# Effectiveness of Research Modules in Undergraduate Curriculum

**W. Haque & K. Alagarsamy**
Department of Computer Science
University of Northern British Columbia
Prince George, BC V2N 4Z9
{haque, csalex}@unbc.ca

## Abstract

*In this paper, we discuss our experience and results in introducing research modules in some of the upper-division undergraduate computer science courses at UNBC. We have observed that inclusion of research component plays a significant role in the students' perception of the subject matter and greatly enhances their ability to pursue a more focused research area. Two distinct groups of courses have been used for this study, the regular $4^{th}$-year courses and strictly research-oriented courses. The research achieves a different objective within the two groups ranging from learning about research methodology to the development of innovative ideas and solutions to the problems. Our findings are particularly important for primarily undergraduate institutes, that is, those with very limited or no graduate programs.*

**Keywords:** Undergraduate research, research modules, undergraduate curriculum, research methodology.

## 1. Introduction

Integrating research into undergraduate curriculum has been an emerging trend in recent years [1-5]. This practice is becoming increasingly feasible, especially in computer science, due to the accessibility of literature through internet and rare need for sophisticated experimental set up or external study and data collection. The undergraduate courses at University of Northern British Columbia (UNBC) are generally classified as *lower-division* ($1^{st}$/$2^{nd}$ year) and *upper-division* ($3^{rd}$/$4^{th}$ year). Over the last several years, many upper-division undergraduate courses[1] have been offered with a research component which has resulted in a variety of positive outcomes. These include better prepared students, enhanced presentation skills, group experience, applied research with industrial collaboration and a much broader coverage of the areas. We have observed that inclusion of such research modules plays a significant role in the students' perception of the subject matter and greatly enhances their ability to pursue a more focused research area. In this paper, we discuss our experience and results in introducing such research modules in a variety of courses. These courses are broadly classified into two categories based on the role of the research component in the course content. We compare the knowledge gained in the courses that contain research modules as complementary topics with the courses which are primarily research-oriented. Our findings are particularly important for primarily undergraduate institutes, that is, those with a very limited or no graduate program.

The rest of the paper is organized as follows: In section 2 we present a classification of courses. Section 3 presents the delivery methodology. Knowledge and expertise gained in these courses is discussed in Section 4. Finally, we conclude the paper in Section 5.

## 2. A Classification

We classify the types of research modules incorporated in computer science courses at UNBC broadly into two categories as shown in figure 1.

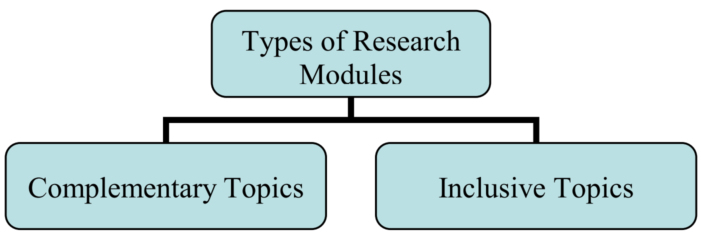


**Figure 1. Classification of Courses**

Complementary topics based research modules are introduced in the courses to broaden the knowledge of the subject matter presented in the course. Traditionally, these courses follow a textbook and are not required to contain a research component. Such courses include Database Systems, Parallel Computing and Computer Networks. A research module is added to each of these courses in the form of complementary topics selected from various areas of research. The students (working in teams) are expected to research the assigned topic and provide a detailed report followed by a presentation towards the end of the term.

The courses involving inclusive research modules are again classified into two categories: Upper-division research courses and Industry-oriented project courses. In the upper-division research course, the focus is mainly on depth rather than breadth of the topic area. These courses are primarily aimed at preparing the students for further studies or jobs in R&D

[1] Mostly at the 400-level.

environments. The topics are usually selected from emerging areas so as to maximize the chance of continuing research in the same or similar area for those who continue on to pursue their graduate studies. In the industry-oriented project course, several pilot projects were selected in consultation with the industrial partners. The aim of the course was to provide students with experience in real-world industrial applications of advanced technology and to foster technology transfer to industry.

