Instructor:
Dr. Judy Hankins
Office: KOM 355
Office hours: 9:00-11:00 MWF and by appointment
Phone: 898-2392
E-mail address: csjudy@mtsu.edu

Prerequisites:
A grade of C or higher in 2170 is necessary to take this course. You must complete 3080 either prior to or during this semester.

Course Materials & Supplies:
Web page: http://www.cs.mtsu.edu/~jhankins. This page will contain links to course information, including this syllabus, calendar for the class, examples, PowerPoint presentations, notes, homework and open lab assignments. In addition, grades, lab assignments, and homework will be available on Desire2Learn(D2L). The Desire2Learn dropbox will be used for turning in assignments and for viewing your grades. To access D2L, use elearn.mtsu.edu and type in your pipeline user name and password.

Text: Carrano and Prichard, Data Abstraction and Problem Solving with C++ (5th ed), Addison-Wesley, ISBN 0-321-43332-7. Chapters 8, 9, 11 – 13 will be covered – not necessarily in sequential order. Other supplemental material deemed necessary by the instructor will be provided and covered in assignments and exams.

Software: Assignments for this class will be developed using Visual C++ in the Microsoft Visual Studio 2008 environment. This software is installed on all departmental lab machines and on the terminal server "shemp.cs.mtsu.edu." Because of the Computer Science Department’s agreement with Microsoft, you may check out Visual Studio installation disks from the Department and install it on your personal computer for academic use. Microsoft software is available for checkout to computer science students at http://msdn05.e-academy.com/elms/Storefront/Home.aspx?campus=mtsu_cs.

Catalog Description:
Topics include algorithm design, analysis of algorithms, advanced tree structures, indexing techniques, internal and external sorting, graphs, and file organizations. (We will emphasize object-oriented programming, including inheritance and polymorphism. The Standard Template Library will be introduced.)

Learning Outcomes:
Upon completion of this course, a student should be able to:
1. write object-oriented programs using advanced techniques such as inheritance, polymorphism, overloaded operators, abstract classes, virtual functions, dynamic binding and generic programming.
2. design and implement more complicated data structures such as priority queues, heaps, balanced search trees, hash tables, and graphs (including searching, inserting, deleting, and traversing elements in each structure).
3. use "Big-O" analysis to critique algorithms.
4. produce effective and efficient programs to solve complex practical problems by choosing the most appropriate data structures, and then using appropriate design, debugging, and testing techniques.
5. recognize the need for, and program in a consistent and well accepted coding style.
6. compare and analyze different internal sorting techniques.

Methods of instruction: Lecture and discussion during scheduled class times. You will also have six to eight programming assignments to be completed in open labs as well as non-programming homework.
**Academic honesty:** All work for this class (including exams and assignments) is to be done on an individual basis unless the assignment specifically says otherwise. There should be no discussion of laboratory assignments with other students. Only when special permission is given should you use code from another source. When permission is given, you should reference the source of the code in your documentation.

For more clarification see the Computer Science Academic Policy on Academic Integrity ([http://www.mtsu.edu/~csdept/Academics/academicIntegrity.htm](http://www.mtsu.edu/~csdept/Academics/academicIntegrity.htm)). The penalty for unauthorized collaboration will range from a grade of zero for an individual assignment to a failing grade for the course.

**Attendance:** You are expected to attend every class. Attendance will be taken every day. You are responsible for making up any material missed by being absent. You are also expected to be on time to class. Being late is considered to be rude and is definitely disruptive to the class.

**Message devices:** If you bring a mobile phone or pager to class or lab it must be either turned off or set to signal you silently. You may not use a mobile phone in class or lab. You may leave the room to use one only if there is an emergency situation. You may not engage in any use of electronic devices that distracts others in the class.

**Grading policy:** Your grade in this class will be based on

65% examination grades –

- There will be three examinations counting 100 points each and a final examination counting 100 points.
- Makeup exams will not be given. If you miss a regularly scheduled exam, the final exam will replace this exam. If an exam is not missed then the final can replace the lowest test score (assuming that the final is higher than the lowest score).
- These exams will cover lectures, assigned readings, homework assignments, etc.

