The Committee will meet at 2:00 PM on Tuesday, April 2, 2002 in Humanities room 346.

Agenda

1. Consideration of Minutes of the March 19, 2002 meeting
2. Chair's Report
3. New Course EN-222 New Media Journalism (see attachment 1)
4. Changes in CS courses and new course CS-220 Discrete Structures (see attachment 2)
5. Old business
6. New Business
**New Media Journalism**
Course Proposal
Dept. of English

1. **Course Number**
   EN-222

2. **Course Title**
   New Media Journalism

3. **Course Description for College Catalogue**
   Students will learn to produce high quality journalism in an interactive environment oriented toward a Web-based readership. They will report, write and edit news stories through electronic media: stories will be both researched and distributed through the Internet, using a variety of new technologies such as Web sites and PDAs (Personal Digital Assistants). Students will rely upon digital technology in all phases of production. As part of their course work, they will contribute to the on-line edition of the campus newspaper, *Communique*.

   This course fulfills an elective requirement for the New Media Information Technology Degree Program.

4. **Prerequisites and Corequisites**
   EN-102

5. **Hours and Credits**
   3 hrs. 3 credits.

6. **Rationale**
   New Media Journalism is part of an integrated, interdisciplinary curriculum for new media and information technology that offers students advanced and specialized training in writing. It helps prepare students for high tech careers in multimedia communication environments. It offers them an opportunity to write in a variety of contexts: from the depth and breadth of print journalism to the speed of radio and the production values of television.

   Course will be offered on request by ECET Department.

7. **Outcomes**
   Students will practice interactive multimedia strategies to gather and report the news. They will learn to analyze and interpret information from diverse perspectives, such as text, audio and video. They will produce basic news stories as well as longer, interactive features, building a cyber-portfolio of their work that demonstrates mastery of new media communication and technology skills. They will learn how to write within the developing conventions of electronic discourse; how to collaborate in electronic environments; how to
navigate the World Wide Web and critically assess the information they gather; participate in building a course Web site.

8.  **Assessment**
The usual performance indicators of an upper-level writing course (EN-200 series), including quantity and quality of stories produced, adherence to deadlines, contributions to all phases of production, assigned papers and final exam.

9.  **Course Outline**

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The Role of the Journalist in a Multi-Media Society</td>
</tr>
<tr>
<td>2</td>
<td>Communication Techniques in Print and Electronic News Media</td>
</tr>
<tr>
<td>3</td>
<td>The Relationship Between News Reporter and News Consumer</td>
</tr>
<tr>
<td>4</td>
<td>The 5W Lead; Alternative Leads</td>
</tr>
<tr>
<td>5</td>
<td>The Inverted Pyramid</td>
</tr>
<tr>
<td>6</td>
<td>Accuracy and Fairness</td>
</tr>
<tr>
<td>7</td>
<td>Sources: Interview and Attribution</td>
</tr>
<tr>
<td>8</td>
<td>Text-Based and Electronic Sources: Documentation</td>
</tr>
<tr>
<td>9</td>
<td>Structure and Style for Writing on the Web</td>
</tr>
<tr>
<td>10</td>
<td>News Production on the Web</td>
</tr>
<tr>
<td>11</td>
<td>Interactive Feature Writing</td>
</tr>
<tr>
<td>12</td>
<td>The News Image: Photographic, Video, and Digital</td>
</tr>
<tr>
<td>13</td>
<td>Investigative Reporting</td>
</tr>
<tr>
<td>14</td>
<td>News Beats</td>
</tr>
<tr>
<td>15</td>
<td>The Ethics of New Media Journalism</td>
</tr>
</tbody>
</table>

10.  **Non-Traditional Methods of Instruction**
New Media Journalism will be conducted in networked writing classrooms. Discussions, assignments and conferencing will be conducted both face to face and in on-line environments like Blackboard.

