Need Analysis of the Students in Programming Courses in Latin America

Technical Report · January 2011

DOI: 10.13140/RG.2.2.10373.09445

Xavier Ochoa
New York University

Cristian Cechinel
Federal University of Santa Catarina

Camilo Jiménez
Los Andes University (Colombia)

Erick Araya
Universidad Austral de Chile

Need Analysis of the Students in Programming Courses in Latin America

Xavier Ochoa, Cristian Cechinel, Camilo Jimenez, Carlos Arévalo, Erick Araya, Sandro da Silva Camargo, Cláudia Camerini Perez, Katherine Chiluiza, Luis Alvarez, Jorge Morales

I. Introduction

The low quality of primary and secondary education in most Latin America countries is a well-known problem. As a response to this reality, the private educational market has been steadily growing in those countries. These private schools, in general, offer a higher quality and personalized education for the students that can afford it. The main selling point of these institutions is access to better resources: better teachers, technologies, materials and pedagogical methods. This difference in education quality creates a problem once students from public schools reach university. The public schooled students have a strong handicap in their performance in a demanding and fast pace environment where professors are more concerned with the delivery of knowledge to large audiences than with catering to the specific needs of each student. This problem is aggravated by the fact that the great majority of public schooled students belong to low-income families. All the problems that arise from this social status in Latin America (need to work at an early age, economical difficulties, etc) also conspire to reduce the probabilities of success of these students. In this light, it is not just understandable, but to be expected, that the private schooled students out-performed their public schooled peers and gain better opportunities at the labour market.

The Innovation for Equality in Latin American University (IGUAL) Project is a joint initiative between Latin American and European Universities to improve, through the use of innovative learning solutions, the accessibility of Latin American Higher Education Institutions (HEIs) to students that have received sub-par education during their primary and secondary studies. A pivotal phase of this project is the capture and analysis of the needs of the Latin American HEIs. To make the project manageable, a common course about Introduction to Programming was selected as a pilot for the application and evaluation of the project. The information collected in this Need Analysis will be used to build tailored learning solutions to help disadvantaged students (specially students publicly schooled) to bridge the knowledge and skill gap with more advanced students (usually the ones that received a private schooling).

The structure of this document is as follows: First the data gathering instruments and methodology is presented, together with the analysis procedures. Second, the results of the analysis are presented and discussed. The work closes with a brief discussion about the implication of the findings.

II. Data Collection

A. Course Selection and Description

From the seven members of the IGUAL project, all the Latin American partners (five of them) collected information about their Introduction to Programming course: ESPOL (Ecuador), UAA (México), UACh (Chile) UNIANDES (Colombia) y UNIPAMPA (Brazil). The Introduction to Programming courses of these universities provide a good sample of this kind of courses in other Latin American Universities. However, each university has its particularities that will be explained in the following sub sections:

1) Escuela Superior Politécnica del Litoral (ESPOL): In ESPOL, the Programming Fundamentals course provides an introduction to the concept of Programming in the C language and it should be the first programming course taken by students. This course is mandatory for all the students of Computer Science, Telecommunications and Telematics majors. Students from other majors can take the course as optional credits. Currently, 270 students are taking this course, divided in nine groups. Historically, this course present a large failure rate (>50%), a reason why it was selected as the impact of the intervention could be easily measured.

2) Universidad Autónoma de Aguascalientes (UAA): Introductory programming is taught In UAA in two semesters (see Figure 1). During the first semester, students are given basic problem solving and diagraming skills. The main objective of the first course is to develop the student's 'programming logic' mainly using Flow Chart diagrams and pseudo-code. Problem solving skills are expected to be developed in this course. Basic programming concepts such as data types, constants, variables, expressions and control structures are also taught. The second course (Programming I) focuses on teaching structured programming using C language. More specific and advanced topics such as functions, data structures, recursion, pointers and binary archives are taught. Historically (UAA 2007), the failure rate of Programming I has been high: around 40%.

3) Universidad Austral de Chile (UACh): The Civil Engineering Computer Science program from the Universidad Austral de Chile has in its curriculum the course "Introducción a la Informatica" (Introduction to Informatics) designed for students newly admitted to the program. This is the basic course that was selected for analysis.

4) Universidad de los Andes (UNIANDES): The basic programming courses (object-oriented algorithmic and programming -APO- 1 and 2) of Los Andes University teach how to use tools and techniques for solving real problems in terms of a computer program. The APO1 course guides students through the different steps in the development process of a computer program. Students learn about problem analysis, design and develop a solution by solving simple exercises. The APO2 course continues the topics covered in the first programming course. This course introduces unit, test, basic algorithms for searching and sorting, reading and writing files and new elements to model entities. Each semester the department of Systems engineering offers over 30 sections of the APO1 course and 15 sections of the APO2 course. Each section has an average of 26 students.

5) Universidad do Pampa (UNIPAMPA): The basic computer programming course at Unipampa is offered every semester for students coming from different programs, as follows: Computer Engineering, Chemistry Engineering, Production Engineering, Food Engineering, Energy Engineering and Physics. The computer programming basic course is called "Algorithms and Programming" and it is the same for all mentioned programs. Students normally take the course together with their colleagues of the same program, however, the classes can be mixed, i.e., containing students coming from these different programs. A regular course of Algorithms and Programming at Unipampa is divided into theory and practice. In the theoretical part the students are presented to computer problems and to those computer programming structures and commands required to solve these problems. In the practical part the students implement computer programs in the laboratory using the C language. The whole course takes 60 hours (around 17 weeks) . Students of Unipampa who filled the surveys came from 6 different programs: Food Engineering (6%), Computer Engineering (47%), Renewable Energy Engineering (15%), Production Engineering (15%), Chemical Engineering (11%) and Licentiate in Physics (6%).

B. Data Gathering Instruments

It was designed among the participants of the IGUAL project that information will be collected in four main instrument:

- Student Context: This instrument captures information about the students, their background, their perception of the course and their stance in front of several technologies frequently used in the classroom.
- Learning Styles: This instrument captures information about the Learning Style preference of the student based on the Learning Style Index.
- Mental Models: This instrument captures information about the capacity of the student to have consistent Mental Models when confronted with easy programming problems.
- Course Content: This instrument tests the students in different parts of the course content to measure their comprehension of the different topics.

Depending on the availability of previously researched instruments, each instrument was created or adapted. In the case of Student Context instrument a 28 question survey was created by the project partners. This survey contained questions about the previous school of the students, their access to a computer before entering the University, their programming knowledge before taking the Programming Course, their access and use of Internet and finally their experience with Learning Mateiral online.

In the case of the Learning Style instrument, the survey and methodology was provided by [5]. This 44-question survey was developed using 'The Index of Learning Styles' Model which is focused on engineering students. This model uses four scales to measure the learning style of a subject: Active-Reflective, Sensing-Intuitive and Visual-Verbal. Results can be 'strong', 'moderate' or 'balanced' between these scales.

For the Mental Models category, the test initially proposed by Dehnadi [] and subsequently modified by Bornat, Dehnadi and Simon [], consisting in questions to determine the level of comprehension of the concepts of assigning values to variables and sequencing.

For the Course Content category, a 40-question test was created in order to measure how well the students perform in the 20 main topics covered during an Introduction to programming course, divided in 7 main groups.

The instruments were converted to online surveys/tests using the LimeSurvey tool [8]. Finally the instruments were validated with a group of graduate students from all the participating universities. The collected information was exported as a CVS file and imported in the R statistical software [6] for further analysis. For the analysis, mainly descriptive statistics were used to obtain information about the context, preferences and difficulties of the students with respect to the Programming Fundamentals course.

C. Data Gathering

In order to adjust to the specific circumstances of each university, the data gathering instruments were adapted to the local version of the Programming Course. For example, the computer language use in the Course Content instrument was C in the case of ESPOL, UAA and UNIPAMPA, but was Java for UNIANDES. As another example, only UACH applied the Mental Models instrument, because it required students completely new to programming. The details of the data gathering are presented in the following subsections:

1) ESPOL: Three instruments (Student Context, Learning Styles and Course Content) were applied to ESPOL students that have already passed the course and that are currently taking the next course in the curriculum. A total of 122 students were invited to take the survey. To assure the participation of the students, the Course Content test was graded and used as part of their current course evaluation. In total, 99 complete responses were obtained in the three instruments, that means that there were 99 out of 122 students that successfully completed all the 3 instruments.

2) UAA: Three instruments (Student Context, Learning Styles and Course Content) were applied. The sample was taken from second semester students of ISC (Computer Systems Engineer) and LTI (Information Technology Graduate) undergraduate programs. At the time of data collecting, both groups were taking the Programming I course. ISC and LTI students participated by direct invitation, during a normal one hour lecture. Data was collected using a computer laboratory (under the supervision of their programming teacher) and an online survey tool. Not all the invited students completed the surveys. The Learning Style survey was completed by 50 students. The Student Context survey was completed by 53 students. The Course Contents test was taken by 42 students.

3) UACh: UACh only applied the Mental Models instrument to students that just entered the program. The Civil Engineer Computer Science program offers 64 vacancies each year. Students must take the University Selection Test (PSU), a national measure, applied as part of the requirements for admission to most universities in Chile. With the score obtained and the GPA of the last 4 years of secondary education, it generates a score that allows a student to apply for a given program. Of the 64 students admitted, 51 participated in the survey.

4) UNIANDES: Two instruments (Student Context and Course Content) were applied to UNIANDES students. This study was conducted with 153 students of the APO1 course immediately after they performed their final exams.

5) UNIPAMPA: Three instruments (Student Context, Learning Sytles and Course Content) were applied in UNIPAMPA. The surveys were put available online and students that took the course of Algorithms and Programming last year (2010) were asked to freely answer the surveys. Students could answer from home or using the university computer facilities. In total, we contacted around 200 students (through email, and personal conversations), and from these, only 22 filled the questionnaires. This lack of data led us to ask for students of the current semester (2011) to also answer the surveys. As the semester was still in the middle, we cut out from the content survey the questions that were still not covered in the current semester, precisely, those related to the topics of functions and procedures. Answering the surveys was not mandatory and students were not compensated with any grade to perform such activity. From the approximately 120 students which were taking the course during the current semester, 41 filled the surveys. In total, our sample was composed by 63 cases.

III. Results and Analysis

In the same way that the collected information was divided into four categories, the analysis is also divide in four subsections: Student Context, Learning Styles, Mental Models and Course Content. Each one of this subsections will help us to gain a deeper insight on the needs and preferences of the students in Latin America in order to tailor the IGUAL learning solutions to those needs.

A. Student Context

1) Age Distribution: The first analysis performed was to obtain the age distribution of students that that are taking or just passed the Introduction to Programming courses. This distribution for each participating University could be seen in Figure 1 on the following page.

These distributions show that there are mainly two groups of students that will be affected by the project. One group is made out by students 18-21 years old. The second group have 22-30 years old students. The presence of these group is different in the participating Universities. UAA and UNIANDES have mainly students from the first group. In the case of ESPOL and UNIPAMPA, the main population consist of students from the first group, but have a significant contribution from students from the second group.

In the study student population, there is a small fraction of older students (30 - 40 years old). This population, while a minority, could be an interesting testing group for the learning solutions to be developed in the project.

2) Years in University Distribution: To complement the previous age analysis, the number of years that the student has been in the University was also obtained. The resulting distributions for each participating University can be seen in Figure 2 on page 5.

As expected, most of the students take the course during their first or second year at the University. The presence of older students in ESPOL and UNIPAMPA could be better explained with this analysis. A significant proportion of ESPOL students take the course by the end of their program (after 4 or more years at the University). On the other hand, UNIPAMPA older students have at maximum 3 years studying at the University. In the case of ESPOL, Telecommunications and Telematic majors seem to leave this subject to the end of their curriculum because Computers Science courses are not fundamental for the main study topics. In the case of UNIPAMPA, it is important to highlight that students in their second and third years (around 30%) is an indication that they are somehow late in their studies since programming topics are normally located at the beginning of their programs.

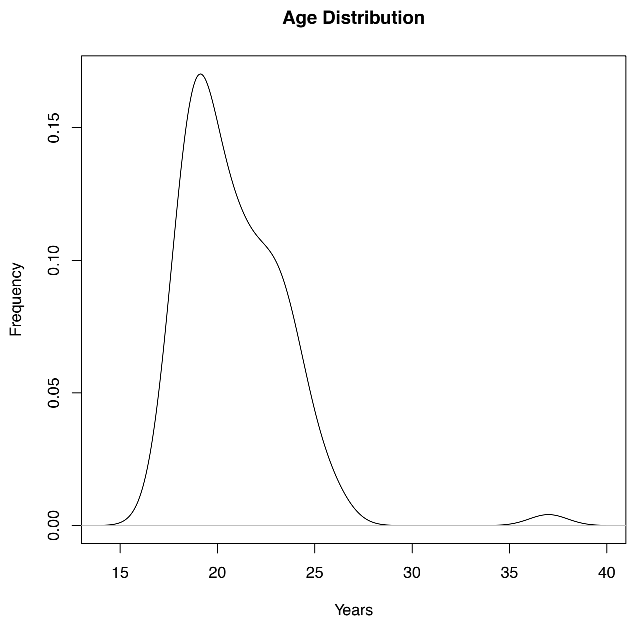
3) Schooling: The students were surveyed about the type of High School (secondary education) that they attended. Their answer are summarized in Figure 3 on page 6. In the case of ESPOL and UNIANDES, the majority of the students came from private schools, while in UAA and UNIPAMPA, the majority come from public schools. It is interesting to note that ESPOL is a public University, while UNIANDES is a private university.

This setting is ideal for the testing of the learning solutions, because these solutions will be piloted both in universities that have a majority of public schooled students and also in universities that have a majority of privately schooled students.

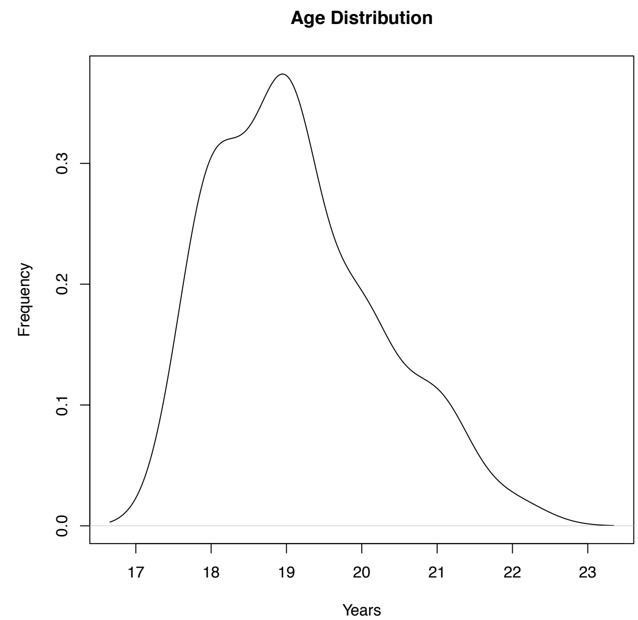
Most students attended schools located in a city as can be seen in Figure 4 on page 7. UAA has the largest rural population and will be interesting to test the learning solutions in this population, given that in Latin America there exist a technological gap between Urban and Rural areas.

4) Gender: The Student Context survey contained a question about the gender of the students. The count of the results can be seen in Figure 5 on page 8.

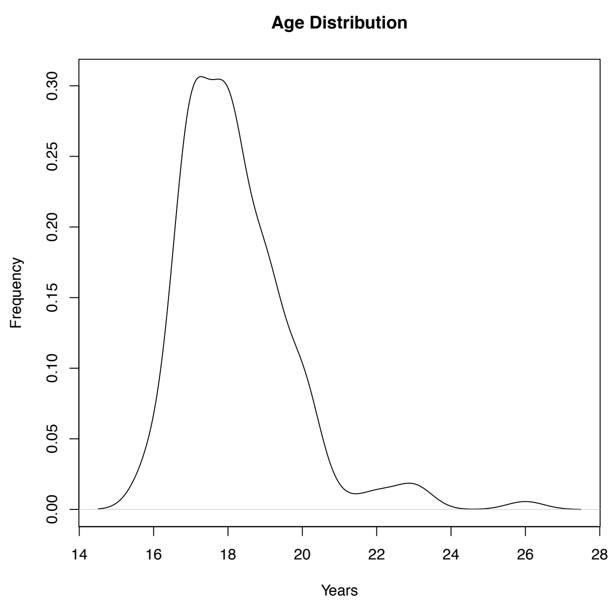
While most students are Male, there is a strong Female presence. ESPOL is the University with the least percentage of Female students (24%), while UAA is the one with the largest proportion (38%). The solutions to be developed should take into account the gender of the student.