35% open lab projects and homework assignments –

Open Lab Assignments

1. You will have six to eight programming assignments to be completed as open lab projects.
2. Refer to “3110 Open Lab Requirements” for a detailed description of requirements for open lab assignments for this class.
3. Each lab will be worth 100 points.
4. Labs are to be written in C++ using Visual Studio 2008.
5. Each project is due at midnight (actually 11:59) of the due date.
6. Open labs with syntax errors will automatically receive a grade of 0.
7. The maximum grade that you can earn on a project will decrease by 10% per class day. Holidays will be excluded.
8. No assignments will be accepted after graded assignments have been returned.
9. Late labs will not be accepted after three class days.
10. All code will be graded for programming style as well as function. Style guidelines are also in “3110 Open Lab Requirements”.
11. All lab assignments must be done on an individual basis – COLLABORATION on lab projects is NOT PERMITTED unless a team project is assigned.
12. The penalty for unauthorized collaboration will range from a grade of zero for an individual assignment to a failing grade for the course. There should be no discussion of laboratory assignments with other students.
13. Programs are graded based upon design, correctness, documentation, style, efficiency, elegance, and adherence to requirements. You must design, write, implement, and debug your own programs. You may discuss with others high level details of program design and implementation. However, the following situations are not allowed and will be treated as cheating:
   a. Showing to or acquiring from other students any materials related to assignments such as source code and documentation.
   b. Helping or seeking help from other students to debug programs. However, you may get help from a lab assistant or instructor.
14. All source code must be original. Only when special permission is given should you use code from another source. When permission is given, you should reference the source of the code in your documentation.

Homework Assignments

15. When non-programming homework assignments are given, they will normally be due the next class day by midnight.
16. Late homework assignments will not be accepted.

Calculating your average:

Your letter grade will be calculated by finding your test average, which is the sum of your four tests divided by 4. Next, calculate your lab/homework average. This is determined by dividing your total points by the total points possible and multiplying by 100. Then your average is calculated as .65 * test average + .35 * lab average.

Grading scale

Letter grades will be determined using a standard percentage point evaluation based on your course average as outlined below.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>90% - 100%</td>
</tr>
<tr>
<td>B+</td>
<td>87% - 89%</td>
</tr>
<tr>
<td>B</td>
<td>80% - 86%</td>
</tr>
<tr>
<td>C+</td>
<td>77% - 79%</td>
</tr>
<tr>
<td>C</td>
<td>70% - 76%</td>
</tr>
<tr>
<td>D+</td>
<td>67% - 69%</td>
</tr>
<tr>
<td>D</td>
<td>60% - 66%</td>
</tr>
<tr>
<td>F</td>
<td>0% - 59%</td>
</tr>
</tbody>
</table>

Important dates: Important dates and a tentative exam schedule will be posted on the Calendar page on the Website.

Reasonable accommodation for students with disabilities: If you have a disability that may require assistance or accommodation, or you have questions related to any accommodations for testing, note takers, readers, etc., please speak with me as soon as possible. Students may also contact the Office of Disabled Students Services (898-2783) with questions about such services.

Scholarship information:

To retain Tennessee Education Lottery Scholarship eligibility, you must earn a cumulative TELS GPA of 2.75 after 24 and 48 attempted hours and a cumulative TELS GPA of 3.0 thereafter. You may qualify with a 2.75 cumulative GPA after 72 attempted hours (and subsequent semesters), if you are enrolled full-time and maintain a semester GPA of at least 3.0. A grade of C, D, F, or I in this class may negatively impact TELS eligibility. Dropping a class after 14 days may also impact eligibility; if you withdraw from this class and it results in an enrollment status of less than full time, you may lose eligibility for your lottery scholarship. Lottery recipients are eligible to receive the scholarship for a maximum of five years from the date of initial enrollment, or until a bachelor degree is earned. For additional Lottery rules, please refer to your Lottery Statement of Understanding form, review lottery requirements on the web at http://scholarships.web.mtsu.edu/tescontelligibility.htm, or contact the Financial Aid Office at 898-2830.

Problems or Complaints: If you are having problems with the course or have a complaint you would like to voice, please bring this to my attention as soon as possible.

This syllabus represents a general plan for the course and deviations from this plan may be necessary during the duration of the course. I reserve the right to modify course policies, the course calendar, assignment point values, and due dates.