Informatics Matura Examination: Useful or Not?[1]

Ewa Kolczyk

Institute of Computer Science, University of Wroclaw
Joliot Curie Str. 15, 50–383 Wroclaw, Poland
eko@ii.uni.wroc.pl

Abstract. One element of the recent educational reform in Poland is the qualification system, which consists of several types of external examinations. The results of the Matura examinations, taken at the end of secondary education, are now replacing entrance examinations to universities, and to other institutions of higher education. The informatics Matura examination, considered as an outcome of the secondary curriculum, and as a regulation for informatics education, could play a significant role in the process of improving school education in this discipline. We decided to compare computing A-level examination in England with the informatics Matura examination in Poland. The comparison is conducted on different levels: subject criteria, scheme of assessment, examples of tasks, examiners reports.

1 Introduction

One element of the recent educational reform in Poland is the qualification system, which consists of several types of external examinations. The role of the system is to ensure the comparability of learning outcomes. The external forms of assessment conducted at every educational stage are intended to help students, parents, teachers, schools and education authorities to make clear which schools exemplify the appropriate level of teaching. The results of these external examinations are accepted as an element of requirements for schools in the next stage in admitting candidates. For example, the secondary schools take into consideration the results of tests passed at the end of middle schools, and of the pupils' grades from the school. The results of the Matura examinations, taken at the end of secondary education, are now replacing entrance examinations to universities, and to other institutions of higher education.

The specification for the informatics Matura examination is, at the same time, an element of informatics education and of the qualification system. Universities were very careful in setting their entrance requirements, particularly based on the informatics maturity examination results. Only few institutions allow this qualification as an additional element of their entrance requirements. Computer

[1] This article was first published in: Information Technologies at Schools, The 2nd International Conference "Informatics in Secondary Schools: Evolution and perspectives". Selected papers, November 7-11, 2006, Vilnius, Lithuania. Eds.: Valentina Dagienė and Roland Mittermeir. Institute of Mathematics and Informatics, Vilnius, 2006.

Science departments at universities mainly require candidates to pass a mathematics Matura examination as continuation of the traditional entrance examination in mathematics.

The specification for the informatics Matura examination, considered as an outcome of the secondary curriculum, and as a regulation for informatics education, could play a significant role in the process of improving school education in this discipline.

The article in previous ISSEP proceedings [1] is about the correlation between passing computing A-level examination and later performance in computing university course. Its authors asked some important and inspiring questions, so we decided to refer to their work and to compare computing A-level examination with the informatics Matura examination in Poland. The comparison has to be limited, because our experience is itself necessarily limited. The first specification for the informatics Matura examination was published in 2000, and there were three terms for students to pass this examination (May 2002, January 2003, May 2005). The comparison is conducted on different levels: subject criteria, scheme of assessment, examples of tasks, examiners reports.

2 Subject Criteria in England

Subject criteria for computing AS/A level are defined by the Qualification and Curriculum Authority (QCA) [2]. They are formulated in two areas.

2.1 Knowledge and Understanding

The main emphasis is placed on understanding the purpose and impact of a range of computing applications and on the characteristics of networks and contemporary processors, input, output and storage devices; the need for and means of connection between devices; and the importance of adopting standards. The candidate should be familiar with key concepts important for software engineering (e.g. system development life cycle, user interface, user requirements, technical and user documentation) and for databases (e.g. data structures, methods of finding, selecting and managing information).

2.2 Skills

The skills are closely related to the analysis, design, implementation and evaluation of real information processing systems.

The subject criteria are developed in the A-level specification prepared by the examination board. After studying the one prepared by the Assessment and Qualifications Alliance (AQA) [4], we found the computing discipline presented from a well-defined point of view. The student is not expected to have experience and practice in programming, but is allowed, rather, to know some programming language

concepts. He, or she, needs to know simple algorithms, but much more important is the ability to select appropriate software, and test the solution in meeting the requirements of the problem. As far as it is possible to explain it in one sentence, the student is expected to present more general knowledge than specific skills. The skills, defined in the AQA specification [4], are assessed by doing a real project. It is a part of the internal assessment, so the student is able to work for an extended period of time on a well-defined activity.

3 Subject Criteria in Poland

In Poland, the subject criteria included in the specification for the informatics Matura examination [7] are formulated in three areas:

- knowledge and understanding,
- using information,
- creating information.

3.1 Knowledge and understanding

The student demonstrates knowledge and understanding of basic concepts, methods, tools and processes related to computing, and is able to:

1. describe computing devices, tools and methods using proper technical terms,
2. describe role, function and principles of using computer systems (e.g. computer, input, output and storage devices, networks),
3. describe different types of application software and its use,
4. assess suitability and reliability of different sources of information and representation of different forms of information,
5. explain basic algorithms (e.g. serial search, binary search, sorting, recursion, numerical calculations);
6. describe IT development in modern society and its perspectives.

