ABSTRACT
This paper considers how gaming has been infused into the computing curriculum of institutions in the United States. To increase motivation of students and improve retention, many programs have begun using gaming in their introductory courses, as upper level electives, or as separate degree programs. The authors review the current use of gaming within curriculums and analyze the content of game development degree programs. Finally, the authors describe plans at their institution to incorporate gaming throughout the computing curriculum and present initial results.

Categories and Subject Descriptors
K.3.2 [Computers and Education]: Computers and Information Science Education – computer science education, curriculum

General Terms
Design, Experimentation

Keywords
Game programming, introductory programming, digital games, computer science education, computer game development, student motivation

1. INTRODUCTION
Nearly everyone involved in computing education has heard or read about the current state of affairs concerning enrollment within the discipline: national enrollment in computer science has dropped 70% between the fall of 2000 and 2005. Between the 2005/2006 and 2006/2007 academic years, enrollments fell 18%.[1] There may be many reasons for this enrollment crisis, including the dot com bust, the publicity of off-shoring of computer programming jobs, and the perception of potential, future students that computing is equal to programming. Future college students are no longer interested in studying computing because they perceive computing to be “boring.”[6][7] If we, as a discipline, hope to produce enough graduates to meet the demands of society, there is a need to stem and reverse the enrollment decline.

2. EDUCATION'S RESPONSE
To address the decline in computing enrollment, increase interest and retention among students, and to continue to improve how we educate students within the discipline, many universities have radically altered their approach to introducing students to the field of computing. Many of these have shown promising initial results. The Georgia Institute of Technology has chosen to have multiple entry paths into the discipline—a course that is taught within the context of manipulation of digital media, another that is taught within the context of robotics, and one that is contextualized around solving engineering problems in MATLAB.[4]. Research has shown that when students learn programming within a context (media-computation, robotics, etc.), especially a context they are already familiar with, they are more motivated to learn, and spend time on task beyond what is required for the class.[2][3][4]

Another approach taken to stem the declining interest and enrollment in computing among college students has been to introduce the context of “gaming” into the curriculum. Gaming has become a hot topic among computing educators, as can be evidenced by the number of submissions at SIGCSE and ITiCSE in the past few years. Current college students are very computer savvy and most have grown up playing games on desktop computers, on hand held devices, and/or on console gaming devices. Computing is truly ubiquitous in their world. Most see computing as a tool or entertainment device yet fewer have interest in studying computing to learn more about programming or “controlling” the tool. An exception to this is gaming, as many students have a desire to create their own game. This satisfies the creative and intellectual nature of the student, whereas older, text based programming courses do not.

This paper reviews the current state of gaming within the computing curriculum, primarily at institutions within the United States. In the following section of the paper we review the different approaches currently being used to incorporate gaming: as a context within CS1, as stand-alone upper level elective courses, and as an entire degree program. This is followed in section 4 by the authors’ current experience of merging all three of the approaches at their institution. Section 5 contains information about the early results of the experience, and Section 6 provides conclusion and plans for the future.

3. INCORPORATING GAMING
While the jury may still be out on whether or not it is “good” to include gaming into the computing courses [10][11], it is now nonetheless being experimented with at many schools. As
categorized by Wolz et. al, [12] the approaches fall into four main categories: (1) to support foundation courses such as CS1, (2) to provide specialized content in an upper level course, (3) to provide curriculum encompassing a thematic approach, and (4) to provide trans-disciplinary experiences. As many institutions now offer an undergraduate degree in gaming, we combined categories 3 and 4 as most instances within these categories fell into schools which offered a degree program in digital gaming. The authors wanted to research the differences within each of the categories.

3.1 Institutional Discovery

The first step in our discovery process was to find a list of faculty and institutions developing or offering curriculum in game and media development within computing. We limited our review to undergraduate programs in the USA, though others are infusing gaming in graduate programs and there is work being done internationally. We examined conference papers, workshops, and program committee members from the following venues to determine a list of faculty and institutions offering game related courses:

2. IGDA Education Summit (2008)
3. Living Game Worlds (2007)
7. Journal of Game Development (Editorial Board)
8. IGDA Education SIG Wiki Curriculum page
9. gamecareerguide.com listing of schools

While not an exhaustive list, these are active venues for publishing research and curricular development related to game and media design/development within computing. We acknowledge that some programs may have been overlooked in our process, and we sincerely apologize to faculty that are doing work in this area but not represented in this study. We encourage such faculty to contact us – we would love to hear from them and learn what they are doing.

This examination process yielded 44 programs; we note that three institutions offered two or more programs (for example, a BA and a BS).

