|  |  |
| --- | --- |
| color_150http://logo.gmu.edu/webguide/logos/color\_150.gif | **Applied Information Technology Department****Course Syllabus****IT 108 Programming Fundamentals** *revised 08.30.2010* |

**Catalog Description** Introduces programming fundamentals and presents software development process. Students learn to write programs in high-level, object-oriented language.

**Prerequisites** The prerequisite for this course is IT 103 (or an approved equivalent course). A grade of "C" or better **must** be achieved in the prerequisite course **before** a student is qualified to take this course. The prerequisite course must be completed prior to, not concurrently with, this course.

**Rationale** Programming is an essential skill for IT students and IT professionals. Understanding how a computer is instructed to accomplish tasks leads to an appreciation of the underlying concepts of the Information Technology discipline. Learning how to solve a problem using a structured or object-oriented programming language provides a strong foundation that will be used in Database, Security, Web Development and Networking courses.

**Educational**

**Objective** To introduce students to problem solving using a high-level, object-oriented programming language as a tool

**Course Outcomes** 1. Design solutions to problems using procedural and object‐oriented techniques.

2.Use procedural programming techniques effectively (including

 expressions, decisions, repetition structures, methods, parameters and

 variables scope) that use both primitive and reference types

 3. Apply techniques of basic object-oriented programming effectively

 (including creating class diagrams, implementing information hiding

 and encapsulation)

**Major Topics**

On successful completion of this course, students will be able to:

* Define and use data of both primitive and reference types effectively.
* Create and use simple and complex static data structures.
* Design solutions to problems using procedural and object-oriented techniques.
* Decide on an appropriate repetition and/or selection structures for given problems.
* Apply the techniques of structured (functional) decomposition to break a program into smaller pieces.
* Describe the mechanics of parameter passing and the issues associated with scoping.
* Discuss and identify the concepts of encapsulation, abstraction, and inheritance.
* Design, implement, test, and debug simple programs in an object-oriented programming language.
* Describe how the class mechanism supports encapsulation and information hiding.
* Understand the conditions on overloading methods in an object-oriented language.
* Describe the relationship between the static structure of the class and the dynamic structure of the instances of the class.
* Describe how constructors and destructors relate to the life of an object.
* Describe the relationship between an object and its corresponding class.
* Discuss the importance of algorithms in the problem-solving process.
* Identify the necessary properties of good algorithms.
* Create algorithms for solving simple problems.
* Use a programming language to implement, test, and debug algorithms for solving simple problems.
* Apply effective debugging strategies.

**References**

**Textbook**

There is one required textbook for this course:

|  |  |
| --- | --- |
| http://www.course.com/images/covers/0-619-21625-5.gif | Java for EveryoneCay HorstmannISBN-10: 0471791911ISBN-13: 978-0471791911Publisher: Wiley (February 2, 2010)Format: Paper; 515 pp |

**Faculty and Staff**

Instructor and Teaching Assistant information can be found via PatriotWeb and the course Blackboard site.

 Administrative support:

[Fairfax campus](http://www.gmu.edu/fairfax/)
 **Shirley Mancada**
 <http://eagle.gmu.edu/map/buildings/engineering.php> , Room 5400
 Phone: 703-993‑3565

[Prince William campus](http://www.gmu.edu/princewilliam/http/)
 **Cindy Woodfork**
 [Bull Run Hall](http://coyote.gmu.edu/map/pw.html), Suite 102
 Phone: 703-993‑8461

**Grading**

Grades will be awarded in accordance with the GMU Grading System for undergraduate students. See <http://www.gmu.edu/catalog/apolicies/> under [Grading System](http://www.gmu.edu/catalog/apolicies/#Anchor51) for more information.

The grading scale for this course is:
 99 – 100% A+ Passing
 92 – 98% A Passing
 90 – 91% A- Passing
 88 – 89% B+ Passing
 82 – 87% B Passing
 80 – 81% B- Passing
 78 – 79% C+ Passing
 72 – 77% C Passing
 70 – 71% C- *Passing*\*
 60 – 69% D *Passing*\*
  0 – 59% F **Failing**
\* Grades of "C-" and "D" are considered passing grades for undergraduate courses. However, a minimum grade of "C" is required in the BSIT program for any course that is a prerequisite for one or more other courses. This course is a prerequisite for several courses in BSIT Concentrations – see <http://www.gmu.edu/catalog/courses/it.html> for more information on those courses.