3.2 Using information

The student applies knowledge to the solving practical and theoretical tasks, and is able to:

1. use typical application software,
2. solve tasks by using existing solutions,
3. use networks for communication, file transfer, publishing own materials,
4. apply searching methods for processing information in relational databases,
5. use algorithms for solving typical problems,
6. choose appropriate application software for solving task,
7. use knowledge and skills for solving tasks from school's and everyday life.

3.3 Creating information

The student uses computing methods for solving problem, and is able to:

1. analyse problem, specify and design its solution,
2. use computing methods (top-down design, structured approach, algorithmic strategies) for solving problems,
3. write task solutions in the form of algorithms using a list of steps, flow charts, programming language,
4. evaluate correctness and efficiency of solution,
5. design and create a relational database,
6. use computing methods for modeling and simulation,
7. use information from various sources for creating multimedia documents,
8. discuss and formulate opinions about social, legal and ethical consequences of the use of information and computers.

In each area, there are formulations related to algorithms, and in the point III. 3 it is directly stated that students should be able to write a solution in a programming language. Further developments of the subject criteria involves choosing the appropriate application software for solving task and it means choosing a programming language, in some cases. The student is expected to make decision which tool should be used for solving a given task.

4 Scheme of Assessment

Subject criteria are formulated at a very general level, so the comparison between examinations should be more fruitful if we look at the scheme of assessment and real tasks.

AS/A level computing specification prepared by AQA [4] meets the general rule, that AS is a first part and A2 – second (advanced) part of A-level examination. Each of AS and A2 levels consists of three units, two theoretical and one practical. Passing all these units takes 360 minutes for 4 theoretical units and some extended period of time to complete the real piece of work in two practical units. This meets the well-known rule that the longer period of assessment time and doing a sample of real work during the examination promotes more honest evaluation of the student's achievements.

The informatics Matura examination in Poland can be passed only at an advanced level. It consists of two units. The first one is a theoretical unit, with no use of a computer. It takes 90 minutes and student has to deal with short answer questions and extended response questions. Unit 2 takes 150 minutes and students work on practical tasks using computers. In both units students work is marked externally by examiners.

At this stage of comparison, the most important aspect seems to be the approach to the practical part of the examination. In the unit 3 of AQA specification [4] the student should solve a real problem formulated by the examination board, using appropriate software and coming through all stages of problem-solving. He has to prepare detailed documentation, which later is marked by the teacher and additionally he has to attach it to his answer sheet during the examination. This approach has two

great advantages: the student has enough time to complete the task connected with real life and to learn how to prepare good documentation.

In the unit 2 of the informatics Matura examination in Poland, the student has limited time to analyse the problem, to choose an appropriate tool, and to solve the task. In several tasks the examiners receive and mark the results only. The correct results could be obtained in a set time, if you know what to do, and you are clever in using appropriate software. The limited time is, equally, both an advantage and a disadvantage. It is possible to find which students are clever in choosing the best tool and in thinking quickly and precisely. On the other side, there is a limited range to the tasks which could be solved in such a limited time. Another problem is to write the documentation. In the 2002 examination, students were asked to write documentation about the solution during the examination. It became clear, that this was very difficult for many students. Later, the role of the documentation was reduced, but it is clear that we were losing the opportunity of evaluating a very important skill.

5 Examples of Examination Tasks

The first example (Table 1) is a practical task from unit 2 of the informatics Matura examination (January 2003) [8].

Table 1. **Numbers** – task from unit 2 (Matura examination, January 2003)

Folder DATA contains text files NUMBERS1.TXT and NUMBERS2.TXT, which consist of natural numbers (positive integers). Each number is put in a new separate line.

Your task is to prepare text file RESULTS.TXT, which contains answers to the following questions:

1. What is the smallest number in NUMBERS1.TXT?
2. How many digits together are in NUMBERS1.TXT and NUMBERS2.TXT?
3. How many times digit 8 is present in NUMBERS1.TXT?
4. Put in the file RESULTS.TXT all numbers from NUMBERS1.TXT, which do not contain digit 7.
5. Put in the file RESULTS.TXT all numbers, which are present in both files NUMBERS1.TXT and NUMBERS2.TXT.
6. If all numbers in file NUMBERS1.TXT are sorted in ascending order, what numbers are at the positions: 1000, 1500 and 2000?