## 3. Delivery Methodology

As described in the previous section, there are two distinct categories of undergraduate courses in which research has been integrated. The first category has its own full course load and prescribed textbook(s). The second category is the 499-type courses which are in fact listed in the calendar as project/research-oriented courses and are taught in a “seminar-like” fashion. Since the course delivery method for the two types of courses vary considerably, these are explained in separate sections which follow.

### 3.1. Courses with research modules as complementary topics

As indicated above, these courses carry a normal course workload including lectures, midterm(s) and final exams, assignments and projects. The majority of the grade comes from these components. The research module is assigned 15% of the grade. Teams of 2-3 students are formed based on their research interests and compatibility[2]. The topics are assigned in the 2$^{nd}$/3$^{rd}$ week of the term. Usually, a starter paper or a reference is provided (ACM/IEEE/SIGMOD, etc journals). The students submit a preliminary report in about 3/4 weeks which includes an extended summary of their work including a brief introduction to the topic, literature review, research methodology, important issues and approaches, future directions and work distribution. The students are provided a quick feedback on this report. To discourage procrastination, the preliminary report is now required to be fairly comprehensive and it is expected that all issues have been identified at this time. A final report is submitted 3-5 days prior to the presentations[3]. The final reports have typically been 25-30 pages long and include a complete bibliography together with a much detailed coverage of all the topics listed above.

[2] Students are invited to suggest topics of interest and preferences for partners; this allows for fewer troubles later on.

[3] All groups have the same deadline for submission of this report.

Presentations are 20-25 minutes long followed by a 5 minute question/answer period. The presentation time is shared by all members of the group. To ensure feedback and reasonable understanding of the topics, groups are also required to submit a critique of all peer presentations. Short questions from the presentations are also sometimes included on the final exam.

Students' research includes web searches, library searches, obtaining papers from the authors directly – that is, all components which are typical of a graduate program. Generally, the research is graded on its own merits, but a small percentage (5%) of the grade is also assigned to competition (relative grading). This has resulted in preparation and delivery of very professional reports and impressive Power Point presentations as groups try to show their skills by including interesting transitions and animations.

A general survey suggests that the students put in an average of 5 hrs per week for a total of 8 weeks. This amounts to 80-120 hrs of research per team (depending on size of the team). For a class with 20 students this represents approximately 800 hours of research for each course. It should be emphasized again that this time is in addition to the regular coursework that includes the exams and routine assignments/projects.

### 3.2. Primarily research courses (Inclusive Topics)

In the upper-division research course, the focus is mainly on depth rather than breadth of the topic area. The research component is assigned 100% of the grade which includes a final oral exam. The course is scheduled for one 3-hour slot per week. Teams are formed based on their research interest within the subject area. To reduce the coordination effort and to increase the ratio of individual contributions, the team size is normally restricted to two. The course is organized as shown in figure 2.

The initial part of the course (subject area introduction and research topics identification) is covered by the instructor for about first three weeks and the rest of the course is carried out by student presentations. During research topic identification session, the instructor introduces a set of carefully selected problems within the subject area and briefly discusses some of the published solutions for those problems during the lectures. The groups then select a research problem from the list provided. Due to time constraint, the instructor supplies all relevant references. There are two types of student presentations: reading presentations and research presentations.

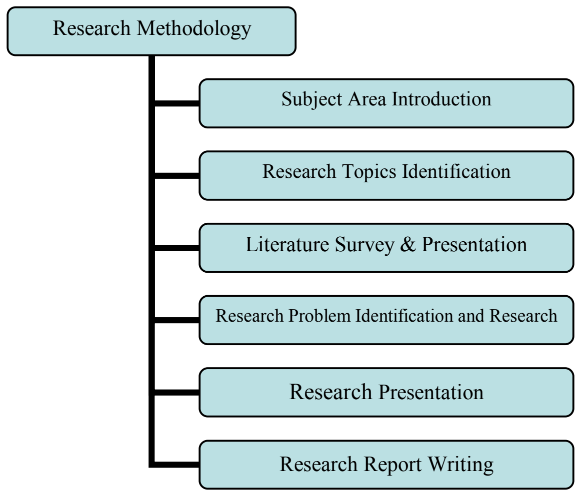


**Figure 2. Research Course Organization**

Reading presentation focuses on an already published work on the selected topic and the research presentation focuses on the students' own contribution to the identified topic. Each presentation is 45 minutes long followed by a 15 minute period reserved for questions and discussions. The presentation is structured in such a way that one student has to present the problem and solution and other has to present the correctness proofs and analysis.