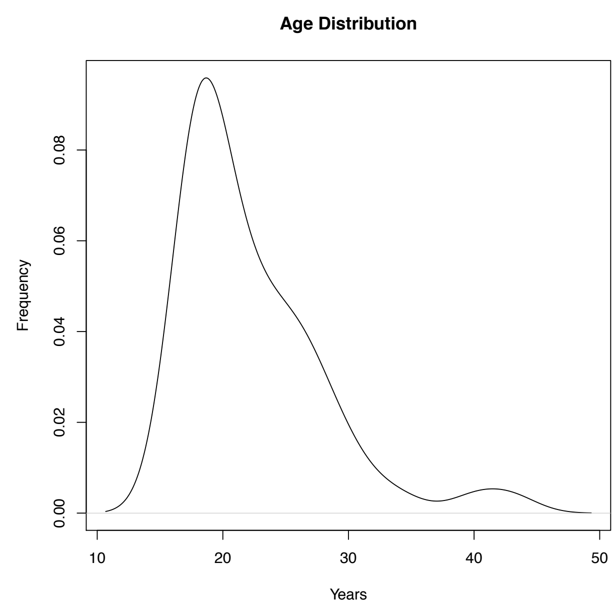
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

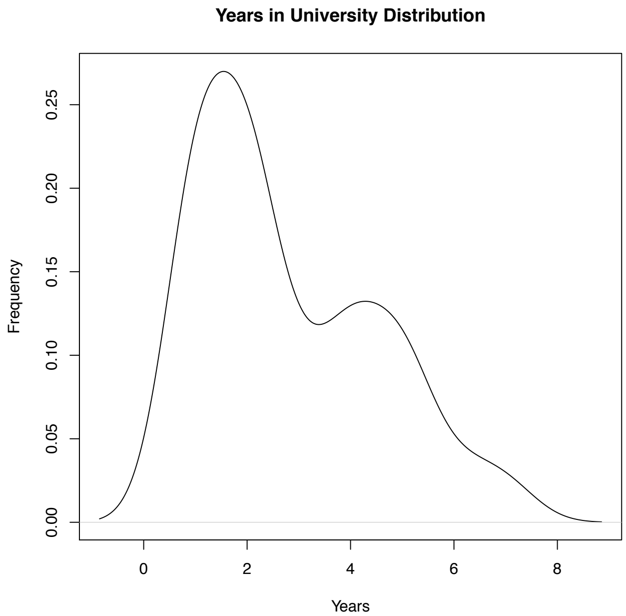
Figure 1: Age Distribution of Students

5) Access to Computers before University: When surveyed about if they had access to a computer before entering the University, the students answered as shown in Figure 6 on page 9. Most of them had regular access to a computer before entering the University. ESPOL and UAA have the largest proportion of students without access (16%), while UNIANDES, being a private and costly University, has only a 1% of students that have never used a computer before entering the University.

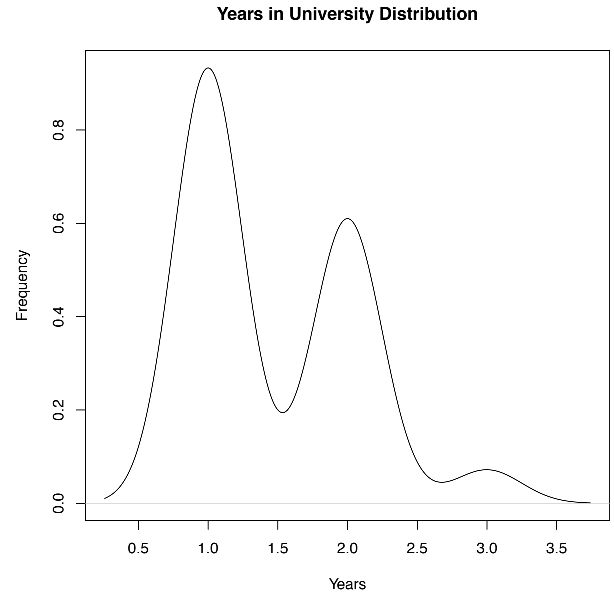
Then, the students that answer yes were inquired about where they had access to that computer. Figure 7 shows that most of them had that computer at home (88% - 99%). A smaller proportion had access to the computer at a Cybercafe, specially in the case of ESPOL and UAA.

Finally, the students answered how they learned to use that computer. Figure 8 on page 11 shows that most of them learned by themselves or at school. Here it can be seen the importance of having computers at school. In the case of UNIPAMPA, there is a significant percentage of students that learned in an academy or another type of learning institution. The rest learnt from they parents or friends.

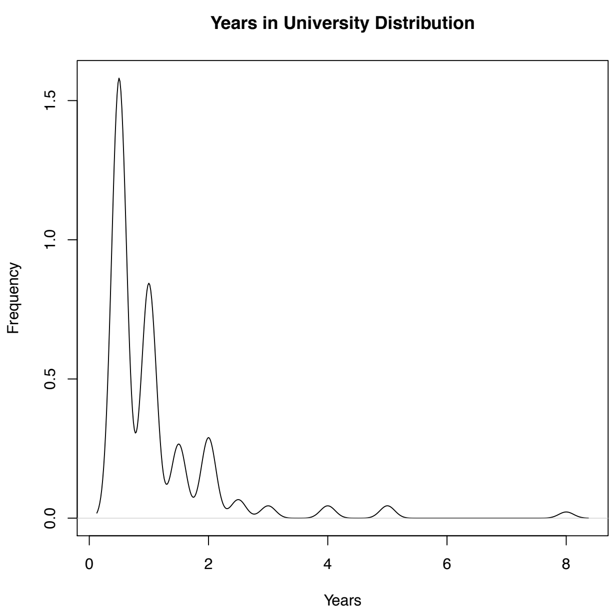
These findings prove that the access to technology problem



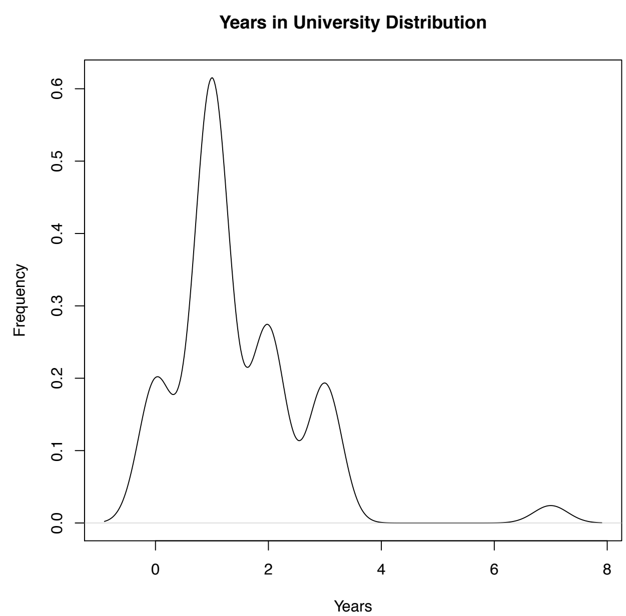
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

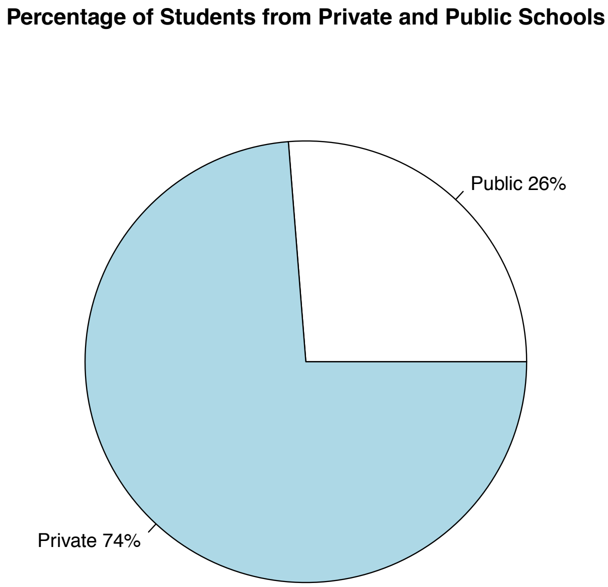
Figure 2: Years in University Distribution

is gradually disappearing in Latin America, but it is still present in a segment of the student population. This difference, specially in a career that requires computers is bound to create a gap between different type of students. From the ones that has access, again, the large majority had the computer at home, however, a small percentage had to access through a Cybercafe, where resources has to be spent, making the availability limited.

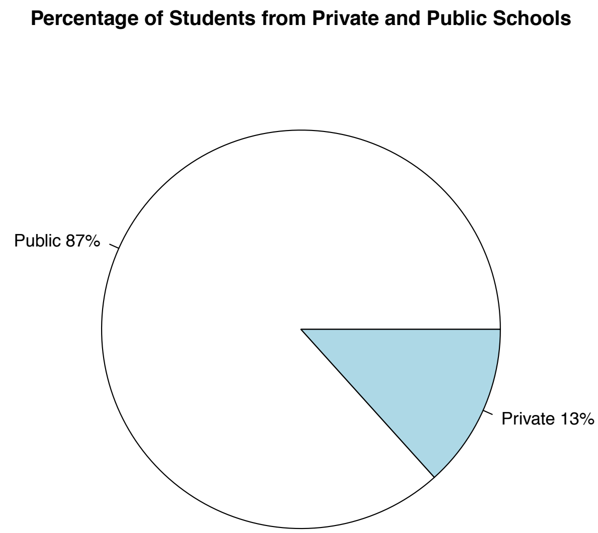
Schools seems to be a large contributor on how a student learn to use the computer. If schools do not provide access to them, it is expected that their students are at disadvantages with their peers at University.

6) Use of Computer before University: The students were survey about their usage of computers before entering University. Six options were presented: to Play, to Browse the Web, Word Processing, Spreadsheets, Educational and Programming. The students grade their use of the computer for each one of this activities in a 5-level scale from "Very Low" to "Very High". The results are summarized in Figure 9 for each of the participating institutions.

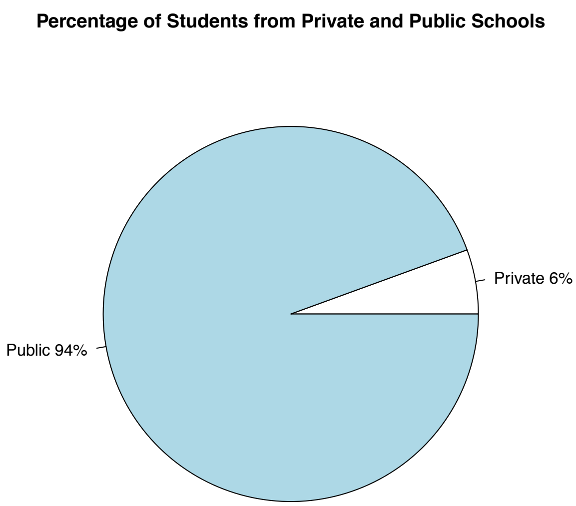
It is clear from the results that programming is not a popular activity before University. The students mainly used the computer to Browse the Web and do Word Processing. There



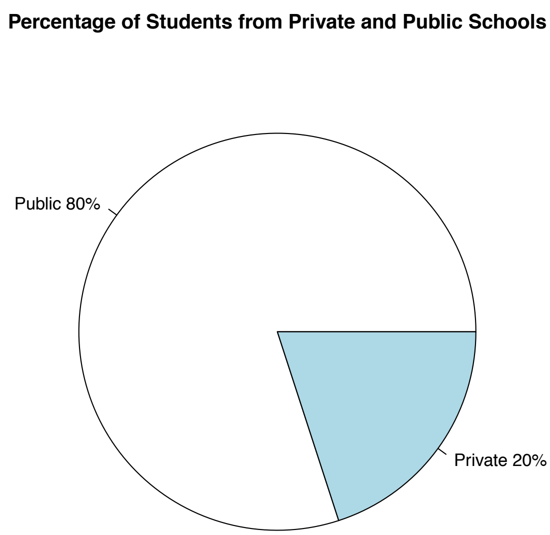
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 3: Public vs. Private Schooling

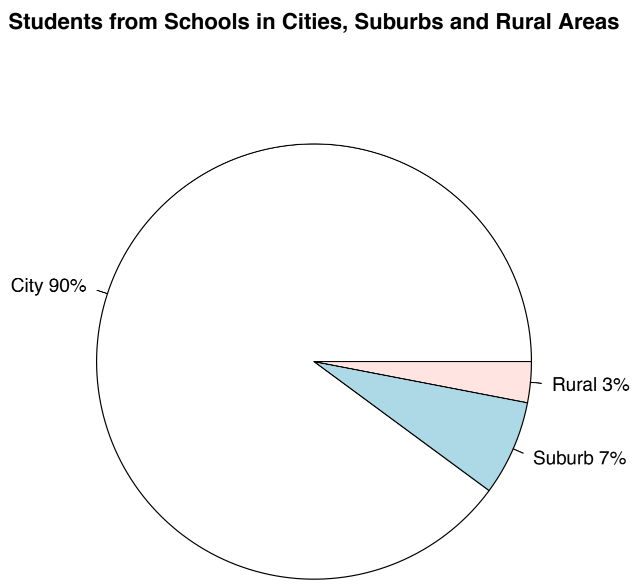
is an interesting percentage of students that use the Computer for Educational purposes before entering the University.

These findings means that most students do not had previous experience programming that could contribute to the understanding of the material presented in the course.

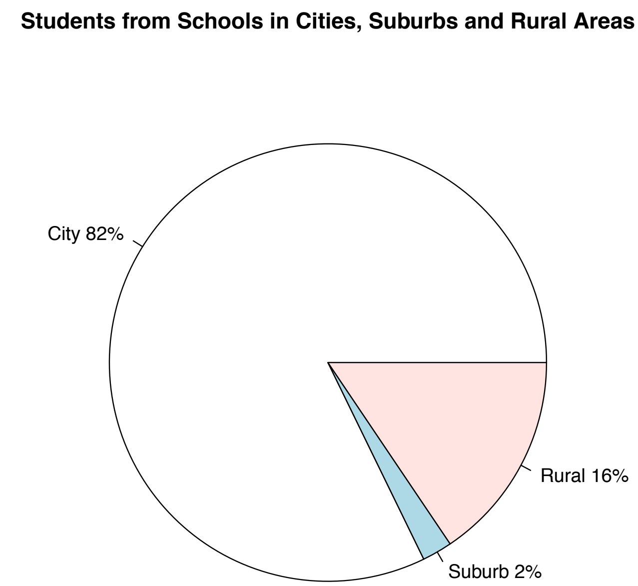
7) Programming before University: The students were more precisely asked about their previous programming knowledge before entering University. The first question was if they followed a programming course before entering the University. The result to this question can be seen in Figure 10. Very few students follow a programming course before entering the University, although that percentage is higher in UAA and UNIANDES. This students are bound to be somewhat bored during the Introductory Programming course and should be considered into the learning solutions to be developed.

The second question was about their previous knowledge of programming. Results can be seen in Figure 11. Most of the students had Low or Very Low knowledge of Programming. Due to previous programming courses, in UAA there is a lower percentage of "Very Low" knowledge. The rest seems to be somewhat comfortable with programming in at least one programming language. This difference is bound to cause problem in any computer course where both groups are mixed.

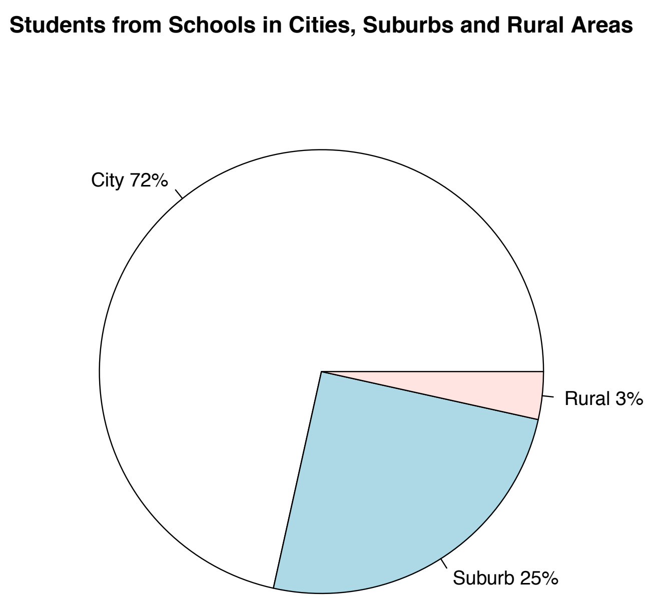
Finally, they were inquire about their level of knowledge in different computer languages. The results are presented in Figure 12 for each participating institution. The most used computer language was Visual Basic. Again here is very



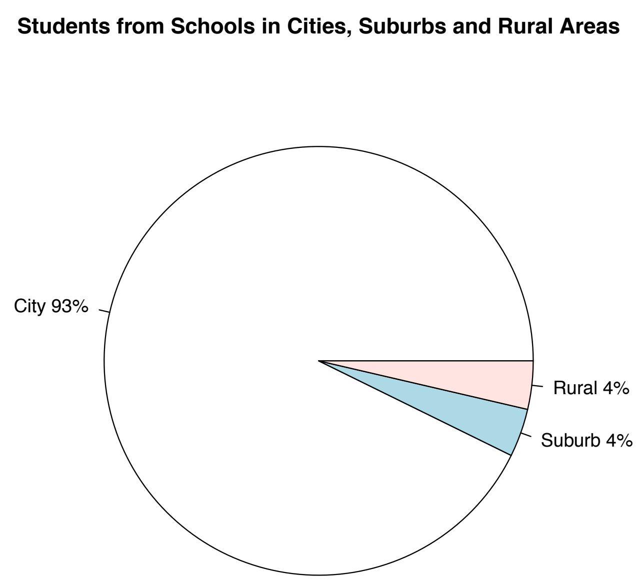
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 4: Location of High School

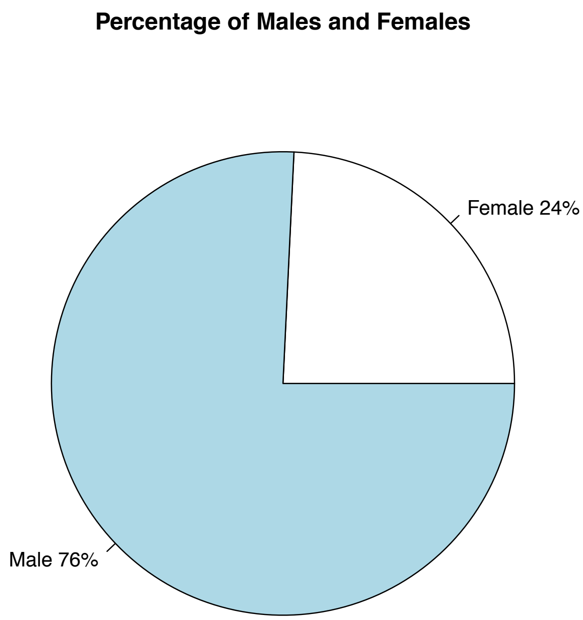
obvious that most students enter the university without any knowledge of programming.

