​​​​CpS 109 - Introduction to Computer Programming

Fall 2024

|  |  |  |
| --- | --- | --- |
| Dr. Hughes | **Instructor:** | Dr. Alan Hughes |
| **Office:** | Alumni 76;  Alternatively, MB203 (CS lab) |
| **Office Hours:** | MWF 2pm (by appt);  Tue. electronic (by appt)  Th – 10am by appt |
| **Email:** | [ahughes@bju.edu](mailto:ahughes@bju.edu) (recommended method of contact) |
| **Telephone:** | Office: 864-242-5100 x 2274  Cell: 864-906-1024 |

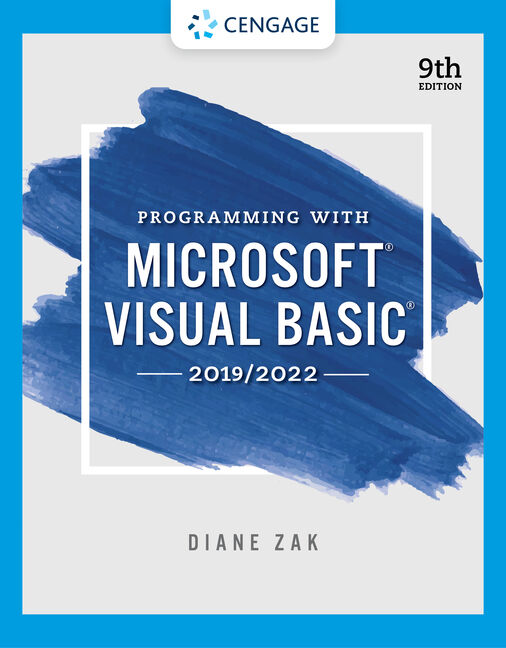
Course Description:

An introduction to programming using a high-level language, with an emphasis on problem-solving using structured techniques and objects, and the development of problem-solving skills through programming exercises.

Course Reading(s):

Programming With Microsoft Visual Basic 2019/2022 - Diane Zak (9th edition),

ISBN: 978-0-357-674000



Context:

The faculty of the Computer Science department has aligned the computer science program with the goals of the Mathematical Sciences Division (MS), BJU Bible and liberal arts core objectives, and BJU institutional goals. The goal of the Computer Science department is to align all courses in the Computer Science and Information Technology majors to support one or more of the following departmental goals. An asterisk indicates a goal fulfilled by this course.

1. Design and implement solutions to practical problems \*
2. Use appropriate technology as a tool to solve problems in various domains
3. Create efficient solutions at the appropriate abstraction level
4. Demonstrate an ability to work effectively in teams
5. Demonstrate an ability to communicate technological information effectively both in written and oral forms
6. Demonstrate an ability to acquire new knowledge in the computing discipline \*
7. Demonstrate an understanding of social, professional and ethical considerations related to computing
8. Demonstrate understanding of fundamental concepts in the (student's) discipline \*
9. Prepare students for graduate school or to secure employment in a related area

CpS 109 Course-specific Goals:

1. Design and implement solutions to practical problems.
2. Demonstrate an ability to acquire new knowledge in the computing discipline.
3. Demonstrate understanding of fundamental concepts in the discipline.
4. Provide the student a platform for continued learning and development of his/her God-given abilities.
5. Instill in the student a desire to use his/her abilities in service to Christ.

Learning Objectives:

At the end of the course, students should be able to:

|  |  |
| --- | --- |
| **Learning Objective** | **Assessment Tools** |
| Analyze and explain the behavior of simple programs involving the fundamental programming constructs covered by this course. | Quizzes, Tests, Final Exam |
| Design, implement, test, and debug programs that use each of the following fundamental programming constructs: | Programming Projects, Lab Tests, Quizzes |
| Choose appropriate conditional and iterative constructs for a given programming task. | Programming Projects, Tests, Quizzes |
| Describe the mechanics of parameter passing. | Quizzes, Written Tests |
| Use VB.NET to implement, test, and debug algorithms for solving simple problems. | Programming Projects, Lab Tests |
| Correctly identify and use of primitive data types in computer programs. | Quizzes, Programming Projects, Tests |

Course Policies:

Qualifications

CpS 109 is for students with a Math ACT greater than 21 or who have placed into Ma 103.

