Teacher Pioneers in the Introduction of Computing Technology in the Swedish Upper Secondary School

Lennart Rolandsson

Department of Education in Arts and Professions
Stockholm University, SE-106 91 Stockholm, Sweden
lennart.rolandsson@mnd.su.se

Abstract. The paper elaborates on programming and computing 1970 to 1983 in the Swedish upper secondary school. Articles from contemporary journals and firsthand text sources are used. During that period, programming never qualified as a separate subject; it was a tool for problem solving in engineering, economics, and mathematics, and did become a literacy subject in its own right. Early adopters of computers in education became important pioneers in the production of discourses about "how-to" and "what-to" teach. The diffusion of computer technology was substantiated in a curriculum for computing in natural sciences, which embraced programming.

Keywords: Curriculum development, education in computing, programming, system architecture, teacher, upper secondary school

1 Introduction

The paper describes the period from 1970 to 1983 when computing was under considerable elaboration in experimental works at schools. Computer technology had been in universities and Swedish industry since the 1940s, but the 1970s was the decade of important reforms when the gymnasium (upper secondary school) merged with vocational education into one school system. Three perspectives describe this historical movement: 1) the pedagogy offered by hardware and software, 2) the experimental works initiated by the National Board of Education, NBE (Skolöverstyrelsen) in some of the major projects concerning the adoption of computing into education, and 3) the voices of teacher pioneers concerning programming.

This paper is part of my thesis project that explores programming as a subject topic experienced by teachers in upper secondary school. The perspective of computing history is important to appreciate better the constraints that technology offers.

2 Hardware and Software Constraints in Education

The learning processes of students when constructing software have always been dependent on features in peripheral and internal hardware. Students could consider themselves lucky if they did not have to load the computers, by hand for half an hour, with necessary software before beginning the work of programming.

In the fall of 1979, NBE made an inventory of different computer technologies used in schools [1]. Fig. 1 displays the computer hardware shifts in upper secondary school during the 1970s. The figure reveals a huge investment from 1977 to 1979 when many schools purchased single unit computers.[1]

According to the DIS-report, the investment for five working places with a connection to a mainframe computer or minicomputer would be approximately 25–35 million SEK [1]. The prices of single unit computers were much more affordable at the time. However, technological development was rapid, which possibly led to some caution with regard to investments. According to NBE's enquiry in 1979, more than 50 percent of the schools had only one or two single unit computers and 88 percent of the schools had at most five single unit computers. In other words, schools were cautious.

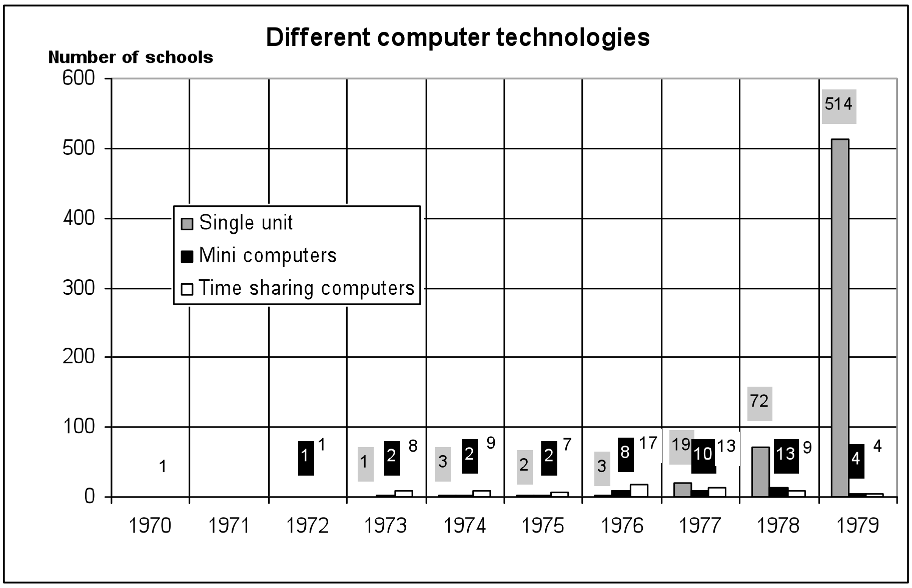

Fig. 1. How computer hardware changed in upper secondary school during the 1970s [1].