In solving the task from Table 1, it is important to notice that file NUMBERS1.TXT contains 2000 numbers and that in both files there are numbers consisting of at most 54 digits. Hence, the numbers in data can not be represented as integers in the software applications used by students. They have to notice this property of data during the analysis of the task and to decide that the numbers should be represented as text (strings of digits). This task reviews also, if students are able to choose appropriate software. The answers to four out of the six questions are easy to find using a spreadsheet and in the case of questions 3 and 4 a simpler solution can be

obtained by writing a program. In the mark scheme, the choice of the tool is not taken into consideration, because students submit only the files with results (RESULTS.TXT). On the other hand the time limits and the character of data eliminate non-effective ways of solving the task.

The next example (Table 2) is the task from unit 1 of the informatics Matura examination (May 2002) [8].

Table 2. **Weighing** – task from unit 1 (Matura examination, May 2002)

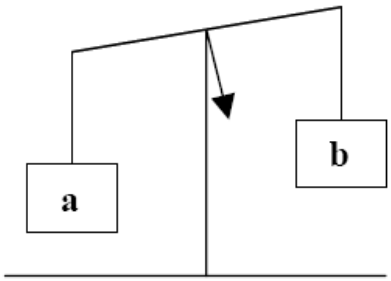

Fig. 1. Scales with two containers

There are *n* small objects of different weight and the scales with two containers, as you can see in the Fig. 1. You can only put two objects *a* and *b* in the same time into containers (this means one operation of weighing) and this way you are able to decide, which of two objects is lighter. Your task is to find the lightest and the heaviest object from given *n* objects, using such the scales.

1. What is the smallest number of operations of weighing needed to find the lightest object? Explain it.
2. Specify exactly the task of finding the lightest and the heaviest object in the same time. Write an algorithm for this task using list of steps, flow chart or programming language, which performs possibly smallest number of operations of weighing.
3. What is the number of operations of weighing performed by your algorithm? Explain it.

In this task, the student is expected to recognize the process of weighing and finding the lightest and the heaviest object as an algorithm for finding minimum and maximum in the file simultaneously. He or she is also expected to be able to explain the complexity of this method and to write an algorithm in a chosen notation.

The following task (Table 3); published as Example one of A2 level on the QCA website [3] looks similar to the previous example (Table 2), although it concerns other algorithmic methods.

Table 3. Example one, Computing Exemplification A2 level

1. A rail company in Greater London plans to store data about its stations. Data about the following five stations is stored in a linked list.

Hainault, Fairlop, Barkingside, Newbury Park, Gants Hill

With the aid of a diagram, show how these stations can be stored in alphabetical order in a linked list.

2. The names

Brentwood, Chelmsford, Dover, Epping, Frinton, Harwich, Ilford

are stored in a sequential file. Show the stages of

- a serial search
- a binary search

to find the name Frinton.

3. State one advantage and one disadvantage of a binary search compared with a serial search.
4. Write an algorithm to merge two files. State any assumptions you make.

The student is expected to apply the concepts of a linked list, a serial and binary search for the concrete situation and limited data. The student should demonstrate an understanding of the methods. In the question 4, he has to describe the method of merging two sorted files. The task (Table 3) deals with more concepts than the previous one, but in the previous task (Table 2), the student should describe the method and its complexity in general terms not only for a specific example.

The next example (Table 4) is from unit 5 (AQA, January 2005) [5] and concerns the relational databases.

Table 4. Question 9, Unit 5, Computing A-Level Examination (AQA, January 2005)

A teacher of Advanced Level Computing uses a relational database to record details of

- Students
- Work done by students on their Advanced level project
- Marks awarded for the system life cycle phases of a student's project
- Description and maximum mark for each system life cycle.

The teacher assigns

- A unique student identifier to each student
- A unique identifier to each system life cycle phase.

Work done by a student on a project is carried out

- On more than one date
- On different system life cycle phases on different dates
- On no more than one system life cycle phase on any particular date
- In only one time period on any particular date.

Students submit their work for assessment after completing each phase of the system life cycle.

The relational database uses four tables:

Student (StudentID, FirstName, Surname, DateOfBirth)

WorkDone (Date, StudentID, StartTime, LifeCyclePhaseID,

DescriptionOfWorkDone, TimeSpent)
MarkAwarded (StudentID, LifeCyclePhaseID, DateSubmitted, Mark, DateMarked, TeachersComments)
LifeCyclePhase (LifeCyclePhaseID, LifeCycleName, MaximumMark)

1. Draw an Entity-Relationship (E-R) diagram for the tables:
   - Student and WorkDone
   - Student and MarkAwarded
   - LifeCyclePhase and MarkAwarded
2. Using the SQL commands SELECT, FROM, WHERE and any others considered appropriate, write an SQL statement to query the database tables for each of the following
   - The first name and surname of every student;
   - The first name and surname of every student together with the mark awarded for the phase of the system life cycle with LifeCyclePhaseID = 1, presented in ascending order of student surname.