These findings make it clear that access to computers is not necessarily translated into knowing how to program. While there is a small group of students that had programmed before, most do not posses programming skills. This imply that the results of the project could be used not only by students from public schools, but also from students from private schools that lack programming skills.

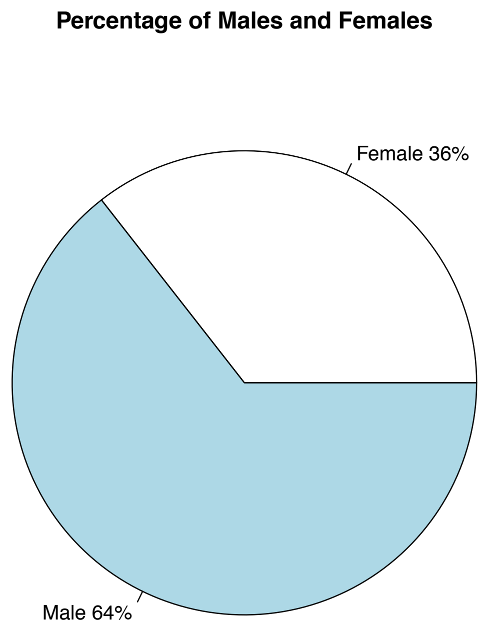
8) Effect of the Programming Fundamental Course: In order to establish if the Programming Fundamentals course had a perceived impact in the level of knowledge that the student think that they posses, they were asked about that level before and after they have taken the course (in the case that they have already taken it). The results are presented in Figure 13.

The results suggest that most students think that the course has a positive impact in their knowledge level, but that impact is not as good as expected. Most of the students answer that they had a Basic level of programming after taking the course. This finding implies that there is a lot of room for improvement in the way in which the course is taught.

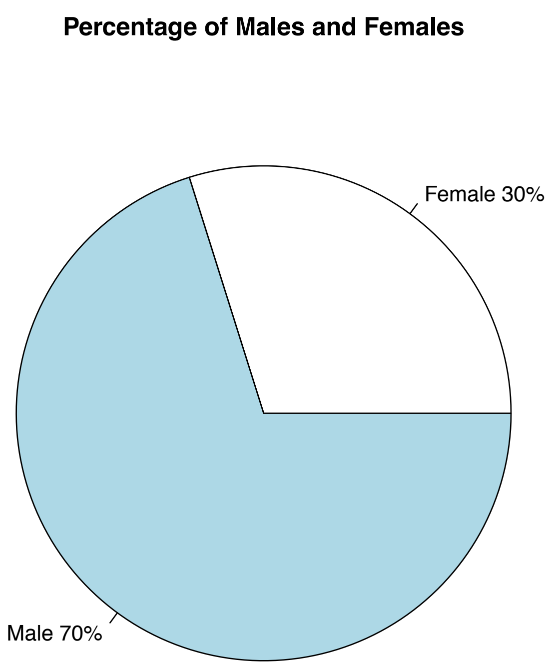
9) Weekly Study Hours: The students answered a question about the amount of time that they study outside the class and mandatory laboratories. The distribution of the self-reported time is presented in Figure 14. It can be concluded that most



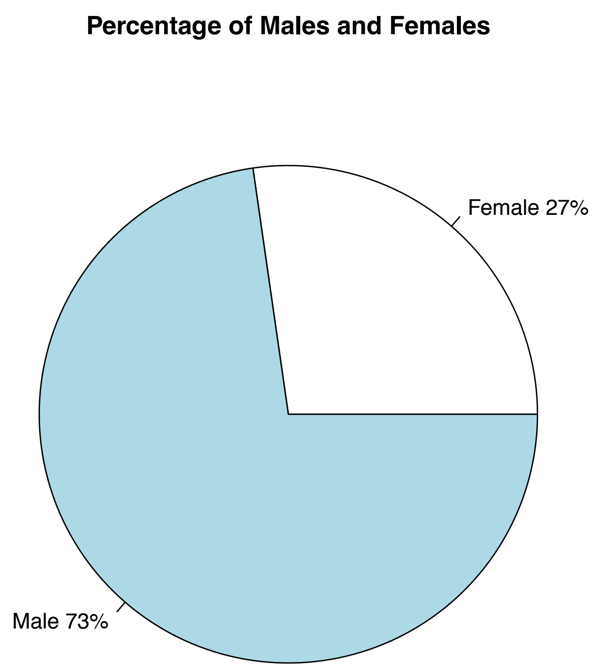
(a) ESPOL



(b) UAA

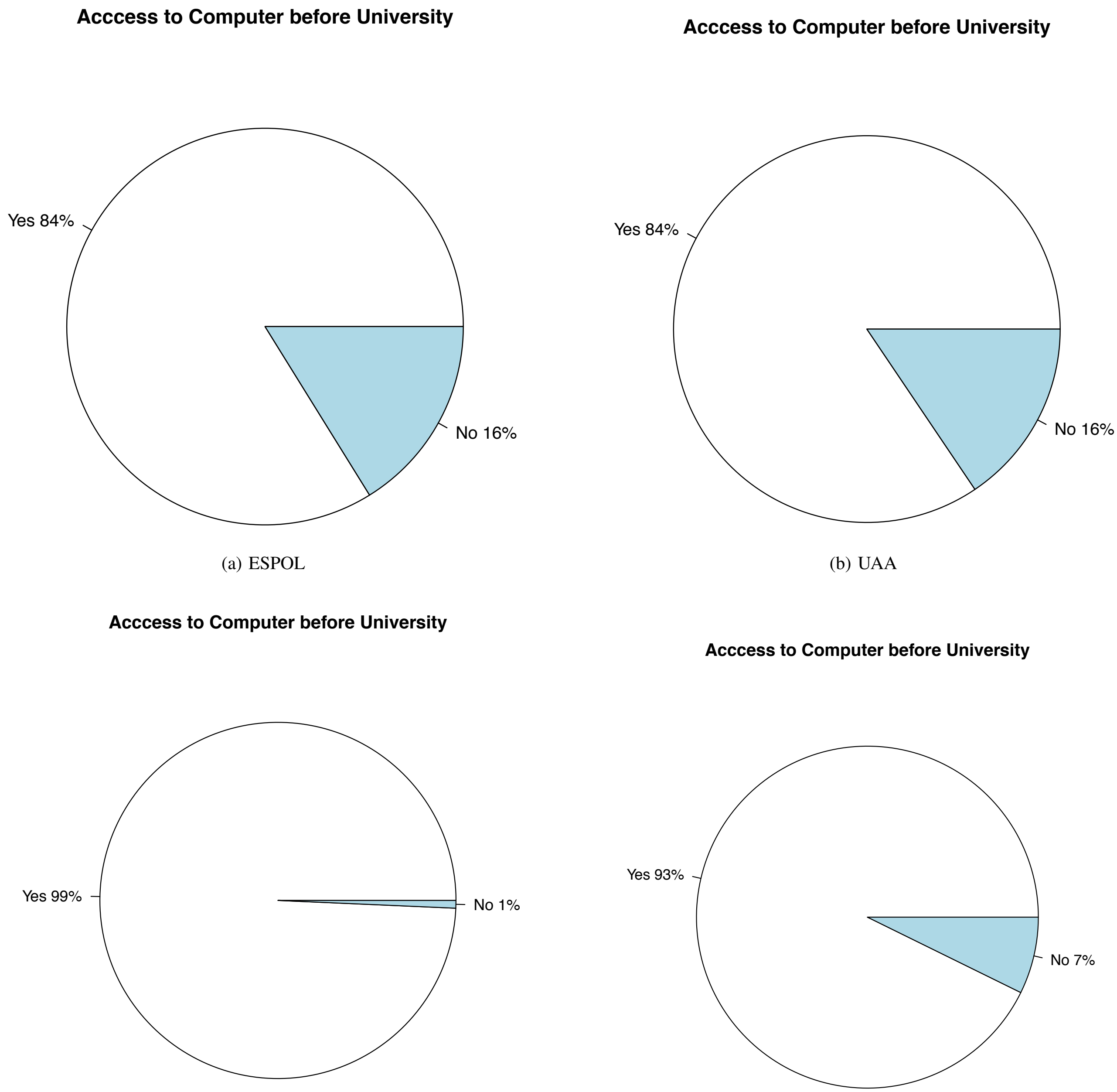


(c) UNIANDES



(d) UNIPAMPA

Figure 5: Percentage of Males and Females



(a) ESPOL (b) UAA (c) UNIANDES (d) UNIPAMPA

Figure 6: Access to Computer

students dedicate between 4 to 6 hours per week to review the material. The project should seek to improve the efficiency and effectiveness of those hours.

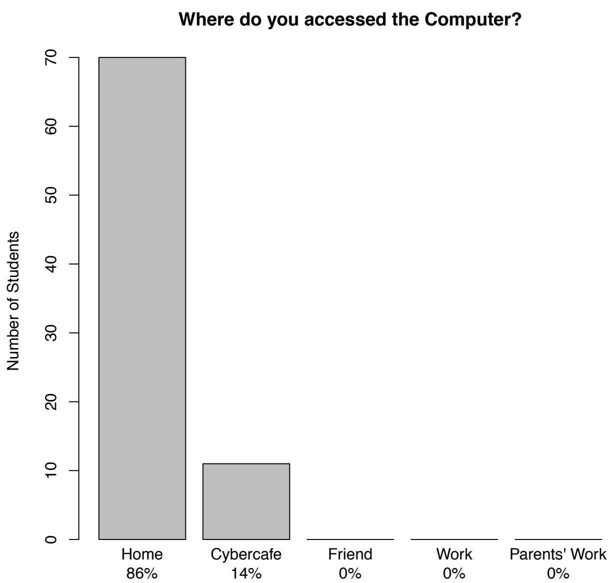
10) Perception of Communication Technologies: The students were surveyed about their knowledge and perception of different Communication Technologies. Four specific questions were made about all those technologies:

- Do you know how to use it?
- Do you like to use it?
- Do your professors use it for your courses?
- Do you like your professors using this technology in your courses?

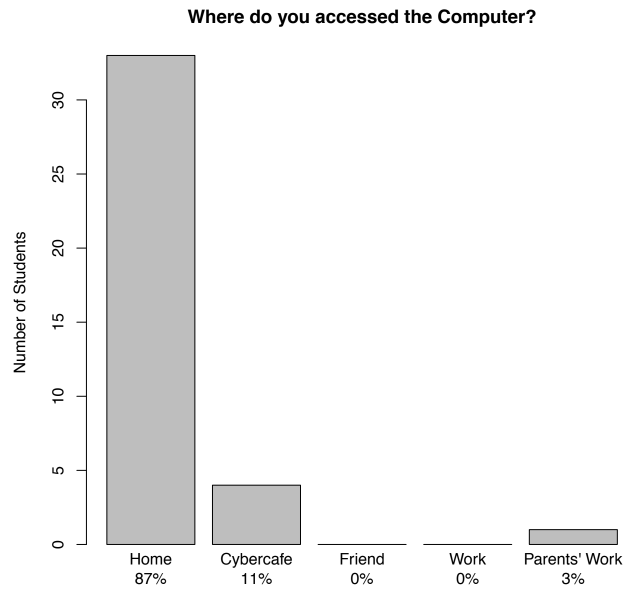
To these questions, the student answered in a 5-level scale from "Very Low" to "Very High". The answers to these questions can be seen in Figures 15 on page 18, 16 on page 19, 17 on page 20 and 18 on page 21.

The first technology presented was the Telephone or regular phone calls. As expected, most students dominate this technology and is well liked. However, there is a strong resistant from the students to use this technology in the classroom. It seems that phone calls are not perceived as a valuable or desirable medium for education.

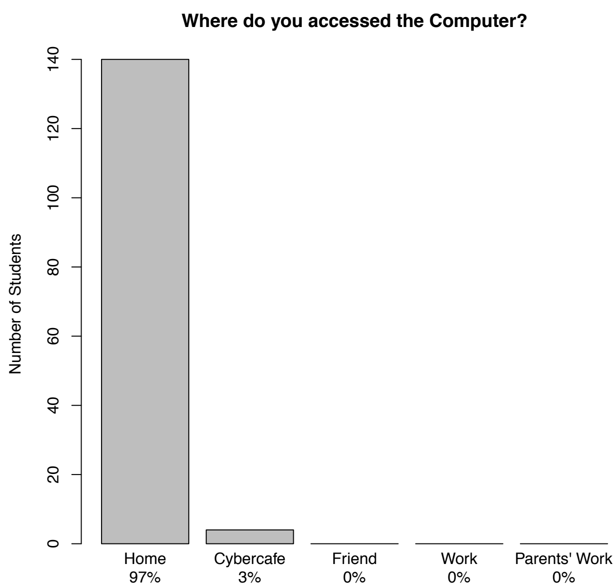
Then the students were inquired about Email. This is by far the most accepted technology. Almost all the students



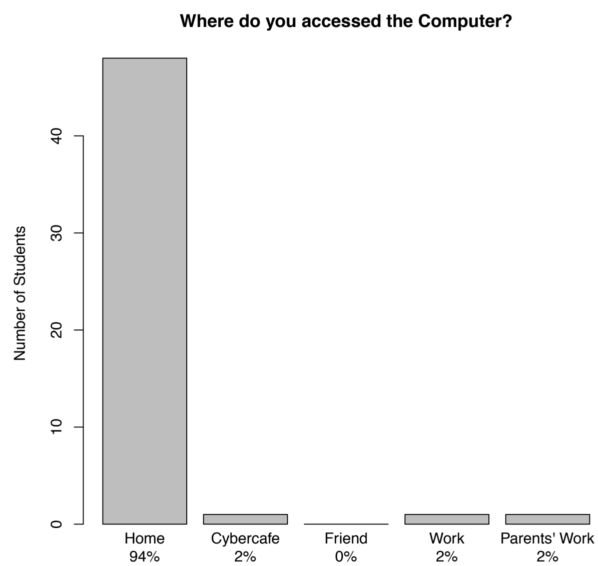
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 7: Access Place

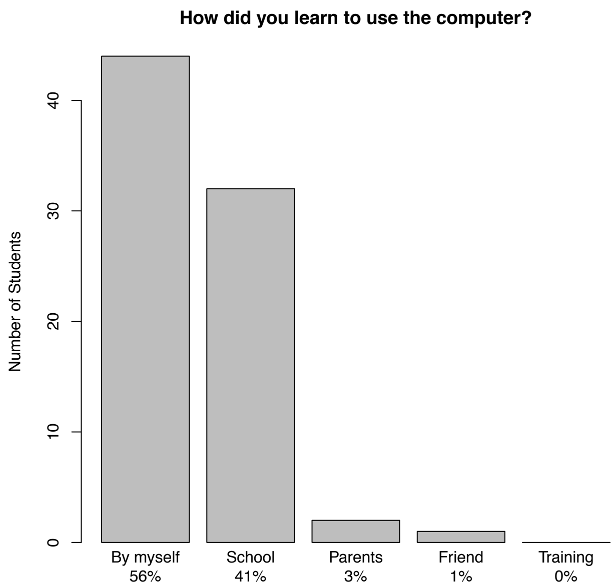
and professors use it and the students like to use it for their education. The project should make use of this medium for the learning solution.

Discussion Groups were rated by the students. Here there is not a clear consensus on the acceptance of this technology. While a significant group would like that their professors use it in the course, there is also a considerable group that does not like it. This finding implies that Discussion Groups should be used only if it has been determined that the student like that type of activity.

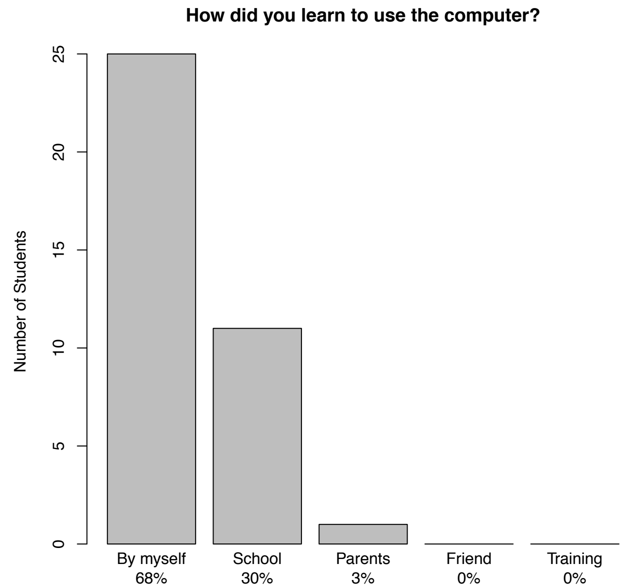
Students answered about their use and preferences about Wikis. It seems that a large group of students do not know how to use wikis (maybe they have never used them). This lack of experience leads to a mostly apathetic perception of this technology.

