnections, performance, memory coupled multiprocessors: SMP and DSM, cache-coherency and memory consistency, examples, message coupled multiprocessors, examples: IBM SP2 and Top500 list. Vector Computers: introduction, performance, example: Fujitsu VPP. Field Computers: basic principle, (historical) example: MasPar. Reliability and Fault Tolerance. "Unconventional" Architectures: quantum computing, DNA computers.

Rechnerorganisation - Computer Organization: Based on the course on computer structures, this course takes the step from a simple computer to a complete computer system. Topics include CPUs, RISC/CISC, assembler, I/O, bus systems, controller, DMA, interrupt handlers, memory, peripheral devices, and networking.

Rechnerstrukturen - Computer Structures: This course is fundamental to further lectures, tutorials, seminars and labs in the area of technical computer science. The following topics are covered: machine-internal data representation, error detection and correction, logic and boolean circuits, design of circuits, coder/decoder, multiplexer, ROM, PLA, adder, circuit networks (synchronous/asynchronous), automata, flip-flops, RAM, memory, shifters, control units, and basic machine architecture.

B German Final High-School Exam Questions

These exam questions[6] as posed in a typical final high-school examination (*"Abitur"*) are to illustrate the level of proficiency expected from a student when graduating from German high-school. It is argued that the topics covered are similar to those of first year courses in the respective areas at the University of British Columbia.

The duration of these exams is 300 minutes and they cover the material from the last two years in high-school. Further to the two exams presented in this paper, the final examination also consists of one additional writting exam (200 minutes) and one oral exam (15 minutes with 30 minutes for preparations based on a handout). If the results in the written exams differ significantly from the results expected based on past performance, the student may be required to pass an additional oral examination. Regulations differ between German provinces; [10] provides a collection of relevant laws. Successful completion of this final examination is the one single prerequisite for studies at an university in Germany.

B.1 Physics

Excitement of Atoms by Electron Collisions

1. Of two identical triodes one is evacuated, the other one contains helium at low pressure. In the following, this tube is referred to as "gas triode." For both tubes the I_A-U_{KA}-line is measured according to circuit 1 (measurement 1). Draw both lines into a diagram and interpret their behavior, taking into account the differences between the two curves.
2. (a) Make a sketch of the circuit of the second experiment and explain its meaning. Based on the values from measurement 2, draw the $I_A(U_{KG})$ graph and explain its characteristics.

   (b) Compare the curve to the graph of a quicksilver-gas-triode as discussed in class. Describe potential processes of energy absorption and emission on the atomic level. Calculate the wave length of the photons that are produced during the experiment.
3. Starting at $U_B \approx 25V$ a slight blue light appears between the grid and the cathode. It is hard to see because the heating of the cathode is very bright. This light consists of the wave lengths of $\lambda_1 = 558nm$ and $\lambda_2 = 447nm$. In a helium atom, the following energy levels can be observed: $-0.6eV$; $-1.2eV$; $-3.4eV$; the ionization energy is $24.5eV$. Make a sketch of the corresponding energy level diagram. Interpret the spectral lines based on this diagram and speculate on additional spectral lines. Place the result from question 2. b) into this context.
4. In the classic Franck-Hertz-experiment the quicksilver atoms emit light with a wave length of $\lambda = 254nm$. Calculate energy and impulse of these photons. An electron needs to accelerate through $4.9V$ in order to stimulate a quicksilver atom. Calculate the kinetic energy of an electron that exactly matches the impulse of a photon. Compare the results and interpret the differences.
5. Assume a vacuum tube with a direct current of $50V$ between cathode $(-)$ and grid $(+)$; the anode is not connected. Assume further that the field is homogenous and the distance between cathode and grid is $1.2cm$.

   (a) Describe the trajectory of an electron that passes from the cathode through the grid towards the anode. Calculate the speed of the electron just before passing the grid. Make a sketch and give reasons for the dependency $v(s)$ of the speed of the electron between cathode and anode in relation to the travelled distance s.

   (b) Make a sketch and give reasons for the dependency $W(s)$ of the kinetic energy of the electron in relation to the travelled distance s between cathode and anode.

Resources: Graphic calculator TI-83, conventional calculator Casio f 100 (opt.), Fischer/Dorn: Physical Formulas und Figures, Klett, Stuttgart 1982

[6] Taken from the school year book of Gymnasium Carolinum Osnabrück, Germany (`http://www.carolinumosnabrueck.de/`). Partly translated from German by the author.

B.2 English as a Foreign Language

[...] But all women become conscious, sooner or later, of that impalpable, but steel-strong pressure to get married, and Mary, who was not at all susceptible to atmosphere, or the things people imply, was brought face to face with it suddenly, and most unpleasantly. She was in the house of a married friend, sitting on the verandah, with a lighted room behind her. She was alone; and heard people talking in low voices, and caught her own name. She rose to go inside and declare herself: it was typical of her that her first thought was, how unpleasant it would be for her friends to know she had overheard. Then she sank down again, and waited for a suitable moment to pretent she had just come in from the garden. This was the conversation she listened to, while her face burned and her hands went clammy, 'She's not fifteen any longer: it is ridiculous! Someone should tell her about her clothes.'

'How old is she?'

'Must be well over thirty. She has working, and that was a good twelve years ago.'

'Why doesn't she marry? She must have had plenty of chances.'

There was a dry chuckle. 'I don't think so. My husband was keen on her himself once, but he thinks she will never marry. She just isn't like that, isn't like that at all. Something missing somewhere.'

'Oh, I don't know.'

She's gone off so much, in any case. The other day I caught sight of her in the street and hardly recognized her. It's a fact! The way she plays all those games, her skin is like sandpaper, and she's got so thin.'