Next, we examined the degree and curriculum descriptions for the represented institutions and gathered a count of the course (or credit hour) requirements for the upper-division portion of the programs for each institution; we limit our examination to upper-division courses so that we exclude “general education” courses from our analysis and focus on program-specific courses. We realize that some lower-division courses may be central to these programs, but most program-specific material is covered later in the programs.

Categorizing the courses required for each program yielded three groups. For some institutions, only a few courses related to gaming were offered as “special topics;” others offered specializations or certificates within a CS or other degree, but the entire degree was not completely related to gaming; still other institutions had complete 4-year degree programs related to gaming. Figure 1 shows the categorization of each type of program we discovered in our search.

![Figure 1 - Program Distribution](image)

The next three sections discuss in detail our findings in each of these three categories of gaming within computing.

3.2 Gaming in CS1/CS2

Many institutions are using gaming as either a complete context for their CS1 courses or as themes for their assignments. References to using games for learning concepts in CS as well as motivating students can be found as early as 1992[13]. In 1995, Holliday wrote about using games as assignments for introductory programming.[14] As technologies and tools have changed, so has the sophistication of the games that students play. Text based games are no longer seen as “motivating” to students that grew up with Nintendos and Playstations. The sophistication level of the games used in CS1 has had to grow as well.

One solution is to use existing game making software for the course. This is the approach taken by Chamillard [15] who uses GameMaker, The Games Factory, and Pie 3D Game Creation System before moving into Java. Another solution is for the instructor to write the game framework; students then write their assignments on top of this framework. This has been done in the development of greenfoot [17], by Baylis and Strout in the RAPT program [18], and Frost with the Ucigame package [19]. Lewis & Massingill use this approach in CS2 [20]. Sung[21] has developed a library that sits on top of the XNA Framework that is used specifically for gaming assignments in CS1 and CS2. Leska and Rabung use games for CS1 projects [24].

3.3 Upper Level Electives and Specializations

Many institutions also offer one or more upper level courses that involve gaming. Gaming is used as a theme to teach multi-agent systems[22], comprehension of graphs[23], and secure communication protocols[25] among others.

Barnes et. al. [16] use existing game engines (GameMaker, Unreal Tournament 2004, NeverWinter Nights, 3D GameStudio, and RPGMaker) in a upper level course to design games for introductory students to play to learn more about programming. Parbery [26] offers a senior level capstone course that involves designing and implementing a game.

In our program analysis, the following institutions were identified as offering upper-division game related courses as electives within their computing programs:
- California State University Long Beach
- Carnegie Mellon University
- Columbia University
- Harvey Mudd College
- Marist College
- Northwestern University
- NYU Media Research Laboratory
- The College of New Jersey
- University of Michigan
- University of New Mexico
- University of North Texas

Still other institutions are identified as offering a certificate, minor, or specialization within their broader computing degree program. Such programs have multiple courses in gaming but not an entire program dedicated to gaming:

- Arizona State University
- Ivy Tech Community College, Bloomington
- Michigan State University
- North Carolina State University
- University of Louisiana at Lafayette
- University of North Carolina, Charlotte

### 3.4 Full Game Related Degree Programs

For full game related degree programs, we continued our analysis by examining the content of each program. We categorize the programs’ content into three groups: art/humanities, computing (i.e., traditional computer science), and game specific. Each program’s upper-division content was examined to determine which course hours fell into each of these three categories. Capstone, studio, and senior project courses were placed within the category that was most closely related to the program’s description, but more often than not, these courses were categorized as game-related since the intent of these programs was game focused. Table 1 provides examples of our categorization.

#### Table 1 - Program Course Categorization

<table>
<thead>
<tr>
<th>Category</th>
<th>Course Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game-related</td>
<td>Game design, level design, game business, game analysis, game engine development</td>
</tr>
<tr>
<td>Computing</td>
<td>Traditional CS courses such as data structures, operating systems, graphics, HCI, algorithms, software engineering</td>
</tr>
<tr>
<td>Arts/Humanities</td>
<td>2D and 3D modeling, art, animation, sound/audio engineering</td>
</tr>
</tbody>
</table>

Since some programs listed course credits while others only listed courses, we normalized the data on a per-program basis. This yielded a percentage contribution to each program for each of the three categories: game, computing, and arts. Figure 2 shows a ternary (triangle) plot of the 21 full degree programs and their relative (%) emphases on gaming, computing, and arts (one program was excluded because it was completely customizable and thus didn’t lend itself to our analysis). For example, if a point resides in the lower left corner, this program would emphasize arts with little or no computing or gaming; if a point resides in the top corner, this program would emphasize computing with little or no arts or gaming; and if a point resides in the lower right, this program would emphasize gaming with little or no arts or computing.