Blogs were also examined. The pattern is similar to Wikis, although the most salient aspect is that the students perceive that they professors do not use them for educational purposes. This is not a favored technology.

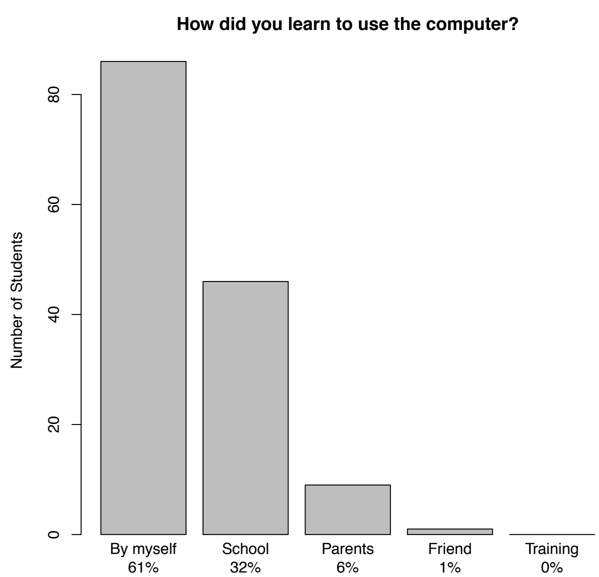
The use and preference of SMS was also examined. As expected, most students use and like SMS, however, they do not think that the professor should use it in classroom. This is again a case where the students want to separate the use of their personal communication tools from academic communication tools.



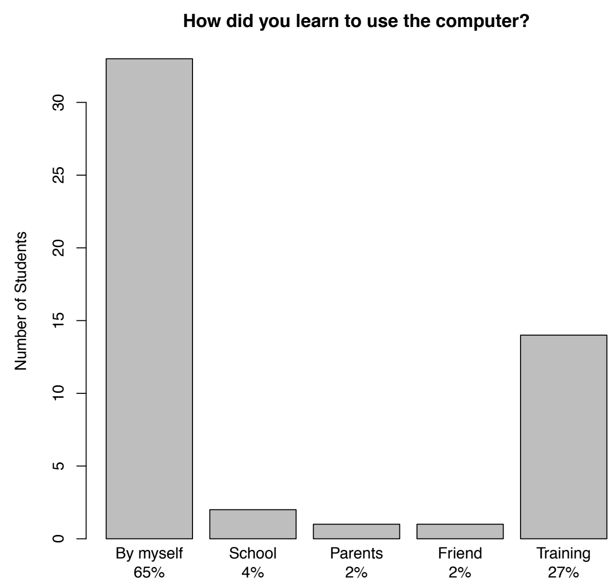
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 8: How students learned to use the computer

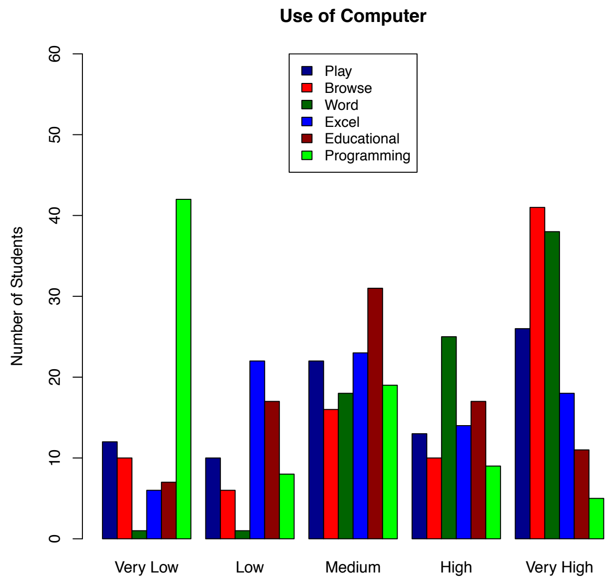
The students were also surveyed about the use of Chat. Similarly to SMS, this is a technology well know and used by students, but with not a clear preference to be used in an educational context.

The use of Instant Messaging was also analyzed. The same pattern of SMS and Chat technologies can be appreciated. This is also not a technology to be used in the educational setting.

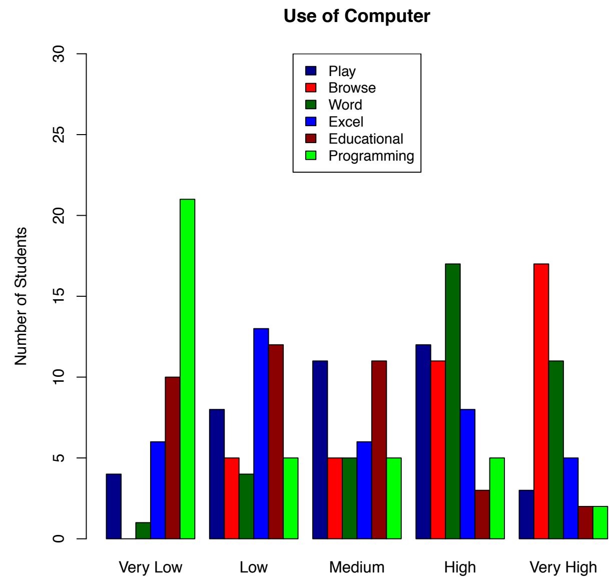
The use of social networks was explored through the use of Facebook and Twitter systems. The widespread use of social networks could be seen the large proportion of "Very High" in the Knowledge and Preference to use. However, there is a strong opposition to use those tools in the course, specially in the case of Twitter. Again, this can be seen as the student desire to separate their personal and academic lives.

The use of videoconference was studied with questions about the most common videoconference tool, Skype, and a generic question about Videoconference in general. The results are almost identical. There is not a clear preference to use them or not use them. In the case of Skype there is a small majority that reject its use in their courses. These advanced communication tools are not clearly welcomed in the face-to-face courses.

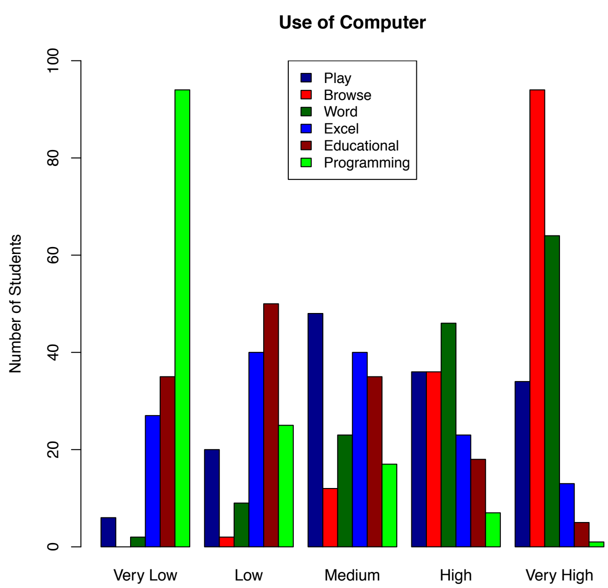
Finally, the students were asked about their use and preferences about the Learning Management Systems (LMS). Here the main conclusion is that they think that their professors do not use the LMS enough and there is a small tendency that



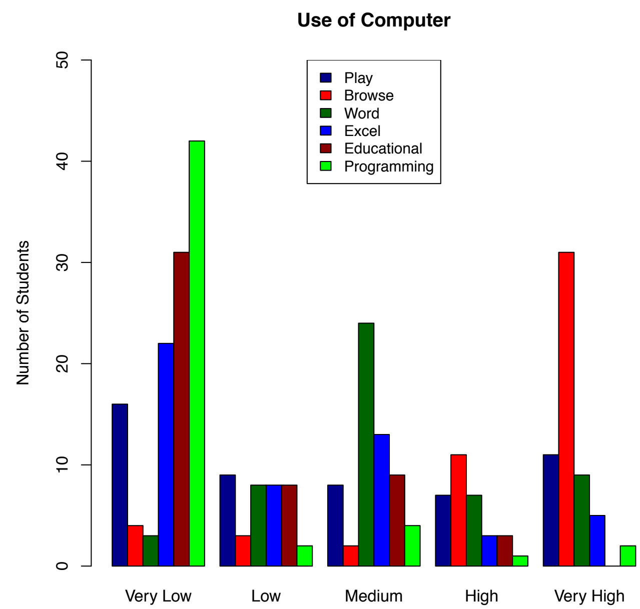
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

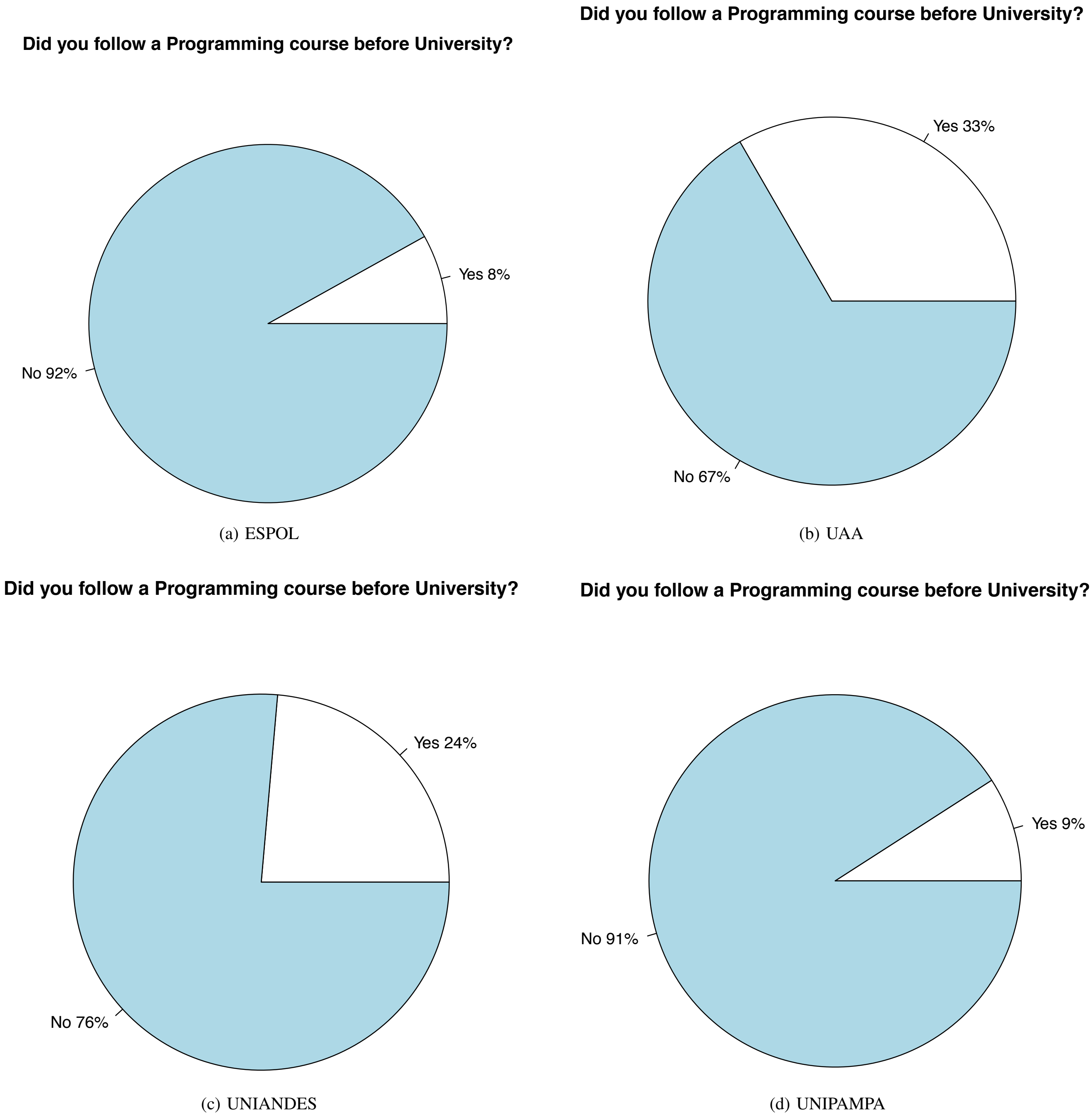
Figure 9: Uses of Computer by Students

would like to seeing used more in their courses. This finding suggest that the LMS could be a good delivery channel for the learning solutions implemented in the project.

As conclusion for this part of the survey is that traditional technologies such as email and LMS are the most appreciated by students. The use of new, more complex technologies should be done in a way that do not overstep over the personal life of the students.

11) Access to Internet: The students were surveyed about their frequency of access to Internet. The answers could be seen in Figure 19. It is clear that all the students have frequent access to Internet while they are at the University, with the majority connecting at least daily. This finding provide support to the creation of online tools to help disadvantaged students.

12) Use of Internet: The survey asked the students about what activities they perform online. The activities in the list were: Entretainment, Communication, Study, Independent Learning, Communication with Class and Work. They rate each one of these activities with a 5-level scale from "Very Low" to "Very High". Their answer can be seen in Figure 20. Especially interesting in these results is the fact that the students frequently use the Internet to Study and Independent



(a) ESPOL (b) UAA (c) UNIANDES (d) UNIPAMPA

Figure 10: Previous Programming Course

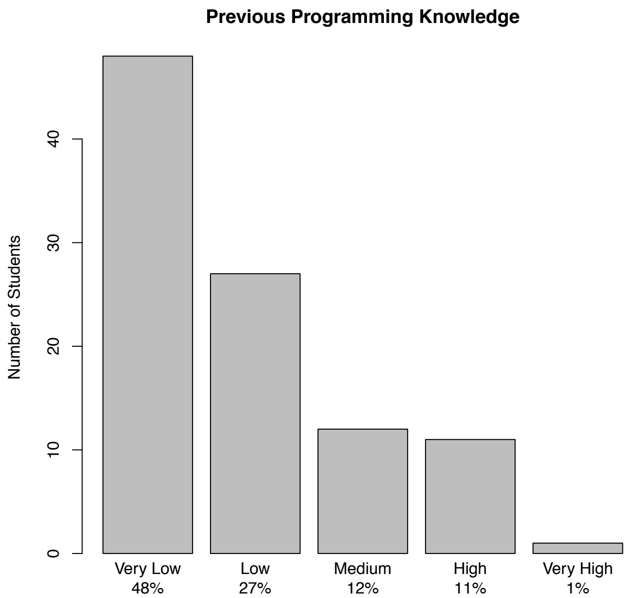
Learning. This again support the premises of the project, namely that the students could use Internet to access learning solutions that could help them in their learning process.

13) Educational Content on the Web: The last part of the Student Context survey asked about Educational Material or Content online. The first question tried to measure if they students have searched on their own for Educational Materials online. The result can be seen in Figure 21. Most of the students have frequently searched for educational content online.

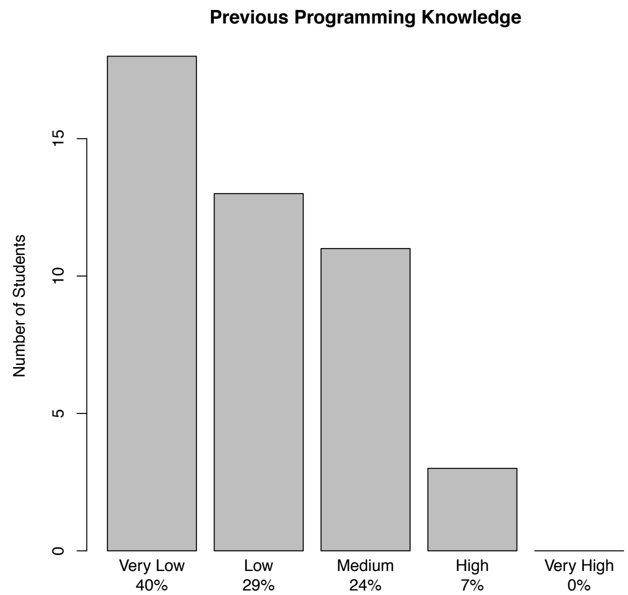
When asked about if they have indeed found educational materials online, they again responded mostly affirmative (Figure 21).

Professors are another source of educational material online. To test this, the students were asked if their professors have provide them with links online resources. They answer, that can be seen in Figure 21, show a mainly positive answer, but less clear than in the previous questions, implying that the professor not always provide those links.