[1] Commodore PET came on the market 1977, Apple II in June 1977, TRS 80 in December 1977, ABC 80 in August 1978, and Compis in 1984.

2.1 Mainframe Computer

In the spring of 1970, a programming project called "Zimmermanska skolan," together with ASEA Education AB, began at an upper secondary school in Västerås, Sweden [2]. The working group consisted of three part-time teachers plus a full-time teacher assistant who worked together with students from the technical upper secondary school. Together they developed courses in Automatic Data Processing (ADP) combined with the regular mathematical education for natural science students. At their disposal, they had a mainframe computer, DEC PDP-8 8k/word, which was equipped with disk memory 32 k/word, seven terminals, and eleven monitors. The educational method fell into one of the following two categories:

1. Classroom teaching where the teacher used a monitor to demonstrate, for example, the parabola of moving objects in physics;
2. Students working in groups of four at typing machine terminals and monitors connected to a common computer in the classroom.

The documentation reveals that it took about ten hours of students' work before they could construct their own programs in some applied subject.

2.2 Desktop Calculator

In Sunnerboskolan, teachers had their first experience of computing in 1970 when they used terminals connected via modems to companies such as IBM, Honeywell Bull, and Datema [3]. The educational setting was expensive and constrained to special times during the day and year. Due to the complex connecting procedures, the education consisted primarily of teacher led demonstrations. At first, teaching focused on programming in FORTRAN, and later in BASIC. The shift was based on the higher threshold for learning FORTRAN.

Due to economic constraints, teachers decided to use the desktop calculator, Compucorp 122E, which offered a higher threshold because of the lack of any advanced level language. The calculator, however, enabled more flexible classroom activities, since students could work more interactively with their algorithms. The students prepared the executable programs in assembly language, which were translated to code on card(s) that were loaded into the calculator later. They then had to wait in a line before they could see the results of their work. If students discovered errors in the code, they had to check their algorithms and repeat the process.

2.3 Minicomputer

Sunnerboskolan was also one of the very first schools that purchased a minicomputer in 1973, the PDP 8, with three teletype devices [3]. The computer's construction meant that you had to load the machine for twenty minutes with a BASIC compiler before one could use it in class. The connection between the teletypes and the minicomputer was slow; the teletypes printed ten characters per second. Therefore, in 1975, the school decided to buy a minicomputer, the Alpha LSI, with a magnetic secondary memory, which resolved many of the tedious preparations. According to documentation, they purchased their third minicomputer in 1978, the Nord 10, which had a much bigger primary memory and a faster response rate with its terminals. The monitors responded one hundred times faster than the teletypes could ever print. In conclusion, the educational setting for computing transformed considerably in a period of five years.

3 The Introduction of Programming in Upper Secondary School

The first secondary schools to offer computing courses in the 1960s were located in one of the following areas: Stockholm (Fridhemsplan), Västerås, Gothenburg, and Malmö. Computing was offered as advanced special courses at upper secondary school; as a student, you could choose between one-year specializations in administrative ADP or technical ADP [4]. Due to its success and to meet industry standards at the time, these first schools were soon followed by eleven others [5] located in Solna, Växjö, Västervik, Norrköping, Linköping, Örebro, Karlstad, Uppsala, Gävle, Umeå, and Sundsvall [2]. The courses reached a peak or "programming boom" at the end of the 1960s with thirty-five to forty classes [5] all over Sweden. The programming languages offered were Assembler, FORTRAN, PL/I, Simula, Basic, Simscript, Cobol, and script languages [6, 7].

In 1977, reorganization from NBE to National Board of Universities and Colleges (Universitets- och högskoleämbetet, UHÄ) meant that many schools were organized under higher education and municipality colleges. However, three schools included a local upper secondary school in the following cities: Umeå, Uppsala, and Linköping [5].

In 1980, collaboration occurred between one college and Berzeliusskolan in Linköping where they carried out the education for the natural sciences computer alignment (from now on named CANS) and evening courses for the municipal adult education [5]. In Malmö, a municipal college offered courses for teachers at upper secondary level and a college in Växjö offered courses for the municipal adult education.

