**CS 1302**

**Principles of Programming II**

**Fall 2025**

**Course Description from the VSU Catalog**

A study of advanced object-oriented programming. The course involves extensive programming that includes inheritance, polymorphism, dynamic binding, object composition, exception handling, file I/O, GUI, class diagrams, and unit testing.

**Course Learning Outcomes, Objectives, and/or Core IMPACTS Statement**

The following learning objectives are addressed in this course. Each of these objectives appears in one or more of the course activities.

1. Design and implement programs with classes and methods based on problem specification.
2. Design and document object-oriented models with class and object diagrams.
3. Develop programs that utilize object-oriented programming concepts, including encapsulation, inheritance, polymorphism and abstraction.
4. Develop programs that use text files.
5. Design and implement basic graphical user interfaces.
6. Develop programs that use exceptions.
7. Develop programs that use recursion.
8. Design and implement programs that utilize collection classes, including lists, sets, and maps.

**Required Course Materials and Tools**

This course requires the following text(s), course materials, and technology:

**Text/Readings/Other Materials**

# The text is written by the instructor and available for free at: <https://cs.valdosta.edu/~dgibson/courses/cs1302/text/index.shtml>.

**Technology Requirements**

In this class, students will regularly use the following applications:

* Eclipse
* Java
* Office 365 for access to VSU email and MS Word
* BlazeVIEW

To use these applications, students must have access to a laptop or computer, preferably with a webcam and mic. Note that the University maintains a lab in 2111 Nevins Hall which has the required software and is available for student use during the week from 8am until the building is locked (7pm?). Arrangements can occasionally be made with the instructor to access the lab during the weekend. No other labs on campus have the required software.

**Grading and Assessment**

**Summary of Grade Criteria**

|  |  |  |
| --- | --- | --- |
| **Assessment** | **Weight** | **Comments** |
| Labs | 10% | 16 labs, each worth 0.625% of final grade |
| Computing Assignments (CA) | 25% | There are 10. CA’s 1-9 are each worth 2.27% of final grade. CA 10 is worth 4.54% of final grade |
| Exams | 50% | 5 Exams, each worth 10% of final grade |
| Final Exam | 15% | Cumulative |

**Grading Scale**

The Course Average is computed from the weights above. Final grades are determined by up to 3 criteria as shown in the table below.

|  |  |  |  |
| --- | --- | --- | --- |
| **Grade** | **Course Avg** | **CA Avg** | **CA 10** |
| A | $$\geq 90$$ | $$\geq 70$$ | $$\geq 60$$ |
| B | $$80\leq Avg\leq 89$$ | $$\geq 70$$ | $$\geq 50$$ |
| C | $$70\leq Avg\leq 79$$ | $$\geq 60$$ |  |
| D | $$60\leq Avg\leq 69$$ |  |  |
| F | $$\leq 59$$ |  |  |

**Grading Procedure and Feedback**

* Labs are grade on completion.
* CA’s are graded by running test cases against your code for which you will receive detailed feedback via email.
* All Exams are closed book, notes, *etc.* For each exam, you will write code by hand. Each exam will be marked up and returned in class with a key. The instructor will go over the exam and then the exam and key will be returned to the instructor. Students may see their exam and key at a later date during the instructor’s office hours.

**Attendance and Engagement**

* You are expected to attend all class meetings. Attendance is taken before class begins, and ends at the time the class starts. Per VSU policy, if you miss more than 20% of scheduled class meetings, which corresponds to 10 or more class session, then you can receive an F grade for the course
* Leaving Early – you are required to stay for the entire class time. Each instance of leaving early counts as ½ an absence.
* Do not leave the classroom and return once lecture has started, excepting an emergency. Should such a situation develop, contact me immediately after class. Each instance of leaving early counts as ½ an absence.

**Academic Honesty and Integrity**

As members of the academic community, VSU students are responsible for knowing and abiding by the [Academic Honesty Policy](https://www.valdosta.edu/academics/academic-affairs/academic-honesty-at-vsu.php) as set forth in the [Student Code of Conduct](https://www.valdosta.edu/administration/student-affairs/student-conduct-office/student-handbook.php). All students are expected to do their own work and to uphold a high standard of academic ethics.

1. **Cheating on Computer Assignments** – All code submitted for Computer Assignments must be entirely your own work. You are encouraged to discuss with others, at a high-level, to understand the problem and how to solve it in words and/or pseudo-code. **You may not:** share code with someone else in any format (electronic, hard copy). Nor may you request or possess any of these from someone else.
2. **Cheating on Exams** – Utilize only authorized materials during exams. Additionally, sharing information with other students before or after an exam is also considered cheating.
3. **“Tutoring" Services** – If a "tutoring" service provides solutions to course assignments, this is considered cheating. Legitimate tutoring services will never directly give answers to course assignments.
4. **AI Assistants** – This is the policy we will use for this course for all Computer Assignments concerning the use of AI Assistants. This policy is identical to the one David Joiner developed at Georgia Tech[[1]](#footnote-1),[[2]](#footnote-2), except that I removed 2 sentences.

*We treat AI-based assistance, such as ChatGPT and Copilot, the same way we treat collaboration with other people: you are welcome to talk about your ideas and work with other people, both inside and outside the class, as well as with AI-based assistants.*

*However, all work you submit must be your own. You should never include in your assignment anything that was not written directly by you.* *If you are unsure where the line is between collaborating with AI and copying AI, we recommend the following heuristics:*

* *Heuristic 1: Never hit “Copy” within your conversation with an AI assistant. You can copy your own work into your own conversation, but do not copy anything from the conversation back into your assignment.*

*Instead, use your interaction with the AI assistant as a learning experience, then let your assignment reflect your improved understanding.*

* *Heuristic 2: Do not have your assignment and the AI agent open at the same time* (choose: Copilot, Sign out of GitHub Copilot)*. Similar to the above, use your conversation with the AI as a learning experience, then close the interaction down, open your assignment, and let your assignment reflect your revised knowledge.*

*This heuristic includes avoiding using AI directly integrated into your composition environment: just as you should not let a classmate write content or code directly into your submission, so also you should avoid using tools that directly add content to your submission.*

*Deviating from these heuristics does not automatically qualify as academic misconduct; however, following these heuristics essentially guarantees your collaboration will not cross the line into misconduct.*

Violation of the Academic Honesty Policy can result in failure of the assignment, failure of the course, and/or further consequences from VSU. Ignorance of this policy is not an excuse or a means to repeal a charge of academic dishonesty.

**Syllabus Addendum**

Changes made to this syllabus after it has been posted will be entered in this section along with the date the instructor notified students of the change.

1. <https://www.cc.gatech.edu/news/new-policies-navigate-role-ai-assistants-cs-courses> [↑](#footnote-ref-1)
2. <https://www.davidjoyner.net/on-my-ai-collaboration-policy/> [↑](#footnote-ref-2)