A Contribution to the Discussion on Informatics and Robotics in Secondary Schools

Alberto Barbero[#1], Barbara Demo[*2], Francesco Vaschetto[#3]

[#] *Istituto Tecnico Superiore "G. Vallauri"*
via San Michele 68, Fossano (CN) – Italy
[1]barbero@Vallauri.edu
[3]vaschetto@Vallauri.edu
[*] *Dip. Informatica, University of Torino*
c.so Svizzera 185 - 10149 Torino – Italy
[2]barbara@di.unito.it

Abstract—**The approach to Informatics in Italian secondary schools is being reshaped after the Reform effective from autumn 2010. Educational robotics can contribute to this reshaping because it allows students acquire a technological competence in Informatics as recommended by the Computer Science teachers and researchers in universities with the "Manifesto for Informatics in secondary school" issued in May 2010. During the school year 2010-2011, first year students of a technical institute, i.e. students about fourteen years old, have developed programming activities using Scratch and S4A, Scratch for Arduino. This experience is proposed as a reference activity for a new Informatics curriculum in technical schools and, integrated with other components, to all types of secondary schools.**

***Keywords*—Educational robotics, Pragmatic robotics, Manifesto for Informatics, Scratch, Arduino, S4A.**

I. Introduction

Several robotics activities are carried out in k-12 education. For primary and middle school students, reports generally describe typical educational robot activities, [1], [2], while also more specialised and pragmatic robot activities are offered to older students. Indeed, the aim of vocational and partly of engineering (or technical) secondary schools is directing students toward their working life thus they shall offer pragmatic robotics experiences supported by robotics industries also. In Torino (Piedmont, Italy) we have headquarters of COMAU, the Italian industry among world leaders in industrial robotics, and in the area several other companies are active in the same field. A recent market research by Piedmont Region and other local administrations points out that these industries will offer a relevant number of specialized technical jobs in the next years [3]. According to this market research these future jobs will concern industrial robotics but many of them, even most of them, will be in the so called service robotics where we have domotics and medical robotics, to name two of the best known areas. These areas are still in a creative and innovative phase. Hence robotics in secondary schools naturally has and will continue to have a double-face presence: the pragmatic robotics, introducing to the robotics industries above, and the educational robotics such as the one cultivated in projects like Terecop and Roberta [4],[1].

Multidisciplinarity, problem solving, programming and Informatics in general are peculiar components of educational robotics activities. Indeed, in primary, middle and non vocational secondary schools, robotics experiences described in the literature are so strictly interwoven with Informatics that we can see Informatics and robotics as joined together. Thus, for an analysis of robotics in schools we suggest considering the "Manifesto for Informatics in secondary schools", published in May 2010 by the main Computer Science academic associations in Italy [5]. The Manifesto points out that in today society people have quite different perceptions of what Computer science is. One is the pragmatic perception of people that consider Informatics to be a set of hardware and software systems, the second is the technical perception shared by people that conceive Informatics as a set of technologies to be used to implement system and applicative packages. Third is the cultural perception conceiving Informatics as the scientific discipline founding computer technology.

The Manifesto analysis fits robotics at least for the pragmatic and the technological perceptions. Indeed the cultural component in the case of robotics likely concerns the several disciplines converging into robotics and deserves a different discussion. But when we mostly consider computer components of robotics we must take into account also the third facte concerning the scientific aspects of Informatics necessary to all students because shall allow them to be conscious or at least to have a general idea of the possibilities (and likely of the limits to the possibilities) of the automated tools they will deal with or will design in their future life. This is particularly true for service robotics that appears to be the best promising field as for jobs offer. Obviously an education considering all three perspectives will provide different levels of competence in each component depending on the school type as it is for other disciplines.

In Session 2, we sketch the Manifesto and its claim for extending students vision to perceive all different conceptions of Informatics in order to acquire a computational thinking mindset [6].

In Italy educators have the chance to define curricula inflected according to the above remarks also because of the secondary school Reform effective from autumn 2010. This Reform has reshaped all secondary school types. There is a larger presence of computer science related subjects: for example Informatics appears both in technical institutes and in Applied Sciences lyceum type since the first year. Previously they were explicitly present from the third year only. The Reform does not rigidly define the approach to Informatics and, consequently, schools must give it a new shape.