At the end of the reading presentation, the group is required to come up with a list of sub-topics which could be open problems, extensions, improvements, or different analysis on the same topic for further research. In many cases, the instructor helps the group to identify such sub-topics. The research results obtained by the group on these sub-topics are discussed in their research presentations and then documented in technical report form. The students are advised to follow some standard format (ACM/IEEE, etc.) to prepare the research report. An oral exam is also conducted for each group at the end of the term.

In addition, an industrially-oriented projects course was offered on an experimental basis. In this course, several pilot projects were selected in consultation with the industrial partners. The aim of the course was to provide students with experience in real-world industrial applications of advanced technology and to foster technology transfer to industry. The partners contributed a token amount of money which was primarily used to provide incentives to the students such as tuition waivers, subscription to IEEE Computer Society, books and prizes for best projects. The pilot projects were then delivered to the partners who then decided whether or not to pursue the work in a more comprehensive manner. It should be noted that this experiment was oriented more towards a “project-type” course and is included here only for reference.

## 4. Attained knowledge-base

### 4.1. Courses with research modules as complementary topics

In this section, we cover the first category of courses, that is, regular courses which could have been taught without a research component. However, by including a small research module in these courses, the students are exposed to emerging technologies and standards without taking a dedicated research course. This results in an in-depth understanding of the concepts in at least one topic and a broader understanding of the other topics presented by peer groups. These topics are usually selected from an area which is either being researched extensively, dynamically changing or for which standards are emerging. The time constraints do not allow coverage of these topics in a regular course otherwise.

*Database Systems:* This is the first and the only database course currently included in our undergraduate curriculum. It is anticipated that this course will be split into two courses in future for a more evenly distributed workload and an in-depth coverage of some of the advanced database design techniques. Currently, the course covers a wide range of topics from basic database concepts to the design and implementation of a real-world application. In addition to the material covered in the class, students complete assignments and are expected to learn SQL on their own time to design and implement 2-3 projects on two platforms[4]. Finally, a research module is completed. Figure 3 illustrates the regular and research components for this course and also provides a sample grade distribution. It is also worth noting that the literature review is very comprehensive – in one case, the group listed over 40 references.

[4] Oracle and MS Access

**Regular Course Load (85%)**

**Topics Covered:**

- Introduction to Databases.
- Storage structure and access methods.
- Data definition and data manipulation languages. SQL.
- The Relational Model.
- Relational algebra and calculus.
- Query optimization.
- Database design - ER model, functional dependencies and normalization (1NF, 2NF, 3NF and BCNF)
- Recovery, Concurrency, Security and Integrity.

**Sample grade distribution:**

| | |
|---|---|
| Midterm | 20% |
| Final Exam | 35% |
| Projects/Assignments | 25% |
| Research Presentation | 15% |
| Quizzes | 5% |

A detailed grading scheme is provided for each component of the project(s) and the presentation.

**Research/Presentation (15%)**

**Sample topics:**

- Temporal and Spatial Databases
- B2B Protocols for non-compatible Databases
- Multimedia Databases
- Active and Deductive Databases
- Data Mining/Warehousing
- Web-database Integration
- Real-time Database Systems
- Expert Database Systems
- Statistical and Scientific Databases
- Object-relational Databases
- VLDBs
- Distributed Databases.

**Sample grade distribution:**

| | |
|---|---|
| Overall presentation | 15% |
| Extended Summary | 20% |
| Concepts | 15% |
| Research | 20% |
| Final report | 20% |
| Critiques | 5% |
| Competitive | 5% |

**Figure 3. Course components for CPSC-422 (Database Systems)**

*Parallel Computing:* This course is offered using the same principles as discussed above for Database Systems. The lectures concentrate on the typical textbook topics, the assignments reinforce the concepts and projects provide applied experience with MPI/OpenMP. In addition, the research component, which is worth 15% of the total grade, includes topics such as

- Parallel Simulation Techniques
- Topological properties of Honeycomb and Diamond Architectures
- HPC requirements of Grand Challenge Problems such as Human Genome project
- Parallel Computing in Bioinformatics
- Scheduling Algorithms for Torus and Mesh architectures
- Explicit Parallel Instruction Computing (EPIC) vs VLIW and Superscalar Architectures
- Systolic Architectures
- Distributed Shared Memory
- Dependable Network Computing

A detailed study of some of the above topics has revealed interesting facts. For instance, the group researching the EPIC architecture this semester identified problems with the Itanium chip and reasons for Microsoft's reluctance in marketing of Itanium even after investing half a billion dollars in its development. It was also interesting to know how AMD's Hammer chip forced an early release of Itanium and the advantages that it provides over the latter.