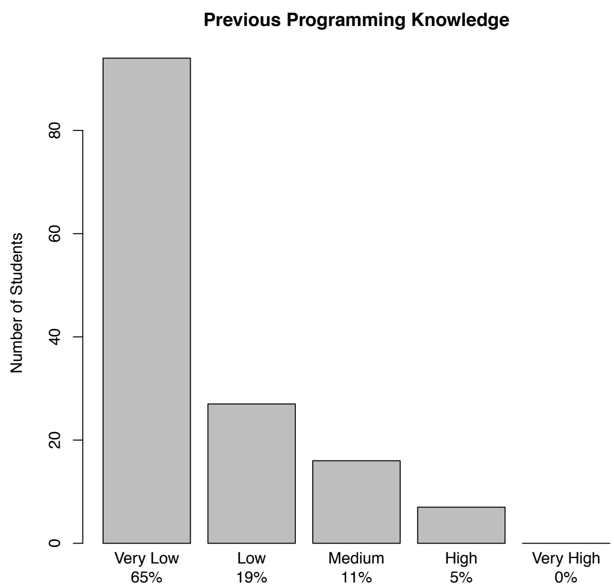
Finally, the students were asked about the quality of the materials that they access online. Figure 21 shows a summary of their responses. The students found that the materials that



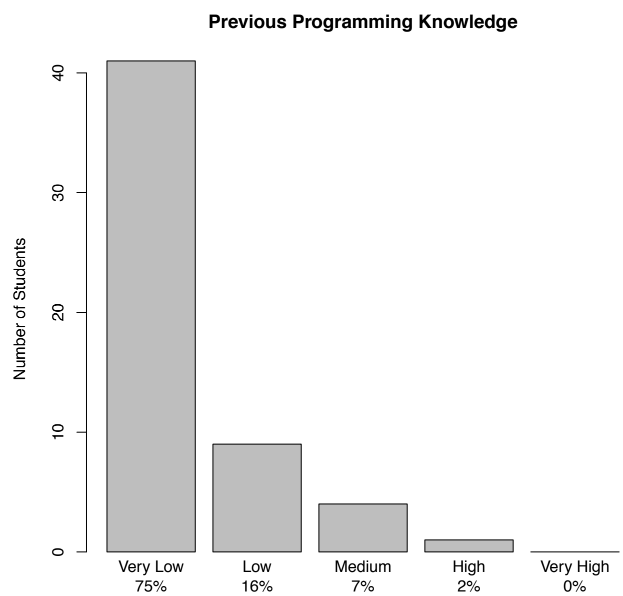
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 11: Previous Programming Knowledge

they access online have a similar quality to the traditional ones.

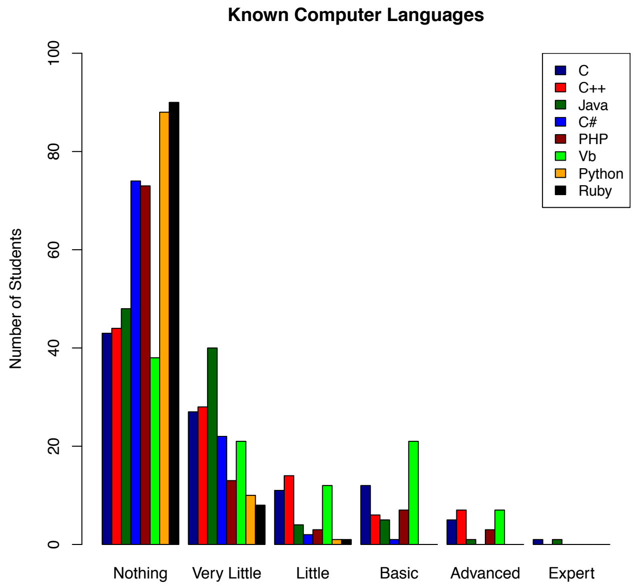
These findings provide strong evidence that the students use and like online materials. This validates again the objectives of the project.

B. Learning Styles

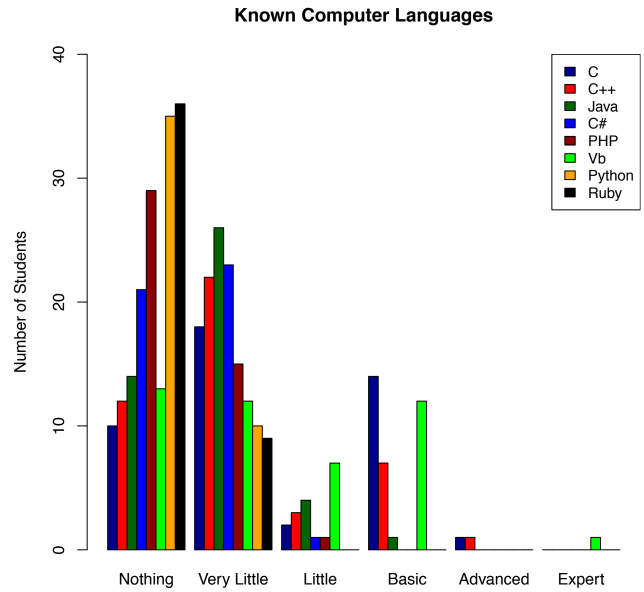
To determine the learning preferences of the students, the Learning Style Index survey was applied to students from ESPOL, UAA and UNIPAMPA. This survey determine they preferences in four axis:

1) Active-Reflective: The main characteristics of Active and Reflective learners are [5]:

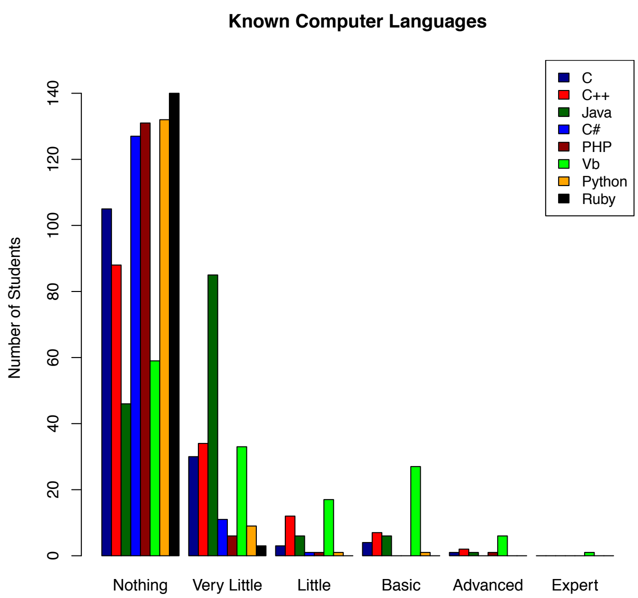
- Active learners tend to retain and understand information best by doing something active with it–discussing or applying it or explaining it to others. Reflective learners prefer to think about it quietly first.
- "Let’s try it out and see how it works" is an active learner’s phrase; "Let’s think it through first" is the reflective learner’s response.
- Active learners tend to like group work more than reflective learners, who prefer working alone.
- Sitting through lectures without getting to do anything physical but take notes is hard for both learning types, but particularly hard for active learners.



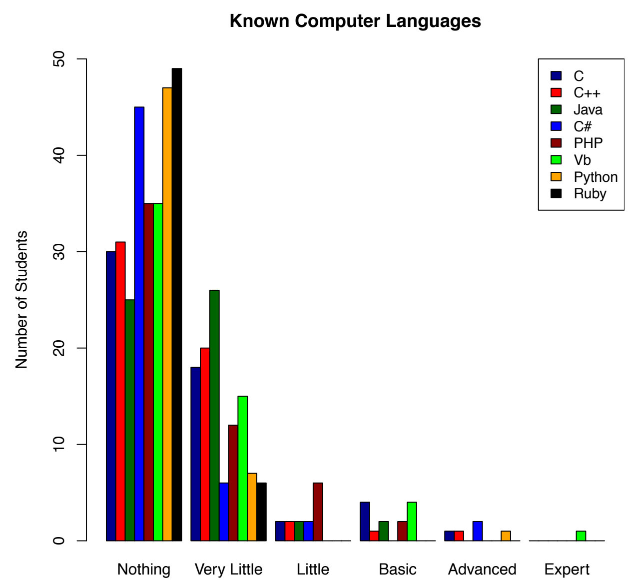
(a) ESPOL



(b) UAA



(c) UNIANDES



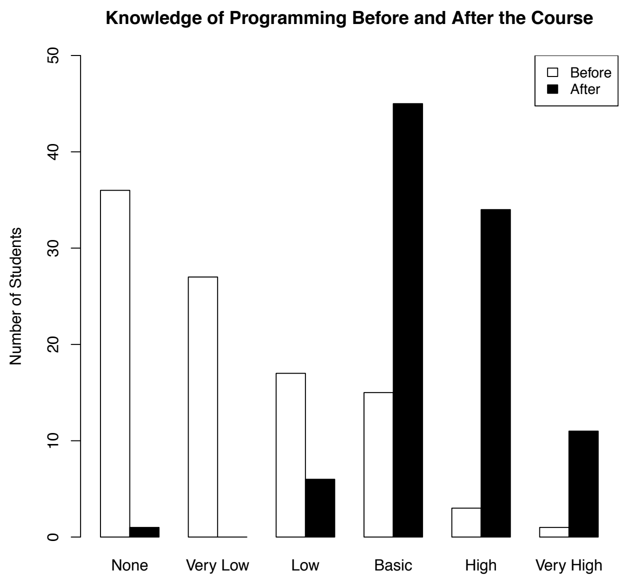
(d) UNIPAMPA

Figure 12: Programming Language Knowledge

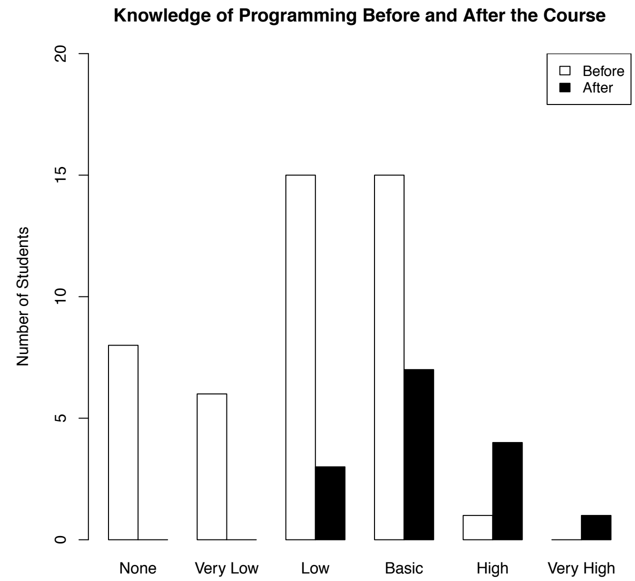
The students surveyed are distributed in the Active-Reflective axis as presented in Figure 22 on page 26. While in there are almost the same amount of Active and Reflective students, there is a small tendency to a mild active stance.

2) Sensitive-Intuitive: The main characteristics of the Sensitive and Intuitive students are [5]:

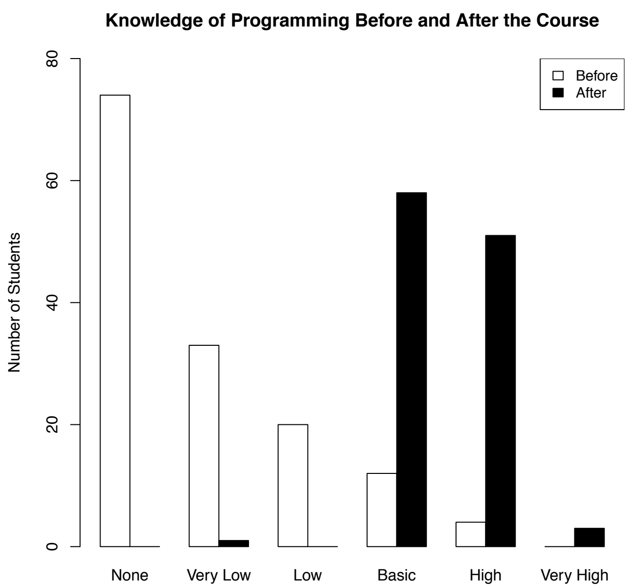
- Sensing learners tend to like learning facts, intuitive learners often prefer discovering possibilities and relationships.
- Sensors often like solving problems by well-established methods and dislike complications and surprises; intuitors like innovation and dislike repetition.
- Sensors are more likely than intuitors to resent being tested on material that has not been explicitly covered in class.
- Sensors tend to be patient with details and good at memorizing facts and doing hands-on (laboratory) work; intuitors may be better at grasping new concepts and are often more comfortable than sensors with abstractions and mathematical formulations.
- Sensors tend to be more practical and careful than intuitors; intuitors tend to work faster and to be more innovative than sensors.
- Sensors don't like courses that have no apparent connec-



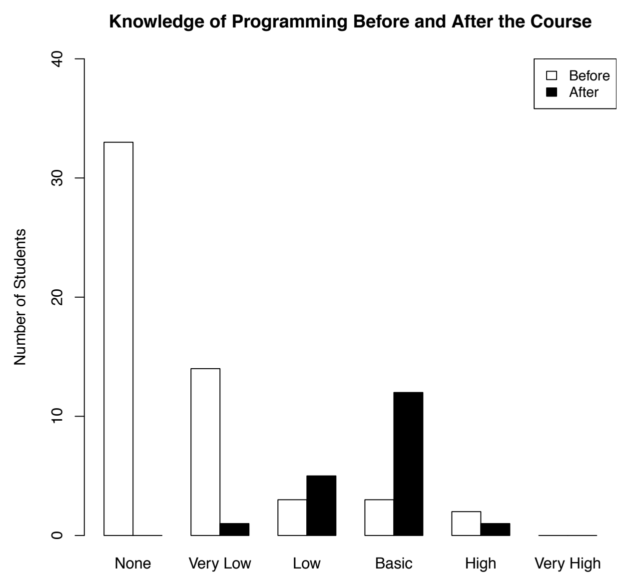
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 13: Programming Knowledge Before vs. After

tion to the real world; intuitors don't like "plug-and-chug" courses that involve a lot of memorization and routine calculations.

The students surveyed are distributed in the Sensitive-Intuitive axis as presented in Figure 22 on page 26. There is a small predominancy of mildly Sensing students with a small percentage that are strongly Sensing.

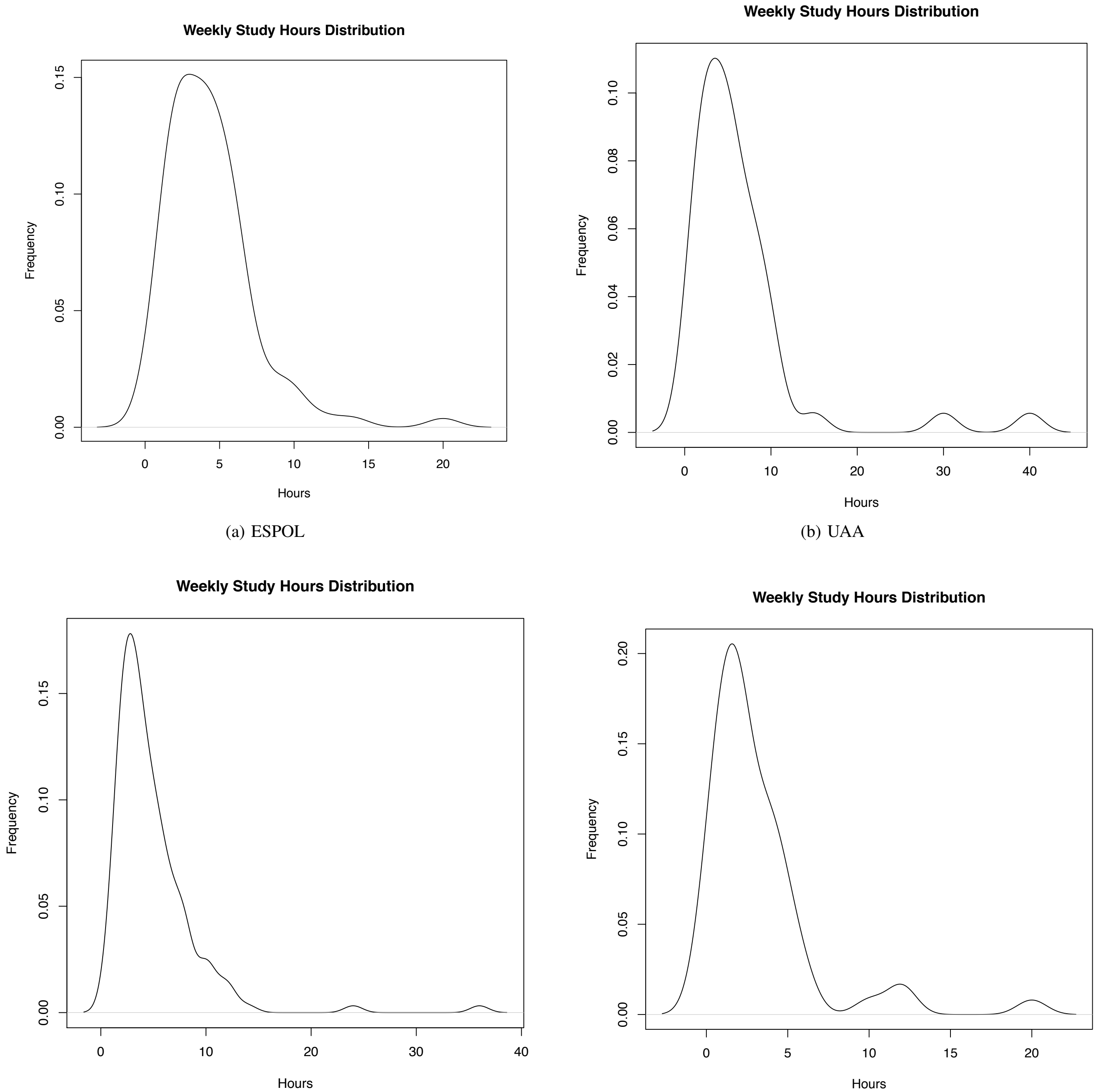
3) Visual-Verbal: The main characteristics of the Visual and Verbal students are [5]:

- Visual learners remember best what they see–pictures, diagrams, flow charts, time lines, films, and demonstrations.
- Verbal learners get more out of words–written and spoken explanations.
- Everyone learns more when information is presented both visually and verbally.