Indeed, for all subjects, the Reform only defines "guidelines" for each school year in every school type leaving to teachers and to single schools defining their education paths with an autonomy and responsibility level higher than with previous school curricula, precisely defined. This autonomy is both an opportunity and a possible critical point for Computer science related disciplines since they do not have a tradition concerning contents and teaching methodologies teachers may refer to or feel sort of mandatory referring to. Besides, due to reasons not to be discussed here, in Italy, teachers of Informatics related disciplines may have many specializations different from Informatics (and these teachers are not going to soon retire). Aware of the chance of proposing new approaches to computer science, several groups of Informatics teachers in schools and researchers from universities are working together in order to design (part of) curricula implementing the Manifesto.

In Session 3 an example is sketched of the activities proposed by one of these groups. The activities have been carried out with first year students during the 2010-2011 school-year at the Technical Institute Vallauri in Fossano (Cuneo), Piedmont, Italy, where two of the authors are teaching. The students are introduced to programming using the Scratch language and its extension S4A (Scratch for Arduino) when the open hardware Arduino is to be used. In these activities multidisciplinary aspects are present that allow students to experience concepts introduced in different subjects of their normal curricula. Teachers also considered that at the end of the first two years in a polytechnic secondary school Italian students must choose among Informatics, Electronics, Mechanics and others, the specialization characterizing their final three years. The curriculum of the beginning two years is thus crucial because, obviously, the choice for the last three years is mostly based on first experiences. Scratch and Arduino activities combine Informatics and robotics allowing students to come into contact with several of the disciplines offered to them as a specialization from the third year. Besides, also for students that shall quit the secondary school at the end of the compulsory beginning two years, it is important they are introduced to software and hardware technologies they will be confronted with in their future life.

The Informatics curriculum sketched in this paper can be proposed also to students in different types of secondary schools integrated with other disciplines for example with philosophy, logics and science history in the lyceum type of school. The conclusion is that we must consider proposing a blend, appropriate to the school-type, of pragmatism with knowledge of the technological components of a robot system and of their scientific foundations.

II. The Manifesto for Informatics in secondary schools

In Italy there are three main national associations of computer scientists: the National Consortium for Informatics Inter-universities (CINI), the Group of Informatics Engineers (GII) and the Group of researchers in Information Science from Italian universities (GRIN). While the Education Ministry was developing the Secondary School Reform, these associations contributed to the discussions in different ways. A common contribution is the "Manifesto for Informatics in secondary school" [5].

The Manifesto points out that in Italian schools Informatics is almost only present as learning technologies to implement some software in technical schools of some specialised type or, most generally, as practicing specialised software such as Office and Open Office, GeoGebra or other such softwares certainly of big help in learning Mathematics or other subjects. In the Manifesto we read that:

"Informatics is becoming the kernel of our modern world both because it is needed to the normal development of our everyday duties and because its development shapes and directs the advancement of our whole society.

Nowadays, in all areas of human activities we can find the influences of digital discoveries and achievements. Indeed, the computer is no more used for the traditional scientific calculus only, yet it is used in all areas of industrial production, medicine, publishing and communication to name only some of its applications. Two billions of people have at least one contact on the net each day. We have around us products full with hundreds of millions of billions (no typos here) of transistors– elementary hardware components supporting information technology – in our cars, in domestic appliances, inside the gas pumps, in our videogames, and they are half of the financial value of the products. Hundreds of billions of software instructions, expressions of human intelligences, give life to these components and, through them, to all processes peculiar to our modern society.

In our everyday language the expression Computer Science or Informatics refers to three different perceptions related but quite different. A person can have a

1. Pragmatic perception of Informatics and see it as a set of hardware and software systems.
2. Technological perception and conceive Informatics as a set of technological tools to be used to implement system and applicative packages.
3. Cultural perception and see Informatics as a scientific discipline founding, thus making possible, computer science technology.

The common man has the first perception of Informatics and sees it as a set of applications. Thus in his point of view knowing computer science means knowing how to use software packages and what digital devices it is reasonable to buy. On the contrary, for technicians, knowing Informatics means to know how to develop software systems."