*Computer Networks:* The research component of this course has included topics such as:

- Intelligent Agents
- GB Ethernet over copper
- Optical Switching
- Virtual Private Networks (VPNs)
- ATM over SONET
- WAP vs VXML
- PKI
- The Carnivore project
- RDF vs XML
- Mobile computing
- Intranets/Extranets
- IPV4 vs IPV6
- 3G technology
- Voice over IP
- Mobile Agents and Security

### 4.2. Research-oriented courses (Inclusive Topics):

These courses are mainly aimed at preparing students for further studies or jobs in R&D environments. The topics are usually selected from emerging areas so as to maximize the chance of continuing research in the same or similar area for students who chose to pursue graduate studies.

*Upper-division research course:* Here we discuss one particular course, namely self-stabilization, taught by one of the authors which yielded surprisingly positive results. We present the experience gained and research results obtained separately.

The experiences gained in this course are as follows.

1. *Few concepts and more time to think and discuss*: This is a totally different experience for the students. The 3-hour slot with flexible freedom to take short breaks created a relaxed friendly atmosphere in the class.
2. *Reading a published research paper from the perspective of presenting it to others and identifying potential research problems in that topic for further research*: Almost every student in the class admitted that this was one of the hardest part and certainly different from reading a text book or lecture notes. Most of the students have to read the paper many times before they could understand the content.
3. *Reading presentation*: For many students, this was their first experience in standing in front of the class and presenting a technical content. However, the small and relaxed friendly class atmosphere helped them to overcome their inhibition and gained confidence for next presentation.
4. *Conducting research*: Doing research in one of the most complex and hot topic in computer science, during their undergraduate level, was a thrilling experience. Many admitted that they did not believe that they could produce such results. Once they started getting the initial results they were really excited and motivated to work more to improve their results. Most groups are still continuing their collaboration with the instructor to make their result publishable.
5. *Research presentation*: We could definitely observe the difference between this presentation and the previous reading presentation. This presentation was more

intensive, lively, and even emotional in some cases explaining about their moments of failures and successes.

6. *Research report writing*: Most of the students, from their reading experience of research papers, immediately understood the difficulty of presenting material in an easily readable and understandable form.

Though every group has undergone the same methodology, the challenges they faced in the individual stages vary. For some groups, identifying a potential research problem was immediate, but the solution turned out to be either extremely difficult or hard to obtain. In some cases, they realized that they are dealing with a long standing open problem. For other groups, identifying the potential research problem to work on was the hardest part. Once they identified the problem, the results were then obtained either immediately or with modest difficulty.

To give an idea of what types of research is conducted in these courses, we present some of the results obtained in self-stabilization [6] course below.

| Group | Research Topic | Research Problem identified | Result Status |
|---|---|---|---|
| 1 | Uniform dynamic self-stabilizing election algorithm [7]. | Bounding the sequence number used in the algorithm of [7]. | Devised a mechanism to bound the sequence number. |
| 2 | Self-Stabilizing Mutual Exclusion Using Tokens in Mobile Ad Hoc Networks [8]. | Removing the designated process used in the algorithm [8] to make it decentralized. | Could succeed in doing that and obtained some unexpected positive results. |
| 3 | Self-Stabilization Minimum Spanning Tree Construction on Message-Passing Networks [9]. | Self-Stabilizing algorithm for the construction of spanning tree with processes assigned to nodes instead of edges as used in one of the algorithm in [9]. | Obtained such an algorithm and shown their correctness through a series of examples. Formal correctness proofs are yet to be developed. |
| 4 | Self-Stabilizing Algorithms for Coloring Planar Graphs [10]. | Self-Stabilizing algorithm for coloring planar graphs using 4 colors instead of 6 as used in [10]. | Could not succeed in getting such a deterministic algorithm, but have come up with another algorithm involving backtracking. Correctness is yet to be tested. |
| 5 | A Self-Stabilizing algorithm for the maximum flow problem [11]. | Proving the algorithm of [11] to work for graphs with cycles which was left as a conjecture in [11]. | Could not succeed in proving the conjecture, but have come up with other nice observations. |

Finally, the pilot studies undertaken in the industry-oriented projects course focused on applied research concepts; the projects included analysis and development of pulp process reports, administrative tracking systems, intranet accounting solutions, and transportation scheduling systems.