![Figure 2 – Full Game Program Emphases](image)

From this analysis, we observe that program emphases in gaming range from 9% to 67%, emphases in computing range from 0% to 67%, and emphases in arts range from 0% to 96%.

As identified by the International Game Developers Association, many game programs stem from existing computing or art programs; the data collected in this study supports this sentiment in that five institutions offer no computing component to their courses’ content and three institutions offer no art components to their course content. Of the remaining 13 programs, nine cluster in offering 40% or more gaming and four cluster in offering 40% or more art.

### 4. OUR MOTIVATION

The authors are currently responsible for redesigning and reinventing courses within our computing curriculum to include a gaming based context throughout the curriculum. By reviewing how others have incorporated gaming into their curriculum we hope to learn from them. It is our hope to improve student motivation within the discipline, especially by having students spend more time on task than they do in a standard computing curriculum. We also hope to improve retention rates. Failure rates in introductory programming classes are rumored to be quite high. Bennedsen and Casperson reported a 33% failure rate in CS1 from their survey. [27] At our own institution, the failure rate in CS1 has ranged from 50% to 22% in the last 6 semesters. By improving student motivation and time spent on task, we hope to improve the success rate of the course. We are also planning to include both mobile and casual gaming as well as simulations into the curriculum to expose the students to more than just a console or desktop experience. We can also broaden student exposure to multiple platforms each with different hardware requirements; Sony’s new Playstation-edu program would allow programs to offer development and architecture courses using very relevant hardware (the PSP and the PS2) [30]. By including games types other than the typical first person shooter games or attack based games we also hope to draw more underrepresented populations into the discipline. The bottom line is that we hope,
in this time of decreasing enrollments, to engage the students and motivate them to see the application of computing as an interesting field.

5. EARLY RESULTS

At the point of this writing, the authors have redesigned the CS1 course at their institution to be centered around a gaming context. The course includes 3 lecture hours per week and one 2 hour closed lab. The labs consist of both traditional introductory programming topics as well as gaming topics such as “what makes a good game” and “object-oriented design of games;” the course content conforms to the ACM Curriculum Objects-first coverage of core topics. The closed lab is done in a gaming lab that consists of 14 Xbox 360s connected to development computers with Visual Studio 2005 and the XNA Framework 2.0. The labs have been designed to allow students to learn the programming concepts while implementing games. Current enrollment in the introductory programming course is up almost 60% compared to last fall. We added another section of the gaming based introductory course to satisfy demand of the students. Students who are not required to take a programming course have enrolled in the course, indicating they just wanted to “see what it was about.”

While it is still too early to have conclusive evidence on the overall results of this initial course, the instructors have noticed anecdotal evidence supporting the goals of the course. The overwhelming consensus is that the students are engaged and are having “fun.” Several times, after the conclusion of the designated lab time, we have had to ask the students to leave lab…they wanted to continue to explore the code and play around with the games beyond what was asked of them in the lab assignment. Most students explore and experiment with the code beyond the scope of the assignment, and they are willing to learn and “put up with” much more complexity than one would normally ask of a beginning student. Some of the comments made by students after the first two labs include:

- “Wow, this programming stuff is fun.”
- “I’m so proud of myself. Look what I made it do.”

Engagement can be defined as an “authentic emotional experience”[28] or be to “emotionally involved”[29]. Students who are having fun while exploring and experimenting with code could be considered to be emotionally involved, especially with comments referring to personal feelings related to their effort, e.g. “proud.” The authors consider the majority of their students are very much engaged in not only the course, but more importantly, the discipline.

Students have already begun asking about the next course and whether or not it will be gaming based. The authors are looking forward to the remainder of the term.

6. CONCLUSIONS AND FUTURE WORK

Engaging current students in our computing programs and attracting new students to the field of computing is of paramount importance. Many institutions have begun utilizing game-based curriculum to motivate students and situate students’ learning in a familiar context. This study presented our analysis of the motivation of using gaming and a survey of institutions that have incorporated gaming into their introductory courses, into specializations and upper-division electives, and into full game-related programs. Our findings demonstrate that there is a wide range in institutional emphases with respect to computer science, gaming, and art/humanities, though patterns do emerge.

We will continue this work by expanding our game emphasis into CS2 next semester and into further CS courses in the future. It is our intention to add distributed and mobile computing content into CS2 by allowing students to program the Microsoft Zune media player using the XNA 3.0 framework. Additionally, we are in the final stages of proposing a full game design and development degree to begin in the Fall of 2009 to provide increased flexibility to our current and future students.

We would also like to expand the scope of this game analysis by considering international and graduate programs.

7. REFERENCES


[29] Green and Brock, 2000: Engagement is to be "emotionally involved."