What must be like the Informatics discipline in schools is a largely debated question. Likely there are different answers depending on the point of view one looks at computer science and the above three different interpretations inspire different ways of dealing with Informatics in schools. Different types of secondary schools can plausibly have different aims in making their students competent in Informatics and consequently they may decide to stress one of the above perspectives most. For a cultural perspective of Informatics we should mainly address its epistemological aspects and focus on its connections with Mathematics, in particular Logics, with Philosophy and History: this is particularly suggested to Italian classical, scientific and pedagogical lyceum type of schools. But the Manifesto concludes that, unfortunately, Informatics as a science is missing in almost all of our schools while it should be an aspect of digital literacy possessed in some form by everyone at the end of any type of secondary school.

III. PROGRAMMING WITH A CAT AND A KING AS COMPANIONS

Due to the very different levels of students familiarity with computers and computer science, in Italian technical schools the guidelines for the first year include the European Computer Driving License (ECDL) syllabus [7]. It turns out that most of the schools find appropriate dedicating to the ECDL certification the entire first year in order to ensure a basic common level of digital competencies. But, as we said, in the reformed school teachers have the responsibility of defining a curriculum from their class guidelines. At the Vallauri Technical Institute in Fossano, Italy, in school-year 2010-2011, teachers decided to advance to the first year part of the competencies of the guidelines for the third year in order to introduce students to algorithms and to programming from the beginning. They aimed at gaining the interest of students entering the secondary school by proposing them computer science motivating activities and at the same time allowing students to acquire abilities required by the first year Reform guidelines. Also, teachers considered important that students could have a concrete grasp on what it is going on from the formal specification of a program to its effects on something concrete, with similar motivations as those introduced in [8]. For this they chose to also propose some experiences with the Arduino board.

A. *Introduction to programming*

Teachers decided to look for a programming language different from Visual Basic, C, C++ or Java, scheduled to be used in last years of secondary schools and in universities, judged too difficult and requiring too much time to obtain motivating results. Scratch is a visual programming language due to Mitchell Resnick and his Lifelong Kindergarten Group of the M.I.T. MediaLab in Boston [9]. Scratch and its open source integrated development environment (IDE) suit the curriculum teachers were thinking of because it was specifically created for introducing basic concepts of problem solving and programming to very young or inexperienced students. In Scratch, programmers can easily implement animations, simultaneously execute different processes and make them interact, use events. The IDE provides a visual block editor where a program is specified by dragging the chosen icons or blocks, each corresponding to a primitive written on the block, having a different colour and shape depending on its function. The blocks have jigsaw puzzle shapes in order to limit language components that can go together. The choice of the Scratch language and its IDE is motivated and detailed in [10].

For sake of space here we describe only the activities around the conversion of a decimal number into a binary notation because it can be used for showing both aspects of the programming and robotics activities considered here. Students worked on different number notations during Informatics and Mathematics hours discussing about numbers, digits, fractions and algorithms to convert a number from one notation to another. After students worked out the mathematical aspects of the problem, each group was asked to specify a conversion algorithm from decimal to binary. The Scratch stage of one of the programming activities is shown in figure 1. The stage is the "window" where sprites are shown. In figure 1 we have the following sprites: the cat Garfield, sprites corresponding each to a bit and the Arduino sprite, not used in the first steps of the conversion activity.

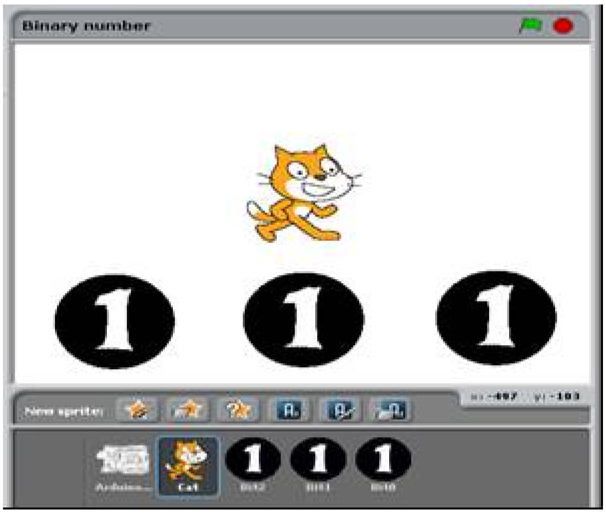

Fig. 1 The Scratch stage for converting from decimal into binary

Indeed each group of students worked on its own solution of the given problem. After each group had found an algorithm and implemented it, students were introduced to comparing different solutions of the same problem and to understand that a solution of a problem by means of an algorithm and a program has properties such as complexity and generality. Thus, though this experience is quite elementary, students could discover at an introductory level properties like complexity and generality by comparing their algorithms.