The students surveyed are distributed in the Visual-Verbal axis as presented in Figure 22 on page 26. Here, there is a strong bias towards strongly Visual, with a small fraction of mildly Verbal individuals.

4) Sequential-Global: The main characteristics of the Sequential and Global students are [5]:

- Sequential learners tend to gain understanding in linear steps, with each step following logically from the previous one. Global learners tend to learn in large jumps,



(a) ESPOL (b) UAA (c) UNIANDES (d) UNIPAMPA

Figure 14: Weekly Study Time Distribution

absorbing material almost randomly without seeing connections, and then suddenly "getting it."

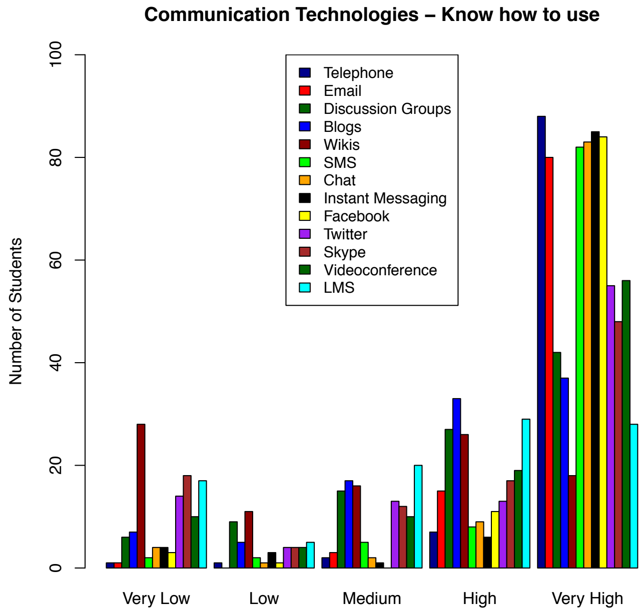
- Sequential learners tend to follow logical stepwise paths in finding solutions; global learners may be able to solve complex problems quickly or put things together in novel ways once they have grasped the big picture, but they may have difficulty explaining how they did it.

The students surveyed are distributed in the Sequential-Global axis as presented in Figure 22 on page 26. There is a small bias towards very mild Sequential students, but in general it is balanced.

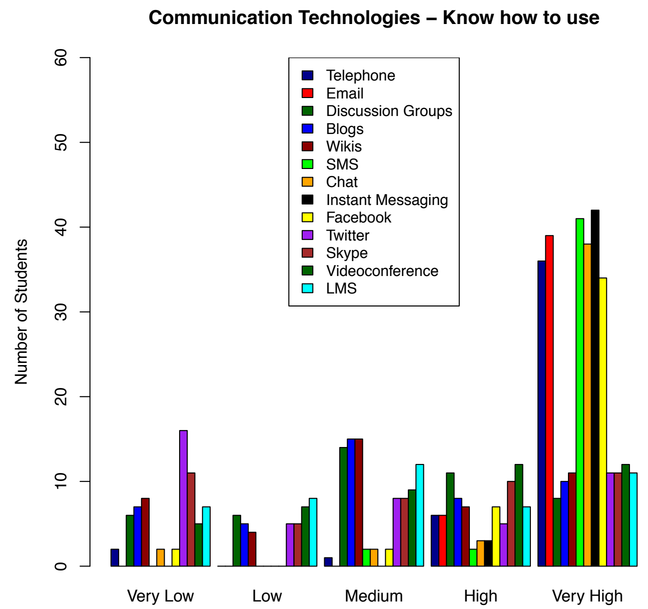
These findings, mainly that students are distributed almost equally between the different access make it more important that the adaptation and personalization of materials according to each student preference.

C. Mental Models

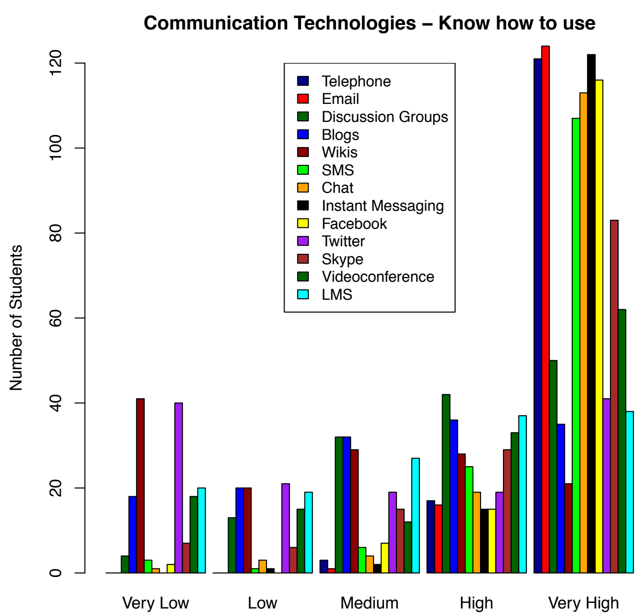
1) Test: The main understanding problems detected in a course of imperative programming, according to a study by Johnson-Laird & Steedman [7], are the mapping and sequencing, as well as recursion. Obviously, the recursion is inherently



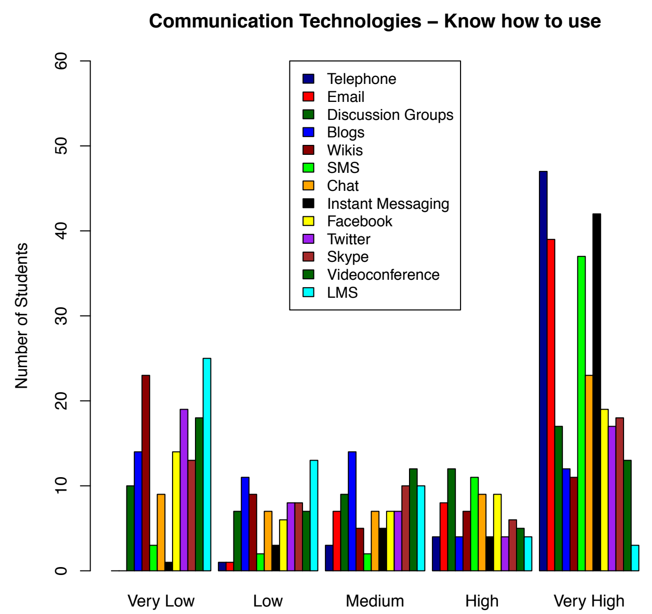
(a) ESPOL



(b) UAA



(c) UNIANDES

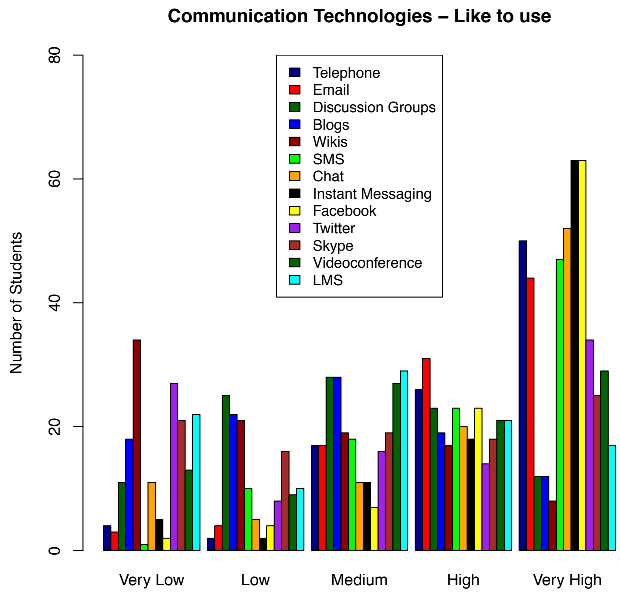


(d) UNIPAMPA

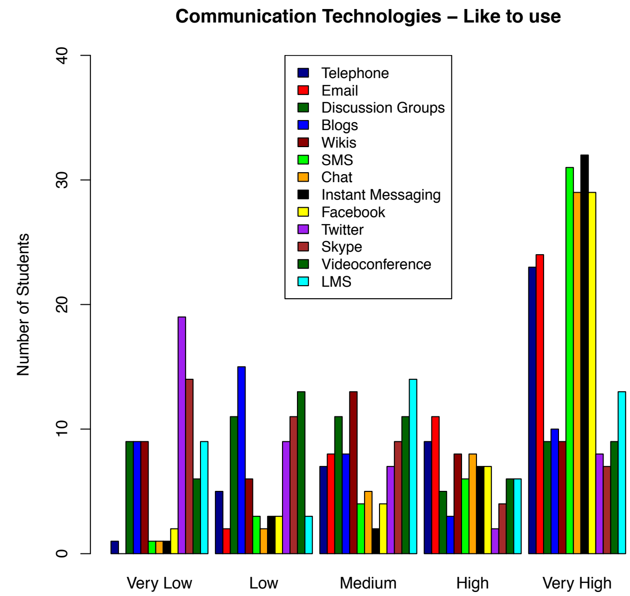
Figure 15: Know how to use the technology

complex in its essence. In the case of allocation and the sequencing is incomprehensible that can not be understood in the context of programming, since they are concepts that are part of daily life patterns. It is assumed that the experience might be transferable to the simplest understanding of scheduling problems that incorporate these concepts. However, it does not. It is an obstacle that students can not overcome, and to worsen matters further, in a conventional programming course, those subjects are taought at the beginning, impacting negatively on the remaining items on the course.

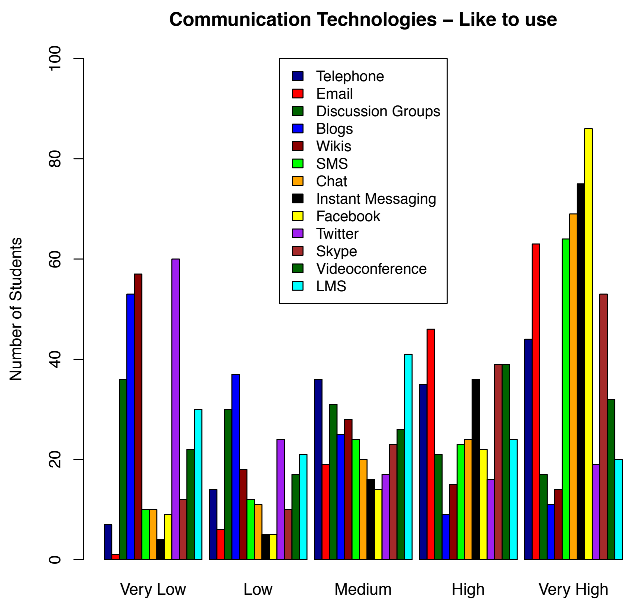
The test of mental models, initially proposed by Dehnadi [3], [2] and subsequently modified by Bornat, Dehnadi and Simon [1], [4], assess how students respond to questions related to assigning values to variables and sequencing. This operations are part of ever the simplest of programming tasks. It is proposed as a condition for applying the test that students are still unaware of the issues of variable allocation and sequencing with relation to computer programming The exercises that students must face are all multiple choice. The first three consider only one alternative. The remaining (9 exercises) should allow the possibility of more than one option, thereby trying to measure the concept of simultaneity and



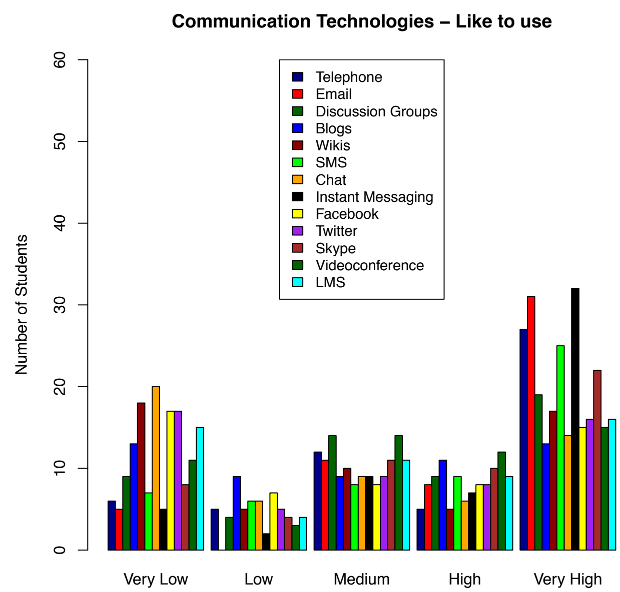
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

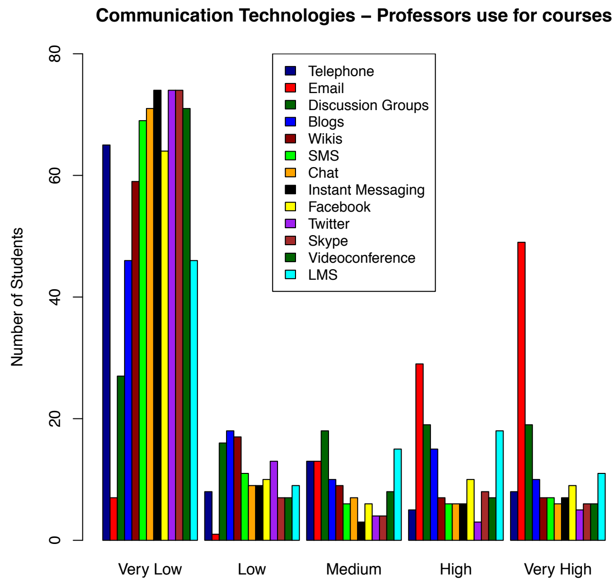
Figure 16: Like to use the technology

sequentiality. Unfortunately, this mechanism was not provided for in the multi-choice test applied to students, but as a first attempt, for the purposes of the IGUAL project objectives, was not necessary.

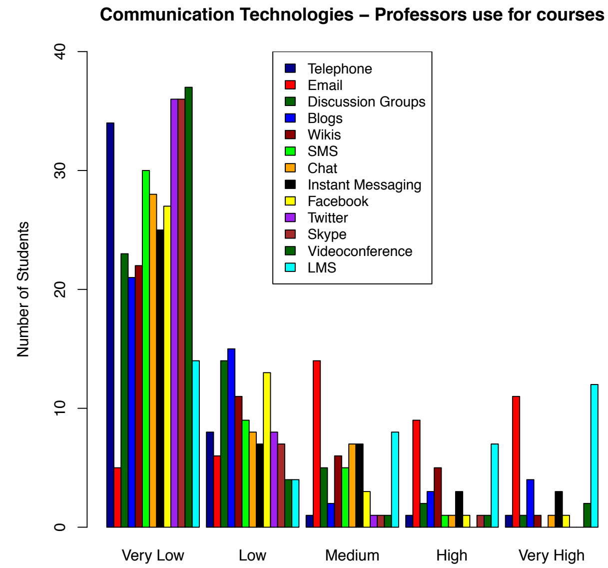
2) Found Models: The student responses can be classified into one of 11 possible models. The M2 model is defined as one that leads to the correct answer. Since there are 12 questions, the test is to establish the degree of "consistency" of students' responses, that is, it is said that a student has a consistent mental model (called C0) if you accumulate a minimum of 8 answers below the same model. Moreover, in order to classify lower levels of consistency, let you group similar pairs of cognitive models (classified as C1), or four similar models (C2) and also to 8 (C3). The latter has the weakest consistency.

For the purpose of the IGUAL project, it is interesting to classify students with consistency ranks C0 and C1 that did not have chosen the right answers. With the mental model defined, we can design strategies and activities to reverse the expected negative results in the programming course.