Raw scores may be adjusted by the Instructor to calculate final grades.

Final grades will be determined based on the following components:

|  |  |
| --- | --- |
| Activity | Distribution |
| Final Exam | 40% |
| Midterm Exam | 25% |
| In-Lab Assignments | 10% |
| Programming Assignments | 15% |
| Project | 10% |

These components are outlined in the following sections.

In-Lab Assignments

Some lab assignments are to be completed during the assigned lab time for the course whereas others will be due several days later. They will either be treated as a quiz or regular assignment, to be determined by the instructor. For regular assignments, students may use any resources available to complete the assignment, including their text, notes, Internet resources and discussion with classmates. Acceptance of late assignments will be at the discretion of the GTA. Students may submit incomplete assignments for partial credit.

There may or may not be lab quizzes during the course. Lab Quizzes are to be completed during the assigned lab time for the course. Students may only use their text and written notes to complete the quizzes. Students ***may not*** use Internet resources and there may not be discussion with classmates. Students ***must*** be present in the lab to submit these quizzes. Students will only be allowed to make up a lab quiz under extreme conditions and approval to do so will be at the discretion of the instructor. No late quizzes will be accepted.

Programming Assignments and Project

Each student individually will prepare and submit a solution to each programming assignment and phases of the project in accordance with requirements to be discussed in class and published on [the Blackboard Learning System](http://gmu.blackboard.com/). Acceptance of late assignments will be at the discretion of the instructor. Students may submit incomplete assignments for partial credit. These assignments must be completed by the student. Assistance may be requested from the instructor, GTA or UTA. Other assistance may be considered an Honor Code violation.

Mid‑term exam

The mid‑term exam is a department-wide exam and will be conducted during the scheduled class session in Week 8 (October 18-21) and will be based on topics addressed from Chapters 1‑5. The mid‑term exam will be “closed book” – no reference materials other than those provided with the exam paper will be permitted. It is possible that a sheet (front and back) of individually-prepared sheet of notes will be allowed for the exam.

Final exam

The final exam is a department-wide exam and will be held during the scheduled final exam session (see [http://registrar.gmu.edu/](http://registrar.gmu.edu/calendars/Fall08exams.pdf) ) and will be based on topics addressed throughout the entire course. The final exam will be “closed book” – no reference materials other than those provided with the exam paper will be permitted. It is possible that a sheet (front and back) of individually-prepared sheet of notes will be allowed for the exam. Final exams will be retained by the [Department of Applied Information Technology](http://ait.gmu.edu/) and will not be returned to students.

Final grades will be posted to [PatriotWeb](https://patriotweb.gmu.edu/), which is the only vehicle for students to obtain those grades. A student with a "hold" on his/her PatriotWeb account will be unable to access final grades until the hold has been removed by the Registrar.