11.  **Texts, References and Aids**
New Media and Internet Resources for Journalists  
(http://www.journalism.berkeley.edu/resources/newmedia.html)

U.S. Freedom of Information Act  
(http://www.journalism.berkeley.edu/resources/FOI/foia.txt)

Citizen's Guide on using the Freedom of Information Act  
(http://www.journalism.berkeley.edu/resources/FOI/citizens.txt)

Electronic Frontier Foundation  
(http://www.eff.org/)

Freedom Forum First Amendment Center  
(http://www.fac.org/)

New York New Media Association  
(http://www.nynma.org/)

Penenberg, Adam L. “Cyber-Vandal Plays Editor at Orange County Register’s Web Site”  
(http://www.inside.com/story/)

11.  **(Cont’d.) Bibliography for the course**
12. Curricula into which the course would be incorporated
   New Media Information Technology Degree (AAS).

13. Transferability as an elective or course required by a major to senior colleges.
   Not pertinent.

14. Faculty
   Linda Reesman
   David Shimkin

15. Facilities and technology availability:
   Humanities Bldg., Room H-407 or allocated space for Smart classrooms in Humanities Bldg.

16. Courses Withdrawn
   None
To: Dr. David Lieberman, Chair  
College Curriculum Committee  

From: Dr. Mona Fabricant and Dr. Joseph Bertorelli  
Dept. of Math & Computer Science  

Date: March 13, 2002  

Re: Proposal for Changes in Computer Science courses  

The following proposals have been discussed and approved by the Department of Mathematics and Computer. The rationale for the requested changes is continued articulation with Queens College. As of Fall 2002, Queens College will require all computer science students entering their junior year to take and pass a computer science proficiency exam. The proposal requests changes in current courses CS-101, CS-201, CS-203 and the introduction of a new course CS-220. These four courses and MA-471 will comprise the material that will be covered in the proficiency exam.

Proposal 1. A Change in Title and Course Description

from: CS-101 Introduction to Computing  
Co-requisite: MA-441  
3 class hours,  
2 lab hours  
4 credits

Introduction to problem solving methods and algorithm development; designing, coding, debugging, documenting programs in a high level programming language.

to: CS-101 Algorithmic Problem Solving 1  
Co-requisite: MA-441  
3 class hours,  
2 lab hours,  
4 credits

Primitive data types; single and multidimensional arrays; strings; control structures; basic I/O; subprograms and parameter passing; references; scope; introduction to recursion; designing, coding, debugging and documenting programs in a high level language.

Rationale. The programming language used at Queens College in courses corresponding to CS-101 and CS-203 has changed from Java to C++. The emphasis in the course is on the development of algorithms. The change is more descriptive of the course content.
Proposal 2. A Change in Title, Hours, Credits, Pre-requisites and Description for CS-203

from: CS-203 Introduction to Computer Science 3 class hours, 3 credits

Pre-requisite CS-101

Continuation of CS-101; multidimensional arrays, graphics, introduction to data structures

to: CS-203 Algorithmic Problem Solving 2 3 class hours, 2 recitation hours, 4 credits

Pre-requisite: MA-441 and C or better in CS-101

User defined data types, pointers and linked lists, ADT’s, stacks, queues, recursion, searching and simple sorting, elementary memory management. Object oriented problem solving.

Rationale. Queens College has increased the hours for the second C++ course to three class hours, two recitation hours, four credits to allow coverage of extended material. The new course description follows a similar description for the equivalent course at Queens College.

Proposal 3. A Change in Title, Pre-requisites and Description

from: CS-201 Computers and Programming 3 class hours, 2 lab hours, 4 credits

Pre-requisite: CS-101

Introduction to machine and assembly language; computer structure; digital representation of data and instruction execution; programming assignments in the computer laboratory to illustrate concepts.

to: CS-201 Computer Organization and Assembly Language
3 class hours,

2 lab hours,

4 credits

Pre-requisite: C or better in CS-101

Computer organization; machine representation of data and instructions; assembly language; control statements; data manipulation; subprograms and macros; system calls and interrupts; program interaction through peripheral devices.