Absences, lateness, and makeup opportunities

1. The overarching guide for class attendance is the [BJU Class Attendance](http://home.bju.edu/life/policies/class-attendance-policy.php) Policy.
2. For planned absences, please email me one week in advance.
3. Written assignments should be submitted before your planned absence.
4. Scheduled tests and quizzes should be taken before your planned absence; please contact me to make arrangements for doing so.
5. For absences due to incapacitating illness or emergency, you should contact me as soon as you are able to return to class in order to make arrangements for making up any graded work without penalty.
6. In other circumstances, tests and quizzes may be made up within one week of your return, with a 10 percent grade penalty for that test or quiz.
7. Leaving class early without prior arrangement will constitute an absence.
8. Each absence costs 2 class participation points; each late costs 1 class participation point.

Late Work:

1. Assignments must be submitted using the electronic submission system before 11:59pm on the day due.
2. The use of the submission system will be explained during the first week of class.
3. Only work missed for legitimate reasons may be made up without penalty.  Legitimate reasons include illness, a death in the family, or a BJU sanctioned tour.
4. You must make up late work according to the number of days missed - that is, missing one day of class gives you one extra day to turn in your work.
5. Any other late work will receive a 20% grade penalty.
6. All late work must be made up within one week in order to receive a non-zero grade.

Grade appeals:

1. Grading appeals must be made by email only no later than one week after the grade was assigned.
2. Appeals will be automatically denied if the student attempts to make the appeal verbally.
3. Email grading appeals should be made respectfully and logically (My grade should be increased because).

Academic Integrity

1. The overarching guide for academic integrity is the [BJU Academic Integrity Policy](http://home.bju.edu/academics/integrity.pdf).
2. Cheating on assignments and tests is a form of deception and stealing, and as such, is prohibited by Scripture and will incur academic penalties.
3. Since the goal of the assignments in this course is to learn to develop the skills covered NOT complete the tasks assigned, and since the use of AI to complete or jumpstart tasks defeats the goal of the assignments, you may not use generative AI tools (i.e. Chat GPT, Bing Chat, Google Bard, etc.) in this course for any assignment without the professors express permission.  Should an AI tool be used with permission, its use must be documented.
4. All work is to be done individually unless Dr. Hughes gives permission for group work.
5. In general students are encouraged to assist one another in debugging programs, *but must exercise care when seeking assistance while writing code*.
6. **Each student needs to become familiar with Visual Basic, and be able to write programs on his or her own. Therefore, please do not copy code from another person, as this constitutes cheating.**

Class Participation:

1. Compliance with the [BJU Student Handbook](http://www.bju.edu/life-faith/student-handbook.pdf) policies is expected during class.
2. Class participation grades are based upon actively participating in lecture class discussions, working diligently on course assignments in lab classes and being respectful to the rest of the class and the instructor.
3. Playing games, electronic messages, working on other subjects, etc. is a demonstration of disrespect for both the instructor and other students, and is not allowed during lecture or lab classes. This includes the use of cell phones or other devices not employed for class work.

Instructor Help Outside of Class

**You are encouraged to use email (recommended) or the telephone to ask Dr. Hughes for assistance.**

Copyright Policy:

Copyright 2009-2024 Alan Hughes as to this syllabus and all lectures. Students are prohibited from selling (or being paid for taking) notes during the course to, or by any person, or commercial firm without the express written permission of the professor teaching the course.