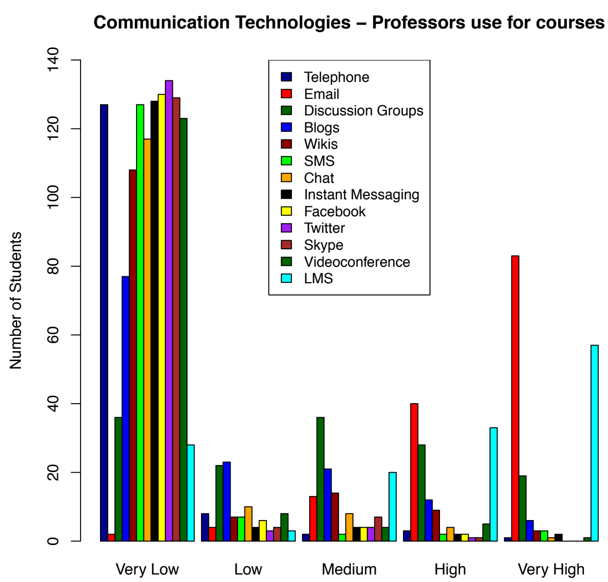
The Mental Models test was conducted in a computer lab, attended by two groups of students, separated in time, with each having approximately 30 minutes to answer the 12-question questionnaire. It is emphasized that they did not



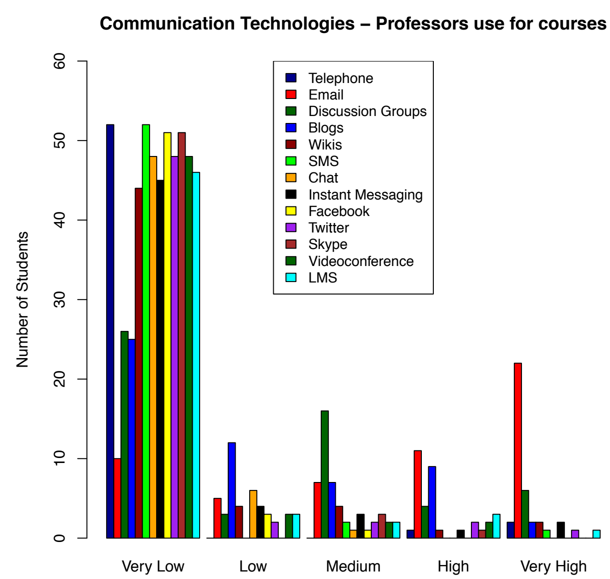
(a) ESPOL



(b) UAA



(c) UNIANDES

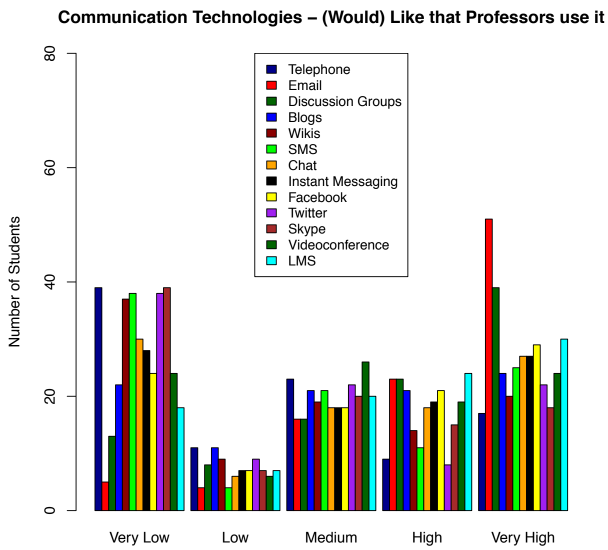


(d) UNIPAMPA

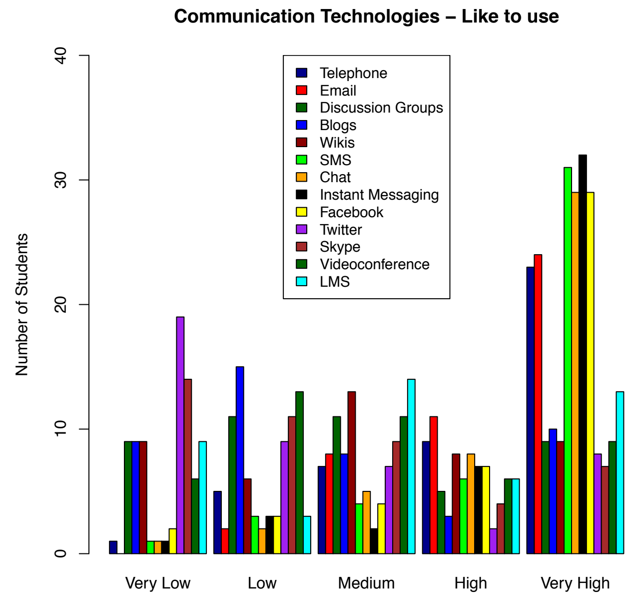
Figure 17: Professors use the technology for courses

know the subject. Any help or explanation was given before or during the Test. Below, the results are indicated:

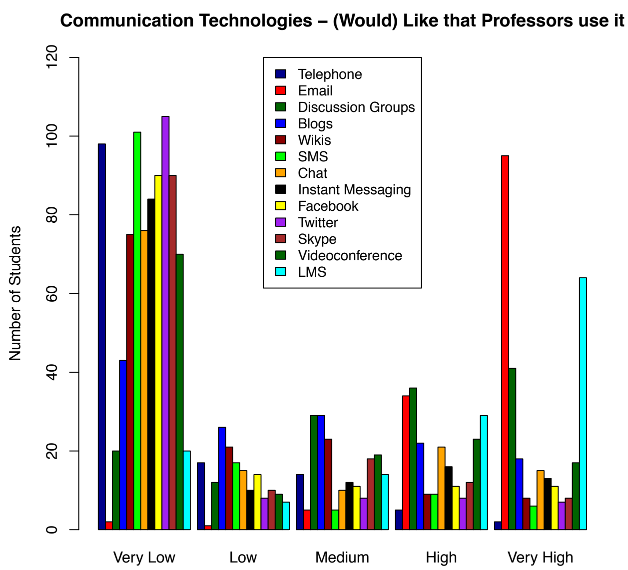
- More than half of students surveyed did not have previous experience in programming languages.
- This last analysis was important to correlate with the direct results (number of correct answers per student) obtained. Note that there were 22 students (43.13%) who got 0 correct answers, compared with 2, which received 10 correct answers.
- Regarding consistency C0, only 12 students were detected in this group. Of these, the models of importance to the project M4 and M9 (M2 corresponds to the model of successful students (5.9%) and therefore are not subject to analysis for the project).
- Model M9, with 7 identified students (13%), corresponds to a model where "nothing happens", i.e. the values of the variables in question remain unchanged, regardless of what happened. It seems that this group shows insecurity. One aspect to consider in designing activities. o Model M4 is the exact reverse of model M2 (right), that is, the operation is performed with the reverse assignment. Students who maintained consistent with this model (4%) could easily be recovered and added to the group of students from model M2.
- The rest of the students can be categorized as inconsistent (76.5%). Their answers do not follow a stable pattern and



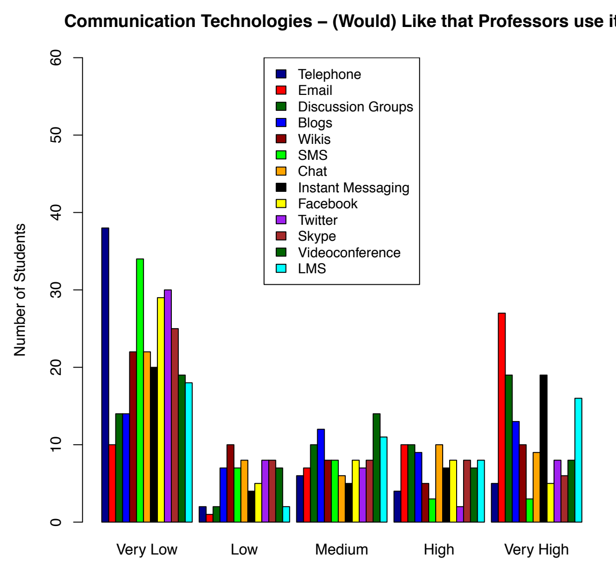
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 18: (Would) Like that the Professors use the technology in my courses

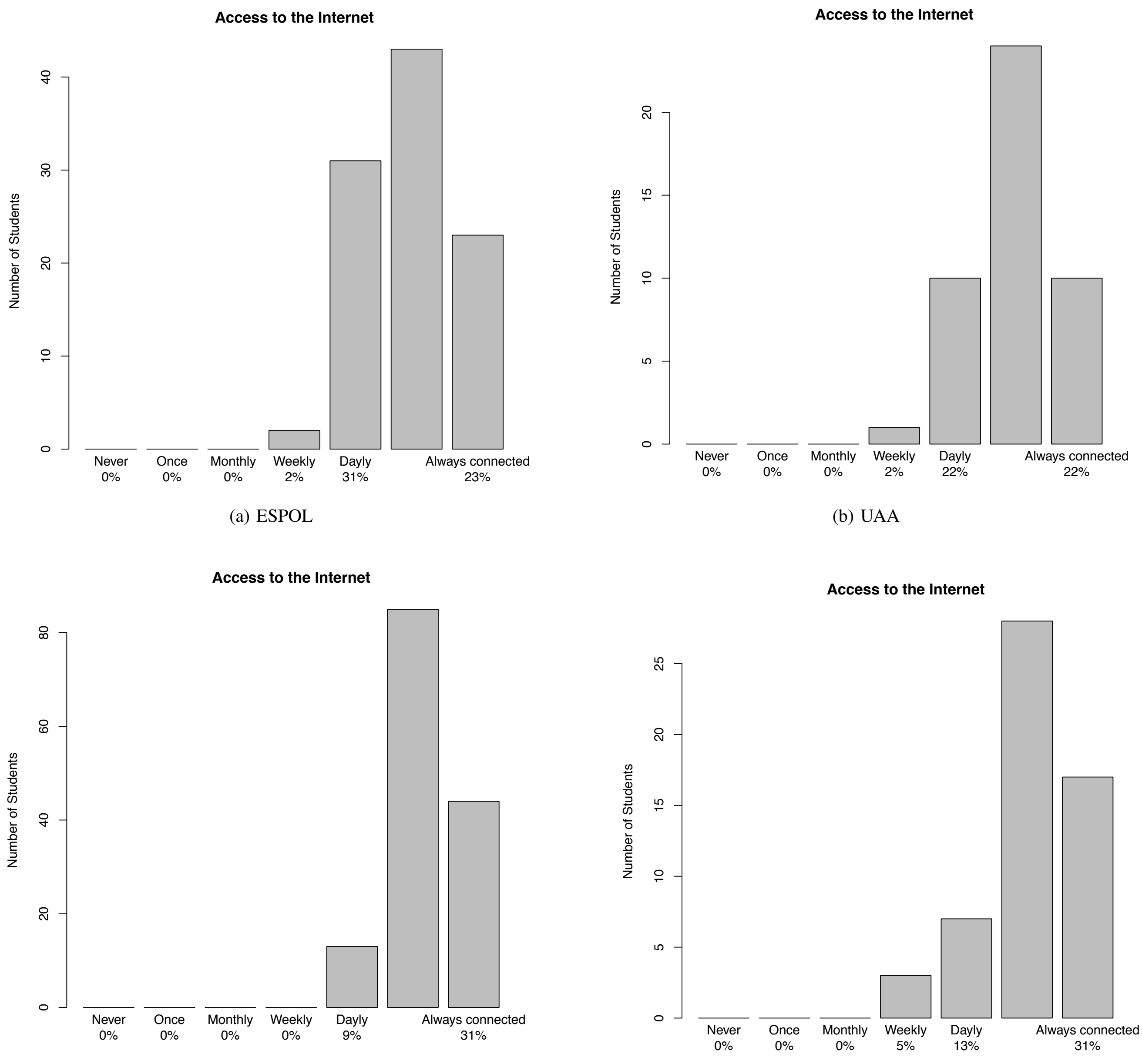
is not possible to obtain more information on this test.

3) Test Conclusions: Mental Models Tests is perhaps the only tool available and validated by its authors to detect patterns of thought that may be useful for the preliminary assessment of success for students in an introductory course in programming. The applying of questions related to allocation and sequencing in early stages to detect their mental models, facilitates the development of strategies to try to reverse the eventual failure of a significant number of students. In the case of the local sample, on which test was applied, it can be inferred that the test is accurate regardless of the type of student's previous education (private, semi-private, public), as the graph shows in Figure 4.

For consistency, which is the main point of the Mental Models Test, it seems that the results indicate the presence of a course largely inconsistent (This announce the a priori probability of failure of a high percentage of students?), so it is necessary to cross these results with the application of other tests or questionnaires to help clarify the profile of students, in order to apply the reinforcements as part of the IGUAL project.

D. Course Content

To determine which topics were the most difficult to learn to the students, a test was created to measure their skills in



(a) ESPOL

(b) UAA

(c) UNIANDES

(d) UNIPAMPA

Figure 19: Access to Internet

different basic topics of Introduction to Programming courses. This test was applied in ESPOL, UAA and UNIPAMPA using the C programming language as the language of the test. The test was also applied to UNIANDES, but using the Java programming Language. Six different topics, divided in 20 subtopics were identified by the professors of the institutions. The following is a complete list of those topics and subtopics:

1) Designing basic algorithms (in psedudocode) to solve a problem
   a) Order
   b) Make the program stop
2) Being able to define, change and use a variables
   a) Being able to differentiate between constants and variables
   b) Know the different data types and their uses
   c) Declare or define variables / constants
   d) Use of assign operator
   e) Recovery of the value from the variable
3) Being able to do arithmetical calculations using the language
   a) Know the rules of precedence and priority of operators (use of parenthesis)
4) Being able to do logical and relational operations
   a) Know the rules of precedence and priority of operators (use of parenthesis)
   b) Being able to do basic logic transformations

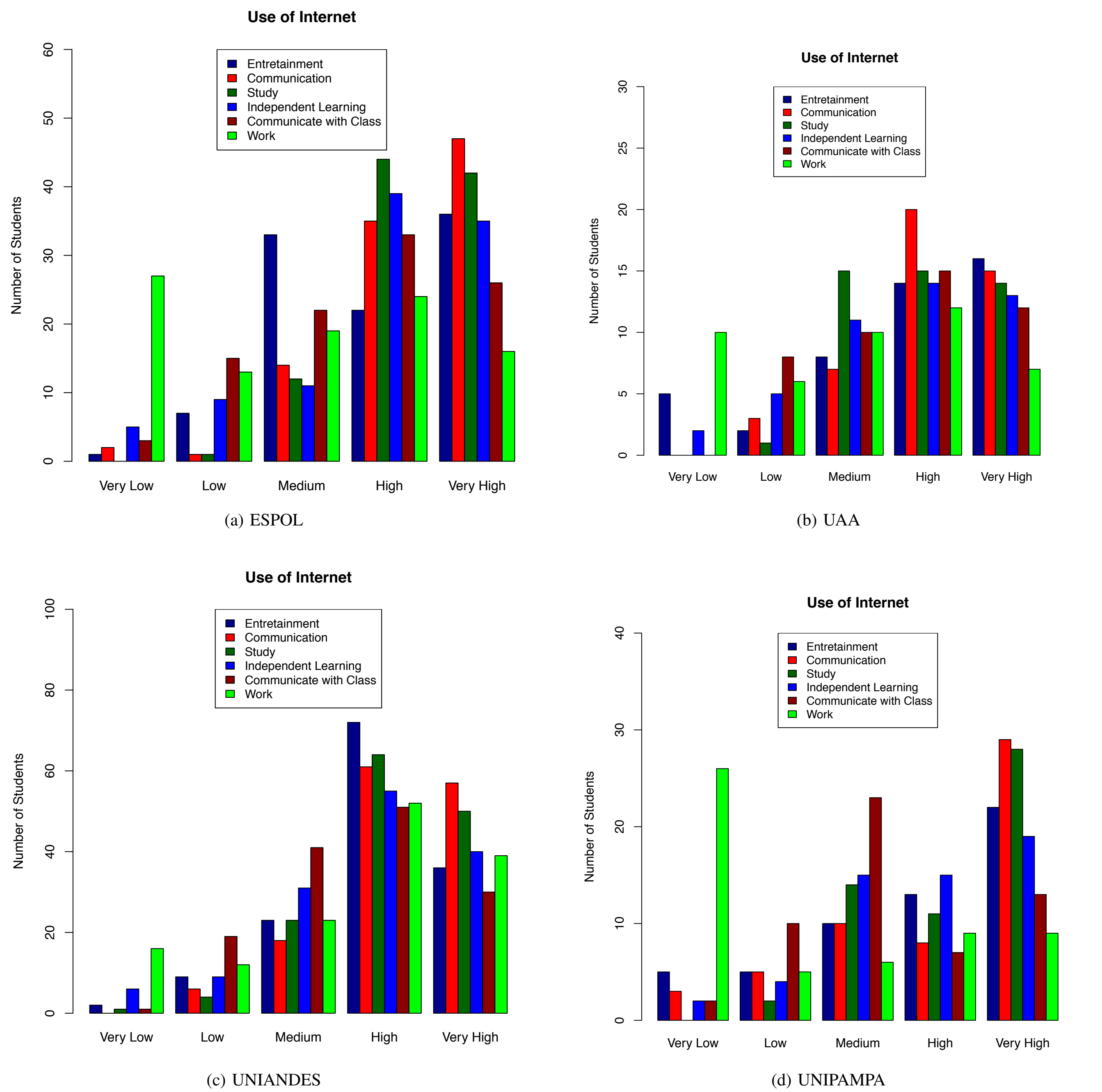
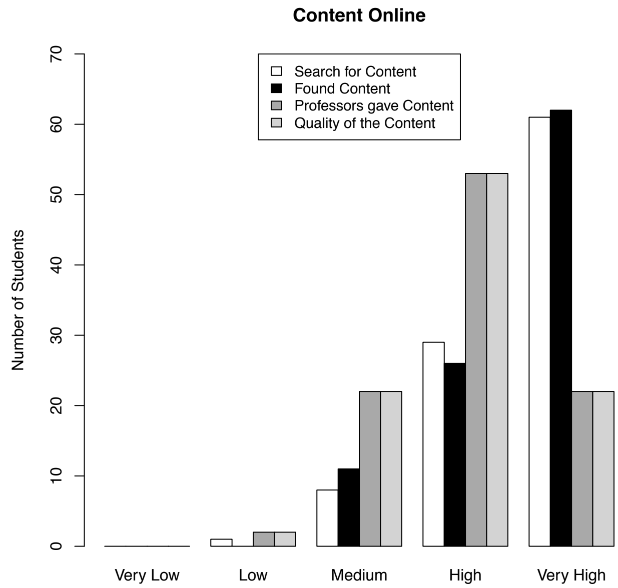