The Scratch experience has been more than a programming experience: students went from operative system features such as components and standard configurations to writing reports or presentations of their different solutions. Teachers verified that through the programming activities students also achieved most of the competences required by the first grade guidelines and were enabled to pass ECDL-like tests that, as we wrote above, in technical secondary schools are often offered at the end of the first or of the second year.

B. S4A: Scratch for Arduino

The binary notation for numbers also has different Arduino versions. Arduino is a well known open board easy to enrich with sensors and actuators suitable to the interactive environments where we want to use it [11]. The interest on Arduino is because, as we said, we consider important that students experience concrete programming, that they can see the immediate result of their code and verify what it is going on from the formal specification of a program to its effects on something “touchable”, in this case Arduino. For Piedmont people Arduino board has a special meaning because it was conceived in Ivrea (Piedmont, Italy), the legend says at the Arduino bar of the Arduino main street. We had an Arduino (955–1015) Margrave of Ivrea and King of Italy. In Ivrea the Olivetti Company was based that is the company many elderly Italians owe for different, not only technological, reasons.

Fig. 2 The S4A code sequence for switching off & on “bits” on Arduino

We consider programming Arduino using the S4A (Scratch for Arduino) environment developed by Citilab in Barcellona (Spain) [12]. S4A is an open Scratch extension supporting simple programming of Arduino. It extends the Scratch IDE with new blocks for programming the sprite Arduino having sensors and actuators connected to the real board. A firmware was developed that, by means of the Picoboard protocol, allows S4A to send commands to the actuators, to get sensors status and to make run the (up to four) engines. S4A and the described firmware are downloadable at http://seaside.citilab.eu/scratch/downloads.

The activity went on through several steps summarized as follows:

a. making a led blink,
b. managing a red-light changing colours either by fixed intervals of time or by touching the sprite corresponding to a color,
c. using activities a and b to show the binary conversion of a decimal number in the range [0,7].

Figure 2 shows the S4A code for switching on and off the three leds connected to Arduino ports number 10, 11 and 13. The resulting binary notation for a decimal number ranging between 0 and 7 is shown on the same stage of Figure 1 and on Arduino using the leds corresponding each to one digit in the binary notation, see figure 3.

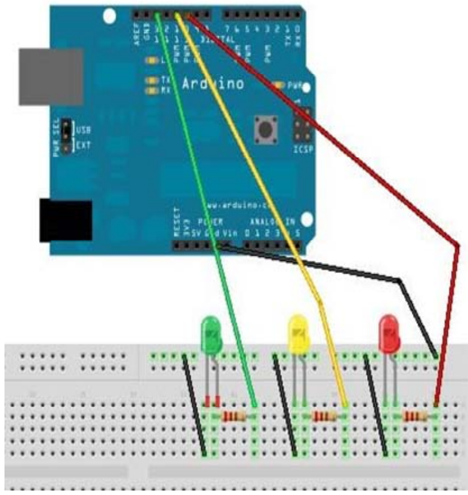

Fig. 3 Arduino and leds used for the red-light activity and for converting into binary

IV. CONCLUSIONS

In approaching robotics activities all types of secondary school shall propose a blend, depending on the school-type, of pragmatic experiences and acquisition of knowledge about the technological components of robot, of their programming and of their scientific foundations. With the enlarged presence of Informatics in the reformed secondary school some schools plan to maintain a mostly pragmatic approach to Informatics and even enrich their offer in teaching how to use packages and automatic systems of several kinds. The pragmatic approach is important to acquire basic competence in dealing with a computer, it is mandatory when we use fantastic applications to experience mathematics, physics, chemistry and the other disciplines. It is necessary for vocational schools where training courses have a relevant presence. In all cases we must consider that the practice is as important as the methodology to acquire practice.