Rationale: All introductory CS courses at Queens College require a C or better to advance to the next course.

Proposal 4. The Introduction of a New Course.

CS-220 Discrete Structures  3 class hours,  3 credits
Pre-requisite: MA-471

Recursion, recurrence and generating functions; relations; graphs and applications; asymptotics; trees; applications in computer science.

Rationale. This course is a continuation of MA-471 and contains material covered on the Queens College proficiency exam mentioned above.
Proposal for a Change in an Existing Course

1. Course number CS-101 (no change)

2. Course title  
   from: Introduction to Computing 
   to: Algorithmic Problem Solving 1

3. Course description  
   from: Introduction to problem solving methods and algorithm development; designing, coding, debugging, documenting programs in a high level programming language. 
   to: Primitive data types; single and multidimensional arrays; strings; control structures; basic I/O; subprograms and parameter passing; references; scope; introduction to recursion; designing, coding, debugging and documenting programs in a high level language.

4. Corequisite  
   MA-441

5. Hours and credits  
   3 class hours, 2 lab hours, 4 credits

6. Rationale  
   The changes in title and course description follow changes in the equivalent course at Queens College (CS 111). The course content places emphasis on the development of algorithms and the description is more specific.

7. Outcomes  
   Students will demonstrate understanding and skills in using a high level programming language, develop algorithms and implement them in a structured programming style.

8. Assessment  
   Written examinations and programming projects

9. Course outline  
   See below.

10. Methods of instruction  
    Lecture and recitation

11. Text  
    See below

12. Curriculum  
    The course can be used to satisfy liberal arts and science requirements and the concentration in science and mathematics for the A.S. (Mathematics and Science)

13. Transferability  
    CS 101 is accepted in place of the first programming course
(CS 111) at Queens College.

14. Faculty available Five full time and several adjunct faculty are available to teach this course.

15. Facilities The department offers the use of three computer laboratories for student projects.

16. Courses withdrawn None.

TITLE: CS-101 Algorithmic Problem Solving 1
4 Credits, 3 Class hours, 2 Lab hours

TEXT: Programming and Problem Solving with C++, Third Edition
by: Dale, Weems and Headington
Jones and Bartlett

OBJECTIVES: Introduction to problem solving methods and algorithm development: designing, debugging, and documenting programs in C++.

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>TOPIC</th>
<th>WEEK (approx)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2</td>
<td>Overview of Programming, Syntax and Semantics</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Types and Expressions</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>Software Design Process, Simple I/O</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>Selection (if)</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Looping (while), Exam 1</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>Functions</td>
<td>6-7</td>
</tr>
<tr>
<td>8</td>
<td>Scope, Lifetime, More on Functions</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>Additional Control Structures</td>
<td>9</td>
</tr>
<tr>
<td>10, 11*</td>
<td>User Defined Data Types, Records, Exam 2</td>
<td>10-11</td>
</tr>
<tr>
<td>12</td>
<td>One and Multi-dimensional Arrays</td>
<td>12-13</td>
</tr>
<tr>
<td>18*</td>
<td>Introduction to recursion, Review</td>
<td>14</td>
</tr>
</tbody>
</table>

* Selections from indicated chapters
Proposal for a Change in an Existing Course

1. Course number: CS-201
2. Course title: from: Computers and Programming to: Computer Organization and Assembly Language
3. Course description: from: Introduction to machine and assembly language; computer structure; digital representation of data and instruction execution; programming assignments in the computer laboratory to illustrate concepts. to: Computer organization; machine representation of data and instructions; assembly language; control statements; data manipulation; subprograms and macros; system calls and interrupts; program interaction through peripheral devices.
4. Prerequisite: from: CS-101 to: CS-101 with a grade of C or better.
5. Hours and credits: 3 class hours, 2 lab hours, 4 credits
6. Rationale: The course description is more specific. Experience indicates that students barely passing the prerequisite have difficulty in the next course. Queens College has also instituted a minimum grade of C on the prerequisite in lower level CS courses.
7. Outcomes: Students will demonstrate an understanding of computers at the level of 1st and 2nd generation languages and write programs that interact with the operating system and peripheral devices.
8. Assessment: Written examinations and programming projects.
9. Course outline: See below.
10. Methods of instruction: Lecture and recitation
11. Text: See below.
12. Curriculum: This course can satisfy liberal arts and
science requirements and the concentration in science and mathematics in the A.S. (Mathematics and Science) degree.

