Objective: Algorithmic problem solving with a modern programming language. Language syntax; basic design methods; algorithms; abstraction.

Prerequisite: CS150 - Introduction to Computing II

zyBook: Subscribe at learn.zybooks.com, code: SIUECS240TornaritisSummer2019, section 001

Course website: The instructor’s web site at www.cs.siue.edu/~stornar. The site lists instructor schedule and assigned course TAs.

Moodle: The course management site, used for this course, accessed at classes.cs.siue.edu. Get access to all course material, e.g. syllabus, code examples, assessment etc.

IDE: Visual Studio Code is our choice of development environment, and a free copy of the IDE can be downloaded from code.visualstudio.com.

Assessment: The following assessment measures will be used.

- Reading (r): 10 weekly reading activities. A 90% completion on each will earn you full marks.
- Homework (h): 8 weekly homework labs (attempt 9, drop lowest attempted).
- Attendance (a): Required and timely. Must be on time to receive credit.
- Exam (e): 2 exams.
- Final (f): Comprehensive final. A 70% is required to pass the course.

| Out of class | [300] | r: [100] | h: [200] |
| In class | [700] | a: [100] | e1: [200] | e2: [200] | f: [200] |

A straight scale will be used for the course grade: A [>= 900], B [>= 800], C [>= 700], D [>=600], F [< 600]. In order to pass the course you must score at least a 70% on the final, otherwise the highest course grade you can earn will be a ‘D’.

Submission policy: No late submissions will be accepted for any reason. It is the student’s responsibility to submit all work on time and by any means, so please have a backup to prevent any failure to submit. Always double check your submission.

Make-up policy: In order to be considered for a make-up, a student must be proactive and not reactive. Students should inform the instructor of any scheduled absence or difficulty in meeting a deadline, in advance (proactive) not after the fact (reactive). On special unforeseen occasions such as a medical emergency, family hardship, natural disaster or anything that is out of a student’s control, the instructor should be notified as soon as possible and proper documentation should substantiate the absence. The instructor will have the final say in all such decisions regarding make-ups and assessed penalties if applicable.

Support services: Students needing accommodations because of medical diagnosis or major life impairment will need to register with Accessible Campus Community & Equitable Student Support (ACCESS) and complete an intake process before accommodations will be given. Students who believe they have a diagnosis but do not have documentation should contact ACCESS for assistance and/or appropriate referral. The ACCESS office is located in the Student Success Center, Room 1270. You can also reach the office via e-mail at myaccess@siue.edu or by calling 618.650.3726. For more information on policies, procedures, or necessary forms, please visit the ACCESS website at www.siue.edu/access.

Student attendance: Student attendance is essential and thus required. In case of an absence, students are responsible for all material covered and/or all announcements made during their absence.

Class decorum: Class time is limited and thus valuable; therefore a student must not be the cause of a distraction to their peers or the instructor by using a cellphone and/or a laptop for personal reasons. The first violation will result in a 25 point deduction and each subsequent violation will result in a 100 point deduction. In case of an emergency a student should step outside the classroom.

Academic misconduct: Academic honesty is a serious issue at SIUE, in the School of Engineering, in the Department of Computer Science, and with this instructor. Penalties for dishonest behavior will be severe. Even a single occurrence of plagiarism of English text, or program code, within a graded activity (e.g., homework, project, or exam) is grounds for academic discipline and a letter grade of ‘F’ in the course.
### Expectations:
Learning is an active process not a passive one, so as an Instructor I expect students to come to class prepared, having read all relevant material (book, notes, code) before as well as after class meetings. Don’t be afraid to ask questions or seek answers. Have an open mind and a willingness to learn and adopt alternative methodologies and practices. It takes two participants to transfer knowledge, the instructor and the learner. If one is absent the other suffers.

<table>
<thead>
<tr>
<th>wk</th>
<th>Concept (z: zyBook)</th>
<th>M</th>
<th>T</th>
<th>F</th>
</tr>
</thead>
</table>
| 1  | z1. User-Defined Functions (1-9)  
z2. Streams (1-7) | 06/03 | 06/05 | r1    |
|    |                     |       |       | 06/07 |
| 2  | z3. Objects and Classes (1-15)   | 06/10 | r2    | 06/12 | 06/14 | h1 |
| 3  | z4. Inheritance (1-5)  
z5. Templates (1-2)   | 06/17 | r3    | 06/19 | e1    |
| 4  | z6. Exceptions (1-3)  
z7. Pointers (1-12)   | 06/24 | r4    | 06/26 | 06/28 | h3 |
| 5  | z8. Lists, Stacks, and Queues (1-18) | 07/01 | r5    | 07/03 | 07/05 | h4 |
| 6  | z9. Searching and Sorting (1-16) | 07/08 | r6    | 07/10 | e2    |
| 7  | z10. Recursion (1-8)  
z11. Trees (1-7)   | 07/15 | r7    | 07/17 | 07/19 | h6 |
| 8  | z11. Trees (8-14)   | 07/22 | r8    | 07/24 | 07/26 | h7 |
| 9  | z12. Graphs (1-8)   | 07/29 | r9    | 07/31 | 08/02 | h8 |
| 10 | z12. Graphs (9-11)  | 08/05 | r10   | 08/07 | 08/09 | h9, f |