3.1 Early Adopters of Programming in Technical Upper Secondary Schools

The result of an enquiry in 1970 initiated by NBE identified fourteen different technical upper secondary schools as users of computers for scrutinizing laboratory data [2]. A report from 1973 reveals the existence of local experimental work carried out at nine different upper secondary schools [8]. The educational content differed in some aspects between the schools, but the main intention was to enhance calculation and problem solving in mathematics and applied subjects. According to the report, education allocated to programming varied between the schools from a few to fifty-one hours, in languages like BASIC, FORTRAN and COBOL. Furthermore, the DIS-report reveals that some schools had received special support from NBE for investments in computer technology since 1973: Erik Dahlbergsskolan in Jönköping, Berzeliusskolan in Linköping[2], Sunnerboskolan in Ljungby, and Zimmermanska skolan in Västerås [1, 8].

Another report from NBE [9] reveals that students of electrical engineering were taught how to program simpler tasks (mainly in BASIC), by the mathematics teacher. Students would acquire procedural knowledge about computing and programming to the extent that:

> … [students] themselves sometimes have to code and sometimes they have to use established programs. It is important that the computer interaction is not time consuming and that students do not perceive computer interaction as an unusually complex way of solving problems [9].

The report from 1976, DIS Bygg, explicitly indicates how teachers of the building/construction course perceived the necessity of programming in education [10]. Some excerpts from these teachers' opinions reveal that there was a tendency to work more with established programs:

> … only one of three building/construction teachers had spent much energy in creating programs … If the school had a computer park with BASIC-compilers and enough terminals, students with interest could probably create their own programs, and in that way also learn a diversity of calculating methods [10].

The role of computers was a calculation tool used for statistics, printing graphs, or sorting data, which would infer programming work. According to a report, the time allocated to the objectives of learning computing, with a major focus on programming, was a suggested two hundred hours distributed over four years at upper secondary school [11].

3.2 A Broad Initiative with Many Projects

During the 1970s, the NBE expressed a sincere ambition to draw up a curriculum for developing computing literacy among students in secondary school. They realized early in the process that computer technology would become a pervasive technology with a huge impact on the whole of society. NBE therefore had to reach a common understanding of how to organize and standardize computing education in secondary school. Some countries considered the importance of a new subject [1] with advanced technical content, while Sweden and NBE, based on recommendations from the DISK report [8], decided to adopt their own implementation strategy, as quickly as possible, in different subjects while experimental work would determine best practices. Subsequently, they would introduce a new subject called "Datalära."

NBE formed a working group in 1974, "Computers in school," a DIS[3] group that based its work on the assumption that computing is better taught as non-specialized

[2] During that time the head master of Berzeliusskolan was also one of those responsible for the report "Computers in the school municipality," DISK report (translated from "Datorn I SkolKommunen").

knowledge with the potential to facilitate and enhance conceptual understanding in other subjects like mathematics, natural, and social sciences.

3.3 The Computing Alignment in Natural Science Program

Meanwhile, in 1976 [12], experimental work for eight to nine hours per week started in the Natural Science Program with a computer alignment (CANS). The intention was to "increase the number of students in natural science for the purpose of making the Natural Science Program more practical with regard to connections to society as a whole" [13]. The intention described in 1981 was:

> …not intended as a difficult alternative… Hopefully students will experience learning about computing in harmony, partly by doing their own programs, mainly in BASIC, and partly by using developed programs in different subject domains. In addition, it will lead to a better understanding of computers in society. The overall objective of the work is to expose the natural scientist to the use of computers, instead of it being regarded as an education for computer specialists [14].

This was one of four different alignments in the Natural Science Program. The other three were in energy, healthcare, and environment, of which computing was considered the most popular. One of the persons in charge of the implementation of CANS describes the process in the following way:

> The experiences so far are good, with a positive attitude from students and teachers. Because of the Natural Science Program and the [computing] alignment's future, it will be very important that education in the subject does not put too much pressure on students. One of the major concerns with the alignment is to give students more time for natural science… [15].

At a conference in 1981 [16], teachers involved with computing alignment shared their experiences while emphasizing special concern about the importance of structure and sound habits when students use system architecture in programming. The language COMAL (a dialect of BASIC) was considered suitable for fostering sound habits because of its implicit structure.