**Schedule**

|  |  |  |
| --- | --- | --- |
| Lecture | Class Content | Reading |
|
| 1 | Problem Solving |   |
| 2 | Introduction to Computers, Programs, and Java | Chapter-1 |
| 3 | Problem SolvingTop-Down Design, Sequence, Selection, Repetition, Algorithmic Design |  |
| 4 | Primitive Data Types and Assignment, GUI and Console Input and Output | Chapter-2 |
| 5 | Typecasting, Compound Assignment Operators, Increment and Decrement Operators and Efficiency | Chapter 2 |
|
| 6 | Selection Statements: ***if, if/else, switch*** | Chapter-3 |
|
| 7 | Selection Statements: tertiary operator, nested selection statements. Translation of selection statements from algorithmic design | Chapter 3 |
| 8 | Repetition – presentation of three loops, choosing the appropriate loop structure, translation of repetition statements from algorithmic design | Chapter 4 |
| 9 | Repetition – use of sentinel value, user input validation, use of ***break/continue*** | Chapter 4 |
| 10 | Methods – using top down design in algorithms, method definition and invocation, use of parameters | Chapter 5 |
| 11 | Methods – local variables and parameters, variable scope, method abstraction and stepwise refinement, method reuse | Chapter 5 |
| 12 | Methods – method overloading, code efficiency, Math class, generating Random numbers | Chapter 5 |
| 13 | Midterm Exam | Chapters 1 - 5 |
| 14 | Arrays – storage and use of basic arrays, efficient use of memory |   |
| 15 | Arrays – using arrays as parameters and method return values | Chapter 6 |
| 16 | Arrays – Searching and Sorting, the Arrays Class | Chapter 6 |
| 17 | Arrays – used in problem-solving, using two-dimensional arrays | Chapter 6 |
| 18  | Objects and Classes – using OO design, encapsulation, objects as reference variables, efficiency and reuse, types of methods used in classes (accessor, mutator, constructor and special purpose) | Chapter 6 |
| 19 | Objects and Classes – using objects as parameters and method return values, construction and destruction of objects. Implementation of objects from algorithmic design | Chapter 7 |
| 20 | Objects and Classes – static variables and methods, efficiency | Chapter 7 |
| 21  | Objects and Classes – arrays as static and instance variables, arrays of objects | Chapter 7 |
| 22 | Objects and Classes – implementation of OO design and stepwise refinement to create a solution to a problem of a significant size | Chapter 7 |
| 23 | Strings – using the String and Character classes |   |
| 24 | Strings – a full lifecycle application of all concepts covered in the course thus far |   |
| 25 | Files – basic files, using text files | Chapter 8 |
| 26 | Files – implementing error detection in applications that use files | Chapter 8 |
| 27 | Files |   Chapter 8 |
| 28  | Final Exam Review |   |
|  |  |  |

***This schedule is subject to revision before and throughout the course.
Registered students should see*** [[the Blackboard Learning System](http://gmu.blackboard.com/)](http://webct.gmu.edu/) ***for the latest class schedule.***

Important Dates

**Dates for adding, dropping the course, etc. are available via: registrar.gmu.edu**

Religious Holidays

A list of religious holidays is available on the [University Life Calendar page](http://www.gmu.edu/departments/unilife/pages/calendar.html). Any student whose religious observance conflicts with a scheduled course activity must contact the Instructor **at least 2 weeks in advance** of the conflict date in order to make alternative arrangements.

**Attendance Policy**

Students are expected to attend each class, to complete any required preparatory work (including assigned reading – see **Schedule** above) and to participate actively in lectures, discussions and exercises. As members of the academic community, all students are expected to contribute regardless of their proficiency with the subject matter.

Any student who expects to miss more than one class session is strongly advised to drop the course and take it in a later semester when he/she can attend every class. On an occasional basis, students may attend the corresponding lecture in the other section if they must be absent during their section.

Departmental policy requires students to take exams at the scheduled time and place, unless there are truly compelling circumstances supported by appropriate documentation. Except in such circumstances, failure to arrive to the exam site on time for a scheduled exam will result in a score of zero (0) for that exam, in accordance with [Mason policy on final exams](http://www.gmu.edu/catalog/apolicies/index.html#final_exams). Students should not make travel plans or other discretionary arrangements that conflict with scheduled classes and/or exams. If the University is closed due to weather or other unforeseen conditions, final exams may be rescheduled – students are strongly advised not to make plans that would prevent them from attending exams that may be rescheduled during the entire [exam period](http://registrar.gmu.edu/calendars/Fall08exams.pdf).

**Classroom conduct**

Students are expected to conduct themselves in a manner that is conducive to learning, as directed by the Instructor. Any student who negatively impacts the opportunity for other students to learn will be warned – if disruptive behavior continues, the student will be asked to leave the classroom.

Electronic devices are potential distractions in the classroom environment. Cell phones, pagers and other handheld devices must be turned off or set to "silent" mode and not used while class is in session. Laptop computers and similar devices may be used only if such use is directly related to the classroom activity in progress – for some activities the Instructor may require that such devices not be used in order to maximize student engagement.

**Communications**

Registered students will be given access to a section of [[the Blackboard Learning System](http://gmu.blackboard.com/)](http://webct.gmu.edu/) for this course. Blackboard will used as the primary mechanism (outside of lectures) to disseminate course information, including announcements, lecture slides, homework and other assignments, and scores for homework and exams.