'But she's such a nice girl.'

She'll never set the rivers on fire, though.'

'She'd make someone a good wife. She's a good sort, Mary.'

'She should marry someone years older than herself. A man of fifty would suit her... you'll see, she will marry someone old enough to be her father one of these days.' 'One never can tell!'

There was another chuckle, good-stunned and outraged; but most of all deeply wounded that her friends could discuss her thus. She was so naive, so unconscious of herself in relation to other people, that it had never entered her head that people could discuss her behind her back. And the things they had said! She sat there writhing, twisting her hands. Then she composed herself and went back into the room to join her treacherous friends, who greeted her as cordially as if they had not just that moment driven knives into her heart and thrown her quite off balance; she could not recognize herself in the picture they had made of her!

That little incident, apparently so unimportant, which would have had no effect on a person who had the faintest idea of the kind of world she lived in, had a profound effect on Mary. She, who had never had time to think of herself, took to sitting in her room for hours at a time, wondering: 'Why did they say those things? What is the matter with me? What did they mean when they said that I am not like that?' And she would look warily, appealingly, into the faces of friends to see if she could find there traces of their condemnation of her. And she was even more disturbed and un happy because they seemed just as usual, treating her with their ordinary friendliness. She began to suspect double meanings where none were intended, to find maliciousness in the glance of a person who felt nothing but affection for her.

Turning over in her mind the words she had by accident listened to, she thought of ways to improve herself. She took the ribbon out of her hair, though with regret, because she thought she looked very pretty with a mass of curls round her rather long thin face; and bought herself tailor-made clothes, in which she felt ill at ease, because she felt truly herself in pinafore frocks and childish skirts. And for the first time in her life she was feeling uncomfortable with men. A small core of contempt for them, of which she was quite unconscious, and which has protected her from sex as surely as she had been truly hideous, had melted, and she had lost her poise. And she began looking around for someone to marry. She did not put it to herself like that; but, after all, she was nothing if not a social being, though she had never thought of 'society', the abstraction; and if her friends were thinking she should get married, then there might be something in it. If she had ever learned to put her feelings into words, that was perhaps how she would have expressed herself. And the first man the allowed to approach her was a widower of fifty-five with half-grown children. It was because she felt safer with him ... because she did not associate ardours and embraces with a middle-ages gentleman whose attitude towards her was almost fatherly.

Annotations:

1. impalpable – not able to be felt or understood
2. susceptible – sensitive / able to feel
3. clammy – moist or damp / covered with sweat
4. She has been going strong (infml): She has been flourishing / thriving
5. chuckle – an amused little laugh

6. She's gone off so much – here: her best days are over
7. She'll never set the rivers on fire – She will never fill other people with enthusiasm / love for her
8. malicious – feeling an urgent desire to harm others
9. stunned – deeply shocked / made unable to think or speak clearly
10. writhing – twisting violently to and fro, especially in pain or discomfort
11. composed herself – calmed herself / got her thoughts and feelings under control
12. profound – deep
13. appealingly – with a manner that wants to arouse sympathy
14. tailor-made – *maßgescheidert*
15. pinafore frocks – sleeveless dresses for wearing over a blouse, sweater etc.
16. core – essential part (*Kern*)
17. hideous – repulsive ugly / extremely unattractive
18. poise - self-confidence / balance
19. ardours – great passions

Questions:

1. Sum up the incident that shatters Mary so much and what changes in her selfesteem are brought about by it.
2. Define the attitude of Mary's friends towards her. Justify your impressions.
3. Delineate the author's means to convey the very complex picture of Mary's character to the reader.
4. Chose one of the following tasks:
   (a) Put yourself in Mary's position. Image Mary has written to an agony aunt about the behavior of her friends. Take over the role of the agony aunt. Give some brief pieces of advice.
   (b) Do you feel sympathy or contempt for Mary's attitude? Support your view with arguments.

Resources: Pons Cobuild English Learner's Dictionary

References

[1] European Commission. *The Bologna Process.*
http://europa.eu.int/comm/education/policies/educ/bologna/bologna_en.html

[2] Confederation of EU Rectors' Conferences and the Association of European Universities (CRE). *The Bologna Declaration on European Space for Higher Education: An Explanation.*
http://europa.eu.int/comm/education/programmes/socrates/erasmus/guide/bologna.pdf

[3] Washington Accord. *Frequently Asked Questions.*
http://www.washingtonaccord.org/wash_accord_faq.html

[4] Freie Universität Berlin. *Amtsblatt der Freien Universität Berlin vom 18.2.2003.*
http://www.fu-berlin.de/service/zuvdocs/amtsblatt/2003/ab022003.pdf

[5] Freie Universität Berlin, Fachbereich Mathematik und Informatik, Institut für Informatik. *Kommentiertes Vorlesungsverzeichnis (KVV).*
http://www.mi.fu-berlin.de/kvv/

[6] University of British Columbia. *Calendar 2003/04.*
http://students.ubc.ca/calendar/

[7] University of British Columbia. *Types of Admission Offer.*
http://www.grad.ubc.ca/policy/index.asp?menu=002,008,000,000

[8] University of British Columbia, Computer Science Department. *List of Required Undergraduate Courses.*
http://www.cs.ubc.ca/prospective/grad/admissions/Unclassified_Year.html

[9] University of British Columbia. *Graduate Studies – Minimum Requirements.*
http://www.grad.ubc.ca/apply/how/require.asp

[10] Deutscher Bildungsserver. *Schulrecht.*
http://www.bildungsserver.de/zeigen.html?seite=72

[11] *Zweitstudium in England.* Neue Osnabrücker Zeitung, p. 16, September 27, 2003