Based on work in the DIS-project [1], NBE devised a new curriculum, in 1983 [17], for Computer Science or "Datakunskap," which became the course subject and guideline for the computing alignment in the Natural Science Program. It was explicitly drawn up with 190 to 240 hours [18] of computing. The curriculum was extraordinary because of its descriptive texts to facilitate the teaching of computer science. Teachers in favor were mainly from the mathematics domain [19]:

> Software development in a methodological sense is a relatively new knowledge domain undergoing huge change. The subject is not part of any teacher education

---

[3] Acronym for "Datorn I Skolan."

[for upper secondary school]. Due to this, extensive commentaries about the module [concerning program development and programming] have been included to clarify and describe a strict educational design and overview of the subject for inexperienced teachers [17].

The commentaries, divided into three levels, are worth reading. They consist of features of the language, elaboration about the programming concepts, and a major work project; a pattern still found today in programming courses A, B and C in upper secondary school. The curriculum description contained twenty-five pages, divided into five different modules:

1. Computer system
2. Program development and programming
3. Use of computers in natural sciences
4. Use of computers in social sciences
5. System technology

A clear statement about the purpose of studying “Datakunskap” is made at the beginning of the document [17]:

> … the education should be designed to enhance the development of students’ ability to work with computers as users … The intention is not to educate people for the programming industry,” meanwhile one of the objectives for the course is to “develop students knowledge in problem analysis and programming.”

One of the first schools carrying out experimental work with CANS was Berzeliusskolan in Linköping. Bandhagen upper secondary in Stockholm started in autumn 1977 [19] and was followed in autumn 1978 by eleven other schools.

Table 1. Number of schools carrying out experimental work within the Natural Science Program (CANS) [14, 15, 19].

	Number of schools within CANS	**Number of students**
1978	11	
1980	20	
1981	35	500 students in year two
1983	47	1,956 students in year two, 1,029 students in year three
1985		2,345 students

3.4 Step-wise Integration of Programming into Mathematics

Some teachers perceived computing as different from the rest of mathematics and sometimes too advanced for the ordinary mathematics curriculum. They state:

> The change of curriculum [with numerical methods] could drastically influence mathematics education, mainly because of the implicit demands of the new way of thinking, which above all shall be integrated with the other chapters [in the literature] …teachers with not enough confidence dare not experiment with technical facilities and therefore [numerical methods] become another example of a theorized module, practiced according to the student literature mainly before central assignments [20].

In 1976–77, twenty classes were doing experimental work within mathematics [21] where step-wise iterations were introduced for solving differential equations which were later enlarged with numerical equations, numerical integration, simulation, and applications in physics, all under the name of the NUMA[4] project [12]. The material originated from a group of teachers at Sunnerboskolan in Ljungby. The overall intention of the NUMA project was to study how practical mathematics could be established in a school context with computers [22]. The following excerpt describes some of the ideas that surrounded their pedagogy:

> The 1980s is the decade for problem solving in mathematics … In school we are trying different strategies for problem solving. Computers and calculators foremost affect the trial and error methods and the simulation methods. With a computer as a tool, students could experience how mathematics could be used experimentally to solve different problems. Computers in the proper hands open up possibilities for the enhancement of creative thinking [12].

Teachers in general considered programming as synonymous with tedious work that took time away from the original subject, mathematics [13]. The dilemma between time-consuming technicalities in programming and increased conceptual understanding in mathematics is obvious. The author of the previous excerpt states:

> If you intend to really do programming you should do it with a broader perspective; formulate your problem, pick one of many solution methods, code it in a language, try the algorithm and document your solution [12].

Another excerpt reveals the importance of a sound and social connection when implementing programming in the school context:

> We have to aim for a broader perspective of computer technology that goes beyond the common dialectic view of either the public fear of hostile technical database registries or the computer technician's light blue optimism. It is also

---

[4] NUMA is an acronym for "Numerisk matematik."

important that a general broader perspective even includes hackers (compulsion programming) among our students and teachers [12].

4 Analysis and Discussion

In the development of the computing curriculum, two intersecting objectives have emerged [8]: 1) the implementation of computing as a non-specialist subject offering literacy, and 2) computing as a tool to facilitate the education in applied subjects. The paper has exposed the existence of programming, during the time span, in different experimental works; in advanced mathematical courses (numerical methods and algorithm construction), in applied subjects for engineering in upper secondary school, and as a module in computing alignment for the Natural Science Program.