Communication with the Instructor on issues relating to the individual student should be conducted using Blackboard Mail, GMU email, via telephone, or in person - **not** in the public forums on Blackboard. Blackboard Mail is the preferred method – for urgent messages, you should also attempt to contact the Instructor via telephone. Federal privacy law and GMU policy require that any communication with a student related in any way to a student's status be conducted using secure GMU systems – if you use email to communicate with the Instructor about any grade issues you **MUST** send messages from your GMU email account.

**Lecture slides are complements to the lecture process, not substitutes for it.**

All course materials (lecture slides, assignment specifications, *etc*) are published on Blackboard in Adobe® Portable Document Format (PDF) or in a format for which a free ***reader*** is available (such as Microsoft PowerPoint). This allows users of most computing platforms to view and print these files. Microsoft® Word (or a compatible word processing application) is required for preparing assignments – it is available on computers in the Mason open labs.

**Privacy**

Instructors respect and protect the privacy of information related to individual students.

As described above, issues relating to an individual student will discussed via email, telephone or in person. Instructors will not discuss issues relating to an individual student with other students (or anyone without a need to know) without prior permission of the student.

Assessable work other than exams will be returned to individual students directly by the Instructor (or by a faculty or staff member or a Teaching Assistant designated by the Instructor, or via another secure method). Under no circumstances will a student's graded work be returned to another student.

Faculty and staff will take care to protect the privacy of each student's scores and grades.

**Disability Accommodations**

[The Office of Disability Services (ODS)](http://ods.gmu.edu/) works with disabled students to arrange for appropriate accommodations to ensure equal access to university services. Any student with a disability of any kind is strongly encouraged to register with ODS as soon as possible and take advantage of the services offered.

Accommodations for disabled students **must** be made in advance – ODS cannot assist students retroactively, and at least one week's notice is required for special accommodations related to exams. Any student who needs accommodation should contact the Instructor during the first week of the semester so the sufficient time is allowed to make arrangements.

**Honor Code**

All members of the Mason community are expected to uphold the principles of scholarly ethics. Similarly, graduating students are bound by the ethical requirements of the professional communities they join. The ethics requirements for some of the communities relevant to Applied IT graduates are available via the following links:

* [ACM Code of Ethics and Professional Conduct](http://www.acm.org/about/code-of-ethics)
* [IEEE Code of Ethics](http://www.ieee.org/portal/pages/iportals/aboutus/ethics/code.html)
* [EC‑Council Code of Ethics](http://www.eccouncil.org/codeofethics.htm)

On admission to Mason, students agree to comply with the requirements of the [GMU Honor System and Code](http://www.gmu.edu/catalog/apolicies/#Anchor12)[[1]](#footnote-1). The Honor Code will be strictly enforced in this course. Honor Code cases are heard by a panel consisting of students – students who meet the requirements are encouraged to nominate themselves to serve on the Honor Committee.

Any use of the words or ideas of another person(s), without explicit attribution that clearly identifies the material used and its source in an appropriate manner, is **plagiarism** and will not be tolerated. Dean Griffiths has mandated a "zero tolerance" policy for plagiarism within [The Volgenau School](http://ite.gmu.edu/). The Instructor reserves the right to use manual and/or automated means (including such services as Safe Assign and MOSS – Measure of Software Similarity) to detect plagiarism in any work submitted by students for this course, and to direct Teaching Assistants and/or other faculty and/or staff members to do likewise in support of this course.

For this course, the following requirements are specified:

* All assessable work is to be prepared by the individual student,
unless the Instructor explicitly directs otherwise.
* All work must be newly created by the individual student for this course for this semester.
Any usage of work developed for another course, or for this course in a prior semester, is strictly prohibited without prior approval from the instructor.

Students may seek assistance with assigned work, (and are encouraged to do so) **provided**:

* The directions for the assigned work do not prohibit such assistance.
* Such assistance is acknowledged in the submitted work, clearly identifying the person(s) giving assistance and the nature of the assistance given.
* Any work to be submitted is prepared entirely and exclusively by the student submitting it. Students are expressly prohibited from sharing any assessable work for this course in any manner with other students (except students assigned as Teaching Assistants to this course and the student's section), unless all students involved have had their work graded and returned by the Instructor, or the Instructor has explicitly approved such sharing.*footer for HTML version*
1. Available at [www.gmu.edu/catalog/apolicies](http://www.gmu.edu/catalog/apolicies/#Anchor12) and related GMU Web pages. [↑](#footnote-ref-1)