As a result of exposure to research in this category of courses, many of our students have gained confidence, are motivated to go for graduate schools and have prepared good research proposals for various scholarships. Some students have already won NSERC awards for their proposal based on the research done in these courses.

## 5. Conclusions

The research modules included in two different categories of courses have met two different research objectives. In the first category where the research component is made a part of regular courses, the objective is to train the students in research methodologies, literature review, understanding emerging technologies, identifying issues and approaches, future directions, report writing, enhanced presentation skills and group experience. The research-oriented courses, on the other hand, stimulate innovative research as students come up with their own solutions to open problems or improve existing algorithms. Both approaches prepare students to take on advanced studies and provide them with a broader perspective of the field of computer science.

The research modules indeed increase the workload for the students, but we have received very positive feedback on the structure of these courses. The attendance has been exceptionally good and the students seem to learn a great deal without taking a specific research course. This allows students to attain a broader understanding of the subject material which is not just superficial as they are expected to study at least one topic in depth and the others at a level that they can carry out an intelligent conversation on the topic. At the same time, the students also become interested in taking a course which is primarily research oriented in order to get a more in-depth understanding of the subject(s). This methodology has the advantage of producing students who are better prepared for postgraduate studies and for R&D careers. In addition, they become more conversant with the current/emerging technologies and have more focused research directions. The students are excited and often come up with constructive criticism and suggestions for further improvements on that topic. The enhanced presentation skills and teamwork experience acquired during the process is certainly an asset. This training also opens avenues for industrial collaboration. Finally, we strongly believe from this experience that by carefully selecting the subject area and course delivery methodology, successful research at the undergraduate level is indeed possible.

## References


1. Anne Bezuidenhout, Integrating Research and Undergraduate Teaching, Teaching Excellence, 7(4),1996.
2. Kenneth W. Stier, Integrating Research into Undergraduate Coursework – To provide professional experiences, ASC Proceeding of the 32nd Annual Conference, 1996.
3. Shani Francis, Keith Schimmel and Neal R. Pellis, Integrating Research into the Undergraduate Curriculum – NASA's Microgravity Bioreactor, Session 3613, ASEE Annual Conference, 1999.
4. Deborah Coppula, Integrating Teaching and Research, PRISM, ASEE, 1979.
5. David Wilson, Striking the balance: Research in the undergraduate curriculum, Högskolan och Industri i samverkan. Konferens för Kemiingenjörrslärare, VII, Karlstad, May 2000.
6. Shlomi Dolev, Self-Stabilization, The MIT press, 2000.
7. Shlomi Dolev, Amos Isreali, and Shlomo Moran, Uniform Dynamic Self-Stabilizing Leader Election, IEEE TOPDS, 8(4), 1997.
8. Yu Chen and Jennifer L. Welch, Self-Stabilizing Mutual Exclusion Using Tokens in Mobile Ad Hoc Networks, DIALM'02, 2002.
9. Zhiying Liang, Self-Stabilization Minimum Spanning Tree Construction on Message-Passing Networks, MS Thesis, The University of Calgary, Canada, 2001.
10. Sukumar Ghosh and M. H. Karaata, Self-Stabilizing Algorithms for Coloring Planar Graphs, Distributed Computing 7(1),1993.
11. Sukumar Ghosh, Arobinda Gupta, and Sriram V. Pemmaraju, A Self-Stabilizing Algorithm for the Maximum Flow Problem, Distributed Computing, 10(4), 1997.


---


Name: Dr. Waqar Haque & Dr. Kuppuchamy Alagarsamy

Department: Computer Science

Institution: University of Northern British Columbia

Address: 3333 University Way, Prince George, BC V2N 4Z9

e-mail: {haque, csalex}@unbc.ca