13. **Transferability**
   This course is accepted in place of CS 240 (old CS 141) at Queens College.

14. **Faculty available**
   Four full time and several adjunct faculty are able to teach this course.

15. **Facilities**
   Three computer laboratories are available.

16. **Courses withdrawn**
   None.

**TITLE:** CS-201 COMPUTERS AND PROGRAMMING  
3 Lecture Hours, 2 Lab Hours, 4 Credits  
**TEXT:** IBM PC Assembly Language Programming, 5th Edition  
by: Peter Abel  
Prentice Hall Publishing Company

**OBJECTIVES OF THE COURSE:** Assembly language for the IBM PC and an introduction to computer organization.

<table>
<thead>
<tr>
<th>TOPIC</th>
<th>SECTION</th>
<th>WEEK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number Systems</td>
<td>Chapter 1</td>
<td>1</td>
</tr>
<tr>
<td>PC Software Requirements</td>
<td>Chapters 2, 3</td>
<td>2</td>
</tr>
<tr>
<td>Assembly Language</td>
<td>Chapter 4</td>
<td>3</td>
</tr>
<tr>
<td>Assembly, Linking, and Executing</td>
<td>Chapters 5</td>
<td>4</td>
</tr>
<tr>
<td>Symbolic Instructions and Addressing</td>
<td>Chapter 6</td>
<td>5</td>
</tr>
<tr>
<td>Exam 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Program Logic and Control</td>
<td>Chapter 7</td>
<td>6</td>
</tr>
<tr>
<td>Screen and Keyboard Processing</td>
<td>Chapter 8</td>
<td>7</td>
</tr>
<tr>
<td>Strings</td>
<td>Chapters 11</td>
<td>8</td>
</tr>
<tr>
<td>Arithmetic</td>
<td>Chapters 12,13</td>
<td>9</td>
</tr>
<tr>
<td>Exam 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Macros</td>
<td>Chapter 21</td>
<td>10</td>
</tr>
<tr>
<td>Subprograms</td>
<td>Chapter 22</td>
<td>11</td>
</tr>
<tr>
<td>Advanced Screen/Keyboard</td>
<td>Chapter 9, 10</td>
<td>12</td>
</tr>
</tbody>
</table>
Proposal for a Change in an Existing Course

1. Course number CS-203 (no change)

2. Course Title
   from: Introduction to Computer Science
   to: Algorithmic Problem Solving 2

3. Course Description
   from: Continuation of CS-101; multidimensional arrays, graphics, introduction to data structures
   to: User defined data types, pointers and linked lists, ADT’s, stacks, queues, recursion, searching and simple sorting, elementary memory management. Object oriented problem solving.

4. Prerequisite
   from: Prerequisite CS-101
   to: Prerequisite CS-101 with a grade of C or better.

5. Hours and credits
   from: 3 class hours, 3 credits
   to: 3 class hours, 2 recitation hours, 4 credits

6. Rationale
   The change in hours and credits will agree with changes in hours and credits implemented at Queens College. Queens College will require a grade of C or better to advance in lower-level computer science courses. The revised course at Queens College will include more material in introductory data structures and this is reflected in the new course description.

7. Outcomes
   Students will demonstrate understanding and skills to develop algorithms and implement them with data structures using an object oriented programming language.