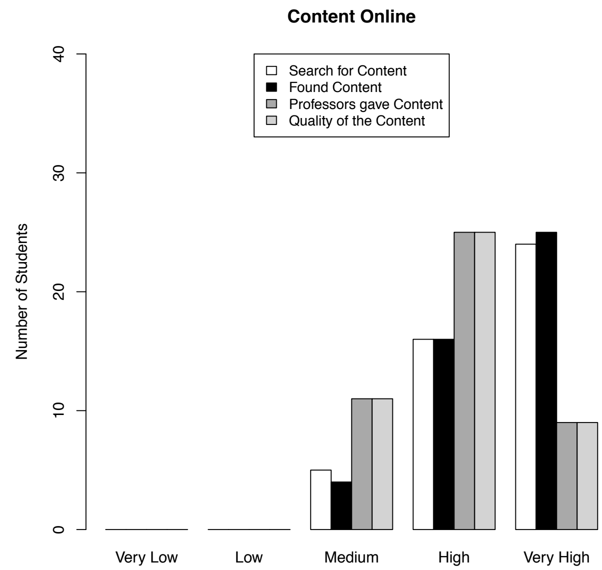
Figure 20: Use of Internet

5) Select and use of effectively Conditional (if / If - Else)
   a) Know how the Conditionals work
   b) Know the structure / syntax of the Conditionals
   c) Know how to use nested conditionals
   d) Know how and when to use advanced conditionals (Switch)
6) Select and use of effectively Iteration (For / While / Do While)
   a) Know how the Iterators work (counters and accumulators)
   b) Being able to select the correct iterator for the problem at hand
   c) Know the structure / syntax of the Iterators
   d) Know how to use nested Iterators
7) Being able to modularize the program through functions and procedures
   a) Understanding of prototypes (prototypes)
   b) Being able to call a function or a procedure

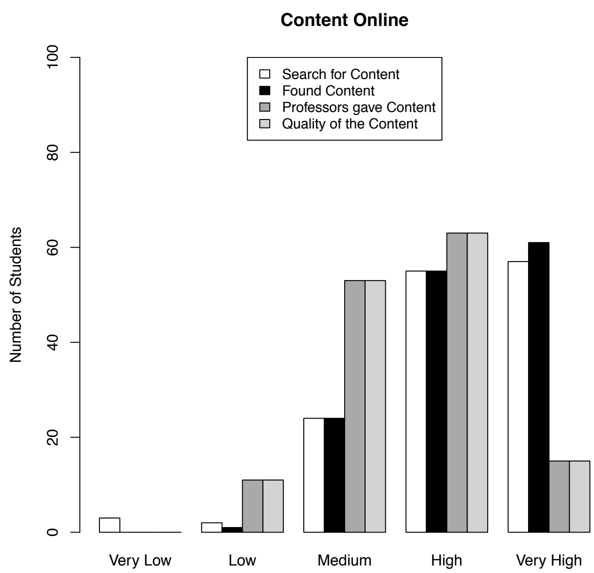
For each subtopic there were 2 questions, in the case of ESPOL, UAA and UNIPAMPA. In the case of UNIANDES, some topics had 3 questions. Each question gave 1 point if corrected correctly. Figure 23 shows the distribution of the score for each topic for the different institutions.



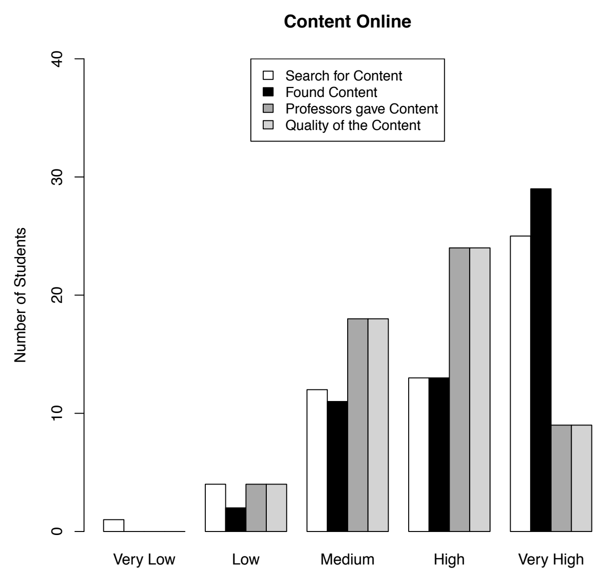
(a) ESPOL



(b) UAA



(c) UNIANDES



(d) UNIPAMPA

Figure 21: Educational Material Online

Topics that have more than 30% with score 0 or that have less than 10% with score 2 or 3 are considered problematic topics. From this analysis it seems that apparently easy topics, such as differentiate between a constant and a variable, select the right data type and rules of precedence are not well understood by students. Also, more complex topics as understanding the conditionals and being able to properly call a function are also difficult to solve for the group of students.

ESPOL will concentrate in create learning materials to facilitate the learning of these topics:

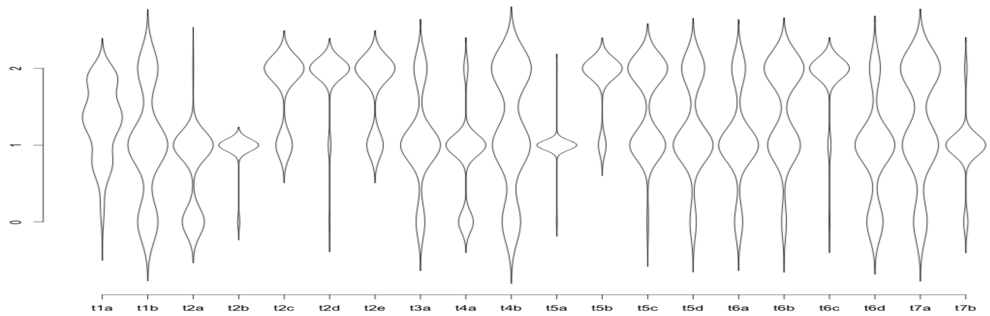
For UAA, the results indicate that the lowest frequencies of correctly answered questions of the content survey were the t2a, t4b, t6a, t6d (the lowest), t7a and t7b topics.

Subtopic	0	1	2
t2a. Differentiate between variables and constants	35%	64%	1%
t4a. Rules of precedence of logical operators	25%	68%	7%
t7b. Being able to call a function or procedure	9%	86%	5%
t2b. Different data types	5%	95%	0%
t5a. Know how conditionals work	2%	97%	1%

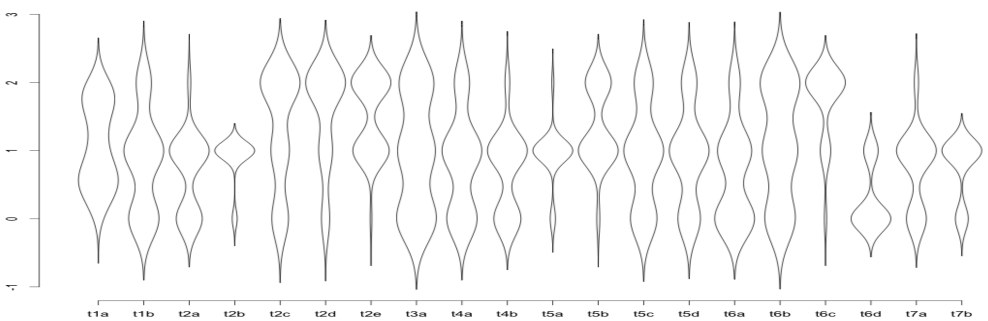
Table I: Problematic Topics ESPOL

For UNIANDES, the most critical topic in the student performance on the course is t4b. However, the next critical topics t3a, t5c and t6d seems to present problems also.

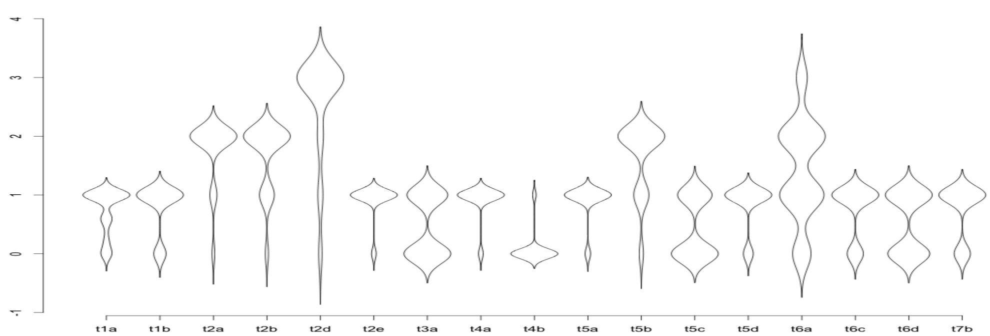
For UNIPAMPA, the top 10 critical topics are, in order of difficulty: t4b, t6b, t1a, t1b, t6d, t2a, t4a, t3a, t5c and t2b.



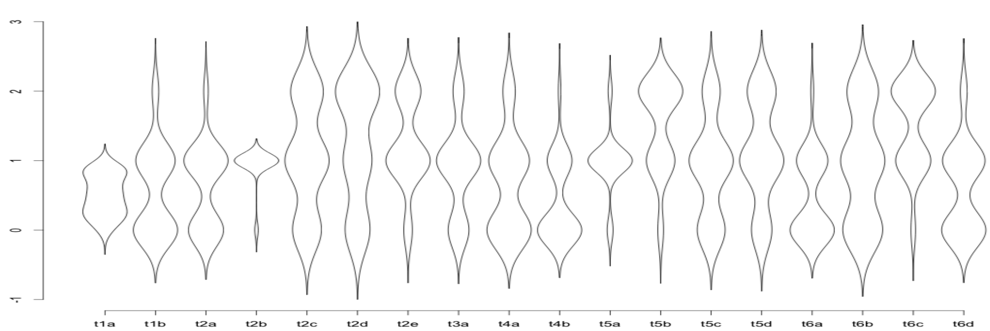
(a) ESPOL



(b) UAA

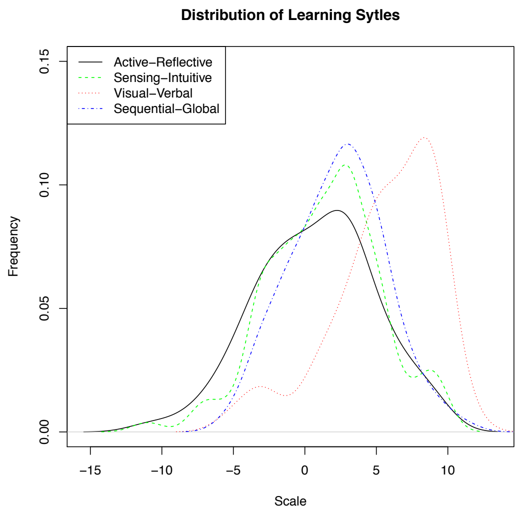


(c) UNIANDES

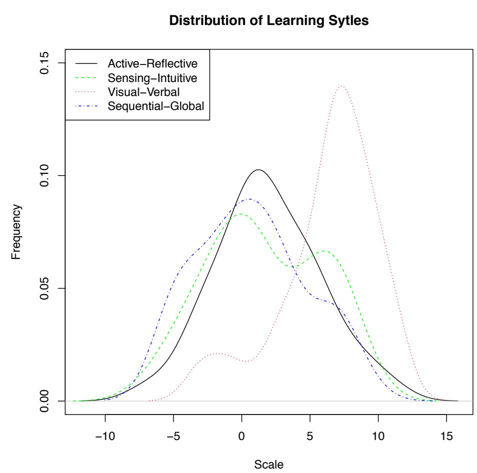


(d) UNIPAMPA

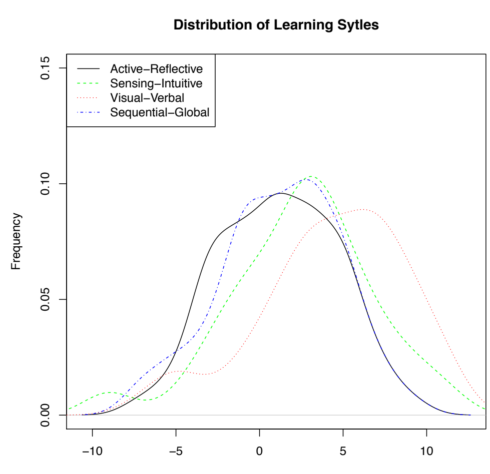
Figure 23: Distribution of Score per Subtopic



(a) ESPOL



(b) UAA



(c) UNIPAMPA

Figure 22: Distribution in the Active-Reflective axis

Subtopic	0	1	2
t6d. Know how to use nested iterators	73%	27%	0%
t6a. Know how the iterators work	47%	40%	13%
t2a. Differentiate between variables and constants	38%	58%	4%
t4b. Evaluate logical expressions	38%	55%	7%
t7a. Understanding Prototypes	31%	62%	7%
t7b. Being able to call a function or procedure	24%	76%	0%

Table II: Problematic Topics UAA

Subtopic	0	1	2	3
t4b. Evaluate logical expressions	93%	7%	0%	-
t3a. Know the precedence rules	53%	47%	0%	-
t5c. Know how to use nested conditionals	58%	42%	0%	-
t6d. Know how to use nested iterators	47%	53%	0%	-

Table III: Problematic Topics UNIANDES

IV. Conclusions

The main conclusions to be extracted from this Need Analysis are:

- There are two different groups taking the Programming Fundamentals course. One is composed by freshmen students that are in their first year at the University. The second group is made up by seniors that are about to graduate after 4 to 5 years of studies.
- The access problem seems to be disappearing, but there are still a minority of the students without access to computers before the University. The project should try to help those students.
- It is interesting to contrast the information about educational content and previous programming knowledge: the majority of surveyed students reported to have searched and found educational materials on line and that the quality of these was –to their perception– high, but most of them did not used the computer to learn to program. This can be an indicator that they did not found programming-related educational material that could have been useful for them.
- There is a perceive development in programming skills after taking the course, however, there still a lot of room for improvement.
- The technologies that students appreciate more for their studies are Email and Learning Management Systems. Although traditional, they are preferred to newer, more personal or more complex technologies. The project should try to make use of them.
- Most University students access the internet daily.
- Students are used to obtain learning materials online and use them for independent learning. This support the proposal of the project of using those materials to

Subtopic	0	1	2
t4b. Evaluate logical expressions	62%	35%	4%
t1a. Order	49%	51%	0%
t6d. Know how to use nested iterators	47%	45%	7%
t1b. Make the program stop	42%	53%	5%
t6b. Select the correct iterator	35%	38%	27%

Table IV: Problematic Topics UNIPAMPA

improve the learning process.

- To summarize the findings of the context survey, it can be concluded that the possible impact of the developed Learning Solutions of the IGUAL project can be high, because a large percentage of students already use computers at home, use the Internet, and they frequently search for and have used on-line learning materials in their learning environments.
- The Learning Styles results provide very clear guidelines as to which are the preferred ways of learning of the target population: they have a strong preference of visual learning materials and a moderate preference to Active, Sensing and Sequential ways of learning. So a premise in the context of the IGUAL project for designing Learning Solutions would be towards software that let the students ‘do’ (active), ‘feel’ (sensing), and ‘see’ (visual) step by step (sequential) pedagogical examples.
- Results of the test show that students had trouble selecting a correct iteration structure and knowing how to create them (two questions of topic 6). They also had problems with questions related to creating functions, which is a way –in C and most structured programming languages– to divide a program into smaller modules. This is a basic skill that has to do with dealing with a complex problem and then breaking it into smaller, less complex parts. Evaluating logical expressions is commonly a difficult subject and also a critical skill that is related to many programming topics (for example, making calculations and creating selection and iteration structures). Finally, at the most basic level, a surprising result is that students had also difficulties differentiating between a constant (a programming memory label whose value doesn’t change) and a variable (a memory label to store values that will be constantly changing), which intuitively seems to be a not too difficult subject. Results of the Content Survey clearly show what the most problematic topics are. This finding provides a useful design guideline to focus the development of Learning Solutions of the project.

This findings will be used to tailor the solutions that will be proposed by the Latin American partners inside the IGUAL Project.

References

[1] R. Bornat, S. Dehnadi, et al. Mental models, consistency and programming aptitude. In *Proceedings of the tenth conference on Australasian computing education-Volume 78*, pages 53–61. Australian Computer Society, Inc., 2008.

[2] S. Dehnadi. Testing programming aptitude. In *Proceedings of the 18th Annual Workshop of the Psychology of Programming Interest Group*, pages 22–37. Citeseer, 2006.

[3] S. Dehnadi and R. Bornat. The camel has two humps (working title). *Draft paper. Retrieved at http://www. cs. mdx. ac. uk/research/PhDArea/saeed/paper1. pdf*, 2(23):07, 2006.

[4] S. Dehnadi, R. Bornat, and R. Adams. Meta-analysis of the effect of consistency on success in early learning of programming. In *21st Annual Workshop of the Psychology of Programming Interest Group*, 2009.

[5] R.M. Felder and J. Spurlin. Applications, reliability and validity of the index of learning styles. *International Journal of Engineering Education*, 21(1):103–112, 2005.

[6] R. Ihaka and R. Gentleman. R: a language for data analysis and graphics. *Journal of computational and graphical statistics*, pages 299–314, 1996.

[7] P.N. Johnson-Laird and M. Steedman. The psychology of syllogisms. *Cognitive Psychology*, 10(1):64–99, 1978.

[8] C. Schmitz. Limesurvey (version 1.71)[computer software], 2008.