During school professional training on technological tools such as robots we must consider the fast evolution of the technology. The risk is making room for ideas like "abstraction shall not be taught to students" heard in a recent workshop on vocational and technical schools. All types of secondary school adopting this motto are condemning its students to find difficult changing from one system to another, from one robot to another, in general when they have to update their knowledge by using their competences. Also in vocational school the three approaches of the Manifesto must be present.

Here we presented a preliminary contribution for a first year curriculum of technical schools implemented during this 2010/2011, first year of the secondary school Reform in Italy. By beginning new experiences in technical schools, where computer science is not a new presence, we try to define original educational objectives and activities for sort of testing them in environments already introduced to the discipline. The aim is to introduce basic technological concepts of Informatics also in all other types of secondary schools where it has not been present till now as, for example, in so called Lyceum, with cultural integrations, as for logics and philosophy.

With our computer science activities in secondary schools we do not aim at educating all students to become good programmers rather to introduce them to the computational way of thinking that many researchers consider essential for our next generations can take advantage of the computational power they will have at disposal.

Jeannette Wing, the President's Professor of Computer Science and head of the Computer Science Department at Carnegie Mellon, during her presentation in the Computer Science Distinguished Lecture Series at Carnegie Mellon in Qatar, said: "Computational thinking is a fundamental skill used by everyone in the world, and should be incorporated into educational programs along with reading, writing and arithmetic to grow every child's analytical ability" [6].

The example here described is a possible component of a curriculum for the first year of secondary school where Scratch is used for the programming and graphics part of activities and Arduino is used for the control part. We designed it guided by the Manifesto for Informatics and knowing that the relationships between robotics and Informatics particularly in schools allows us to avoid reducing robotics in education to a mechanistic exercise.

ACKNOWLEDGMENT

Many thanks to all our colleagues of the Association DSchola. Thanks also to the researchers of the TERECOP project, particularly to Michele Moro of the University of Padova. Many discussions have been fostered by the unvaluable context of the Meeting "Education and Robots" organized by the Museo Civico of Rovereto (Trento) under the responsibility of Franco Finotti and Nello Fava. Stefano Monfalcon's students activities shown during that meeting are best and inspiring examples of educational robotics: the Depero's puppet moved by two Lego NXT robots was surprising.

REFERENCES

[1] (2004) The Roberta project website. Available: http://www.roberta-home.de/en
[2] M. S. De Michele, B. Demo and S. Siega, "A Piedmont SchoolNet for a K-12 Mini-Robots Programming Project: Experiences in Primary Schools", in *Proc. Terecop Workshop, First SIMPAR Conference*, Nov. 2008.
[3] (2011) The Piedmont Region website. "Competenze e professioni nel futuro del Piemonte". Torino, June, 9th 2011 Available: http://www.regione.piemonte.it/europa/notizie/comp_prof.htm
[4] (2009) The TERECOP project website. Available: http://www.terecop.eu/
[5] (2010) GRIN website, The Manifesto for Informatics in secondary school", http://www.grin-informatica.it:8080/opencms/opencms/grin/infoescuola/riformascuola/
[6] Wing, J., Computational thinking is a fundamental skill, Presentation in the Computer Science Distinguished Lecture Series at Carnegie Mellon in Qatar, April 2011, http://www.cmu.edu/news/blog/2011/Spring/computational-thinking.shtml
[7] ECDL website, Available: http://www.ecdl.com/
[8] B. Demo, G. Marcianò and S. Siega, "Concrete Programming using Small Robots in Primary Schools" in *Proc. ICALT 2007*, p. 124
[9] Resnick, M. and all., Scratch: Programming for All, CACM, Vol. 52, No. 11, pp 60-67, November 2009
[10] A. Barbero and B. Demo, "The Art of Programming in a Technical Institute after the Italian Secondary School Reform," in *Proc. ISSEP 5th, 2011*, (to appear).
[11] Arduino Open hardware Project, www.arduino.cc
[12] S4A - Scracth for Arduino, http://seaside.citilab.eu/scratch/arduino