Schedule (***subject to modification as needed, during the semester***)​

|  |  |  |  |
| --- | --- | --- | --- |
| **Please note test dates and project due dates.** ​ ​ ​ | | | |
| **Date** | **Day** | **Class** | **Assignment** **Due** |
|  |  | **Week 1** |  |
| Aug 28 | W | Introductions; syllabus overview; First look at Visual Basic Community (2019 or 2022)  **Practice Project**  [Download Visual Studio Community](https://visualstudio.microsoft.com/vs/community/" \o "Visual Studio Download" \t "_blank)  Be sure to register if it asks you to do so.  Once it is downloaded and installed, click Help to see if it wants you to register. | Read Chapter 1; Review Syllabus |
| Aug 30 | F | Lab – Practice Project | **Practice Project Due(Hello World)** |
|  |  | **Week 2** |  |
| Sep 2 | M | Labor Day |  |
| Sep 4 | W | Chapter 1 – Create an App; Windows in VS; Properties; Save a Solution  Chapter 1 – Adding controls; adding code; Assignment statements; comments; pseudocode | Read Chapter 1 |
| Sep 6 | F | Lab |  |
|  |  | **Week 3** |  |
| Sep 9 | M | Chapter 2 – Designing the User Interface (labels, textboxes) |  |
| Sep 11 | W | Chapter 2 – Designing the User Interface (labels, textboxes) | Quiz 1 Read Chapter 2 |
| Sep 13 | F | Lab – hotkeys, tab order | Project 1 (Weather) due |
|  |  | **Week 4** |  |
| Sep 16 | M | Chapter 3 – variables and memory; data types and names; DIM statements; TryParse | Quiz 2 Read Chapter 3 |
| Sep 18 | W | Chapter 3 – arithmetic; ToString; named constants |  |
| Sep 20 | F | Lab |  |
|  |  | **Week 5** |  |
| Sep 23 | M | Chapter 4 – Selection structures, if-then-else | Read Chapter 4  Quiz 3 |
| Sep 25 | W | No classes |  |
| Sep 27 | F | Lab – Select-Case statements; checkboxes and radio buttons | Project 2 (Car Rental) |
|  |  | **Week 6** |  |
| Sep 30 | M | **Test 1 (Ch 1-4)** | Read Chapter 5  Quiz 4 |
| Oct 2 | W | Chapter 5 – Loops, infinite loops, Do-while, for-next; list boxes and loops |  |
| Oct 4 | F | Chapter 5 |  |
|  |  | **Week 7** |  |
| Oct 7 | M | Chapter 6 – sub-procedures and function procedures | Quiz 5 |
| ​Oct 9 | ​W | Chapter 6 – passing parameters; comboboxes |  |
| ​Oct 11 | ​F | **Lab Test 1** |  |
|  |  | **Week 8** |  |
| ​Oct 14 | ​​M | Chapter 7 – Strings;; padding; trimming, | Quiz 6  Project 3 (Summer Reading) |
| Oct 16 | W | Chapter 7 – substrings; Random numbers, |  |
| Oct 18 | F | Lab |  |
|  |  | **Week 9** |  |
| Oct 21-22 | M | Fall Break |  |
| Oct 23 | W | Chapter 8 –Arrays; storing data | Quiz 7 |
| Oct 25 | F | Chapter 8 – Arrays, traversing an array; two-dimensional arrays |  |
|  |  | **Week 10** |  |
| Oct 28 | M | Chapter 9 – Menus, sequential files | Quiz 8 |
| Oct 30 | W | Chapter 9 – Sequential files |  |
| Nov 1 | F | Lab | Project 4 (Taste Test) |
|  |  | **Week 11** |  |
| Oct 4 | M | Chapter 10 – Classes and objects | Quiz 9 |
| Nov 6 | W | Chapter 10 – constructors; overload methods |  |
| Nov 8 | F | Lab |  |
|  |  | **Week 12** |  |
| Nov 11 | M | Chapter 11 – SQL Server Databases | Quiz 10 |
| ​Nov 13 | W | Chapter 11 – SQL Server Databases |  |
| Nov 15 | F | Lab | Project 5 (Burgers and Booths) |
|  |  | **Week 13** |  |
| Nov 18 | M | Chapter 11 |  |
| Nov 20 | W | Chapter 11 | Quiz 11 |
| Nov 22 | F | **Test 2 (Ch 5-9)** |  |
|  |  | **Week 14** |  |
| Nov 25-29 | M-F | **Thanksgiving Break** |  |
|  |  | **Week 15** |  |
| Dec 2 | M | Data Structures |  |
| Dec 4 | W | Data Structures |  |
| Dec 6 | F | Lab – Personal project | Project 6 (Yacht Rental, using data structures) |
|  |  | **Week 16** |  |
| Dec 9 | M | Lab – Personal project |  |
| Dec 11 | W | Lab – Personal project |  |
| Dec 13 | F | **Lab Test 2** | Personal Project (Saturday) |
| Dec 16 | M | **Final Exam 3:30-4:40pm** |  |

Grading

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Item** | **Pts.** | **Total** |
| 12 | Quizzes | 30 | 360 |
| 6 | Programming projects (labs) | 40 | 240 |
| 1 | Personal Project | 100 | 100 |
| 1 | Lab Test 1 | 50 | 50 |
| 1 | Lab Test 2 | 150 | 150 |
| 2 | Tests | 100 | 200 |
| 1 | Final Exam | 100 | 100 |
| ​ | **TOTAL** | ​ | **1200** |

|  |  |
| --- | --- |
| **Grade** | **Score** |
| A | 90-100 |
| B | 80-89 |
| C | 70-79 |
| D | 60-69 |
| F | < 60 |