The paper also reveals the tension between the features inherent in computing devices and the intent to teach programming in an educational setting. Furthermore, the paper presents the existence of teacher pioneers who succeeded in merging technology and pedagogy to such an extent that their work became exemplary for NBE in the development of the computing curricula. Programming, however, does not appear as a subject itself, during this period, which could be understood from the contextualized view and the programming perspective as a natural tool in problem solving [24]. Nevertheless, one of the teacher pioneers explains the beauty of the conceptual understanding of loops[5], when he writes:

> The class have conquered the computer with a simple BASIC program and forced it to deliver the true answer with few loops! Is not that a miracle in itself? [12]

Finally, I would like to pose the question concerning advanced special courses in upper secondary school within ADP. Why did the legacy from these courses not have a greater influence in the development of computing education? They obviously used the same resources for several years.

References

1. Skolöverstyrelsen: Datorn i skolan. SÖ:s handlingsprogram och slutrapport SÖ-projekt 628, pp. 8–9, 13, 51–56. Stockholm (1980)
2. Andersson, Å.: Datorn i skolan (DIS): förstudie 1. Kungl. Skolöverstyrelsen, pp. 4, 14. Stockholm (1970)
3. Emanuelsson, G., Nilsson, R.: En gymnasieskola med lång erfarenhet. In: Att köpa datorer, pp. 89–91. Liber Utbildningsförlag, Stockholm (1984)
4. Skolöverstyrelsen: Aktuellt från Skolöverstyrelsen, number 68, p. 3 (1972/73)
5. ADB-lärarföreningen: Alfanumeriska meddelanden 6, no. 1, pp. 3–11 (1980). In private collection
6. Skolöverstyrelsen: Aktuellt från Skolöverstyrelsen, no. 42, pp. 24–25 (1976/77)
7. Skolöverstyrelsen: Aktuellt från Skolöverstyrelsen, no. 5, p. 3 (1975/76)

[5] A learning object in programming.

8. Fagerström, P.: Datorn i skolkommunen (DISK): Kartläggning, beskrivning: Förslag till åtgärd, p. iv, Appendix 3. Linköping (1973)
9. Skolöverstyrelsen: Datoranvändning i eltekniska ämnen (DISEL). In: National Archives, Arninge, Sweden. Dnr L 77:809, p. 7 (1977-09-01)
10. Skolöverstyrelsen: Datorns användning i byggtekniska ämnen: Delrapport (DIS Bygg). In: National Archives, Arninge, Sweden. Dnr L 76:1897 (1976-08-13)
11 PRODIS report. In: National Archives, Arninge Dnr L 81:530 (1981-06-16)
12. Björk, L.-E.: Datorns intåg i svenska skolan. J. Nämnaren 1, pp. 32–33 (1983)
13. Wendelöv, L.: Fortbildning i datalära Mer tid för matematik. Remissvar på kursplaneförslag för NT-linjerna. J. Nämnaren 3, pp. 7–11 (1980)
14. Borg, K.: N-linjens datatekniska variant. J. Nämnaren 2, p. 55 (1981)
15. Wendelöv, L.: Gymnasielinjernas matematik. J. Nämnaren 1, p. 7 (1981)
16. Wendelöv, L.: Gymnasielinjernas matematik. J. Nämnaren 4, p. 7 (1981)
17. Skolöverstyrelsen: Läroplan för gymnasieskolan, Supplement, 95, Datakunskap. Liber Utbildningsförlag, Stockholm (1983)
18. Wettstam, L.: Datakunskap. J. Nämnaren 2, p. 12 (1984)
19. Borg, K.: Varierande undervisning på datateknisk variant. J. Nämnaren 4, p. 60 (1981)
20. Schreiber, B., Schreiber, A.: Vad blev det av NUMA? J. Nämnaren 2, pp. 41–42 (1984)
21. Vejde, O.: Några fakta om NUMA försöket. J. Nämnaren 2, p. 40 (1984)
22. Borg, K.: Några terminer vid en minidator. J. Nämnaren T, pp. 122–124 (1976)
23. Björk, L.-E.: ADM-projektet. Några synpunkter på gymnasieskolans datakunskap. Rapport 27, pp. 13–15. Uppsala (1987)