8. Assessment
   Written examinations and programming projects
9. Course outline See below.

10. Methods of instruction Lecture and recitation.

11. Text See below.

12. Curriculum The course can be used to satisfy liberal arts and science requirements and the concentration in science and mathematics for the A.S. (Mathematics and Science)

13. Transferability CS 203 will articulate with the second programming course (CS 211) at Queens College.

14. Faculty available Five full time faculty and several adjuncts with the department are able to teach this course.

15. Facilities The department offers the use of three computer laboratories for student projects.

16. Courses withdrawn None.

Course Outline

Text: Programming and problem Solving with C++, 3rd edition by: Dale, Weems and Headington Jones and Bartlett

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Topic</th>
<th>Weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Review of simple data types, user-defined data types</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>Intro to Recursion</td>
<td>½</td>
</tr>
<tr>
<td>13</td>
<td>Searching and Sorting</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>More on Strings and Recursion</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Intro to Classes, Private/Public Constructors</td>
<td>2</td>
</tr>
<tr>
<td>14</td>
<td>OOP including Inheritance and Polymorphism</td>
<td>2</td>
</tr>
<tr>
<td>*</td>
<td>Operator Overloading and Friends</td>
<td>1</td>
</tr>
<tr>
<td>*</td>
<td>Separate Compilation and Namespace</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>Pointers, dynamic data and memory management</td>
<td>1</td>
</tr>
<tr>
<td>16, *</td>
<td>Linked structures, stacks and queues</td>
<td>2</td>
</tr>
<tr>
<td>17, 18</td>
<td>Templates and Recursion</td>
<td>1½</td>
</tr>
<tr>
<td>2 exams, review</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Programming assignments and Final</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Teacher’s notes
Proposal for a New Course

1. Course number    CS-220
2. Course Title   Discrete Structures
3. Course Description  Recursion, recurrence and generating functions; relations; graphs and applications; asymptotics; trees; applications in computer science.
4. Prerequisite   MA-471 (Introduction to Discrete Mathematics)
5. Hours and credits  3 class hours, 3 credits
6. Rationale   This course will articulate with the second of the two semester sequence in discrete mathematics / discrete structures required of CS students at Queens College (CS 220). This course includes material covered in the proficiency examination given at Queens College for entry to upper level courses in CS.
7. Outcomes   Students will demonstrate proficiency in applying mathematical concepts in discrete mathematics to the applications of computer science.
8. Assessment   Written examinations and homework assignments
9. Course outline   See below.
10. Methods of instruction  Lecture and recitation
11. Text   See below.
12. Curriculum   This course can satisfy liberal arts and science requirements and the concentration in science and mathematics in the A.S. (Mathematics and Science) degree.
13. Transferability   This course will articulate with the Queens College course of the same name and number.
14. Faculty available   Five faculty teaching CS courses are able to teach this course.
15. Facilities   Classrooms and, if desired, computing laboratories are available.
16. Courses withdrawn  None.

Course Syllabus

by: Kenneth Rosen  
McGraw Hill

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Topic</th>
<th>Week</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Advanced Counting Techniques</td>
<td>1-3</td>
</tr>
<tr>
<td></td>
<td>recurrence relations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>solving recurrence relations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>divide and conquer relations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>inclusion-exclusion principle</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Relations</td>
<td>4-6</td>
</tr>
<tr>
<td></td>
<td>review of relations and n-ary relations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>representing relations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>closure of relations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>equivalence relations</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Graphs</td>
<td>7-10</td>
</tr>
<tr>
<td></td>
<td>introduction to graphs and terminology</td>
<td></td>
</tr>
<tr>
<td></td>
<td>representing graphs, isomorphisms</td>
<td></td>
</tr>
<tr>
<td></td>
<td>connectivity</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Eulerian and Hamiltonian paths</td>
<td></td>
</tr>
<tr>
<td></td>
<td>shortest path problems</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Trees</td>
<td>11-14</td>
</tr>
<tr>
<td></td>
<td>introduction to trees</td>
<td></td>
</tr>
<tr>
<td></td>
<td>applications</td>
<td></td>
</tr>
<tr>
<td></td>
<td>tree traversal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>spanning trees</td>
<td></td>
</tr>
</tbody>
</table>