This is a typical task, and students are expected to draw E-R diagrams on the basis of given definition of the tables and to write simple queries using SQL language.

Examples of similar tasks can be found in every practical part (unit 2) of the informatics Matura examination in Poland. A task dealing with a database usually requires a longer description; we therefore decided not to present here such a task from our informatics Matura examination and only to point out some characteristics of such tasks. In the practical part of the examination, the students are given some data in text files. The structure of the data, or the conditions of the task, suggest the definition of tables in database. In some tasks, students are expected to design the tables and relation among them according to given data. The students should be able to import data from text file to the database system and than to prepare some queries and finally to export the results back to text files. It depends on a task if the examiner has to check only the results, or also the queries.

After studying the above examples of tasks, we found that the British candidate, who decides to pass computing A-level examination, should demonstrate broader, more general knowledge and understanding of the discipline. In our informatics Matura examination we found more tasks in which the students are expected to demonstrate efficiency in solving simple problems and the ability to think critically.

6 Examiners' Remarks

We have had access to the reports on the examination from two sessions (May 2002 and May 2005) in Poland [9] and we read the report from AQA examination board from the January 2005 [6]. The general remarks are similar. In both countries, the examiners have to comment on the need to read the questions carefully. In the British report, is also stated that "candidates should take more care in expressing themselves clearly in order to ensure they obtain full credit for their answers" and "it is expected that candidates will use the correct technical terms at this level."

Another problem reported by AQA examination board is insufficient exposure to

programming techniques presented by many candidates. The examiners stated that candidates need more programming experience. On the other hand in Poland, the main difficulty was in insufficient preparation to specify problems and to formulate algorithms. Many candidates, in both countries, fail if the task has required analysing the written algorithm to find its result, or to identify the purpose of it.

Candidate, who seems to have had practical experience of using different programming languages did well in the majority of questions in unit 5 of AQA examination, also in the questions concerned relational databases. In Poland, teachers were advised to take care about doing more tasks in which the students have to analyse real problems.

7 Conclusion

Coming back to the questions asked in the article from previous ISSEP conference [1], we agree that it is difficult to prove if passing computing A-level examination or the informatics Matura examination will be useful for later performance in computing course at the university level. We understand also that universities are careful in setting their entrance requirements based on informatics maturity examination results. But the question is, whether the name *General Certificate in Education* (connected with the A-level examination) or the name *Matura examination* (the name originates the word *maturation* – the process of becoming completely developed mentally or emotionally) is likely to describe only a person ready to take up university studying. In Poland, the work done to establish the informatics Matura examination played, and still is playing, a significant role in the discussion about informatics education at secondary level. The need for preparing students for this examination has influenced teachers' consciousness and caused the reflection on their own qualifications and teaching methods. In the future, the value of this examination may change, but for today it could be considered as a useful definition for the secondary curriculum, and as a model for informatics education, generally.

References

1. Clark, M., Boyle, R.: The Transition from School to University: Would Prior Study of Computing Help? In: From Computer Literacy to Informatics Fundamentals: International Conference on Informatics in Secondary Schools – Evolution and Perspectives, ISSEP 2005, Klagenfurt, Austria, March 30 – April 1, 2005, Proceedings, Lecture Notes in Computer Science, Volume Nr. 3422. Springer-Verlag, Berlin Heidelberg (2005) 46–52
2. Subject criteria for computing AS/A level, [February, 2006] http://www.qca.org.uk/3111_2375.html
3. A level exemplification and performance descriptions: computing, [February, 2006] http://www.qca.org.uk/3111_1156.html
4. AQA GCE A/AS Computing Specifications, [February, 2006] http://www.aqa.org.uk/qual/gceasa/comp.html
5. AQA GCE A/AS Computing Assessment Materials, [February, 2006] http://www.aqa.org.uk/qual/gceasa/comp_assess.html

6. AQA GCE A/AS Computing Examiners' Reports, [February, 2006] http://www.aqa.org.uk/qual/gceasa/comp_exam.html
7. Informatics Matura examination – Specification, [February, 2006] http://www.cke.edu.pl/podstrony/inform_matur/informatyka.pdf
8. Informatics Matura examination – Assessment Materials, [February, 2006] http://www.cke.edu.pl/podstrony/egzaminy/m_arkusze.html
9. Informatics Matura examination – Examiners' Reports, [February, 2006] http://www.cke.edu.pl/images/stories/Wyniki/mat_przyrodniczy.pdf