

Course Info

Summer 2019 • CRN 30213 • Mon/Wed 6:00-8:45pm
15 sessions (Discussion/Lab mix) • TTC Room 2150

Prerequisites: PC literacy is required. Students enrolled in this class should have successfully completed CIS 100 or BUS 103. Others may request the instructor's permission.

Course Description: This course introduces students to the essential programming knowledge for advancing to a programming language course. Learning will include the fundamentals of structured program design, including standard business programming algorithms and documentation practices. It familiarizes the student with the flow of program logic, flowcharting, use of the three necessary control structures of programming, an introductory structured programming language and design methodologies. Concepts such as multiple level accumulation and array processing are included. Minimum benchmarks will be enforced.

Learning Objectives

- Flowchart the logical steps for programming problems incorporating calculations, nested If's/comparisons, record selection, data table / array creation and searches, modifications and elementary read, write and move commands
- Create 3 and 4 level Hierarchy/Structure Charts of modules (subroutines) in a program
- Identify modules and create program flowcharts using structured programming design following the 3 control structures of sequence, selection and iteration
- Follow the Program Development Cycle of Analyze, Design, Code, Test and Documentation
- Appropriately document your program
- Code and execute all programs using a beginning procedural programming language, those without event or object driven processes
 - Visual Studio Visual Basic console programming in this course
- Understand the concepts of Sorting, Sequential and Random File processing

General Education Outcomes

- Effective written and oral communication
- Ability to think critically and to solve problems
- Information, numeric, and technology literacy
- A highly developed sense of ethics
- Strong personal management skills

Textbook: [Starting out with Programming Logic and Design, 4th Edition](#) by Tony Gaddis

Instructor/KVCC Info

Office: 7338

Campus email: jburns@kvcc.edu – (best contact method)

Cell phone (will share if we find that necessary)

Campus phone: 269.488.4113

Voice mail: 269.488.4701 ...extension 6142

Office hours: M/W – 5:30-6:00ish classroom (if available) OR by appointment

KVCC Public Safety: 269.488.4575 --- located in room 5120 (TTC)

Class Policies

- Cheating/Plagiarism- [Student Handbook \(KVCC PDF\) Page 17, Section II & III](#)
- Graded Items
 - Must be turned-in, as scheduled.
 - Prior arrangements may be made - at the discretion of the instructor.
- Hand-in items due as defined in schedule, or via class session discussion
 - Occasionally, due dates “may be” modified due to snow day or other closures; however, in general – we are an electronic society

Grading

Attendance	15 sessions	
Project 1-3	50pts each	150pts - See "To Be Graded"
Quiz 1-3	25pts each	75pts - Class & Reading topics
Article Review	25pts	25pts total (current events)
Final Exam	100 pts	Final: 100pts - Comprehensive
	Total	350

	Grade	Points	%
Outstanding	4.0	315-350	90-100
Excellent	3.5	283-314	85-89
Good	3.0	253-282	80-84
	2.5	227-252	75-79
	2.0	204-226	70-74
	1.5	182-203	65-69
	1.0	163-181	60-64
	0.0	0-162	0-59

Definitions

4.0... = someone that understands a majority of the subject matter for the course and can apply that knowledge to create a variety of well-written programs. They understand where to go to find additional information and how to weave-in that new knowledge. To an employer, this student is a self-starter and requires little, to no handholding.

3.5... = better than a 3.0 and less than a 4.0

3.0... = someone that understands the subject matter at a level below the 4.0, can apply most of the knowledge to create programs similar to those used within the course, and knows where to find additional information, but struggles at how it fits into their current knowledge. To an employer, this student will require some handholding over the first few months to a year as the student gains additional experience.

2.5... = better than a 2.0 and less than 3.0

2.0... = someone who struggles with the subject matter, fails to apply this knowledge to new programs, yet can work through course example programs if given enough time. This student relies completely on the instructor and course book and has not learned how to use external knowledge sources, nor how to apply that knowledge. The student is one who has had outside influences affecting focus on coursework, procrastinated, or just did not try. To an employer, this student would not be valuable in the subject-matter arena.

0.0, 1.0, or 1.5 ... This student should repeat the course.

Note: If maintaining a higher GPA is important to you - and your ability to do well in the course has been compromised, I highly recommend that you withdraw from the course before the cut-off period. Take the course again, when you are able to focus.

External References

Examples of Programming Logic (YouTube)

- "Concepts...": <https://www.youtube.com/watch?v=DF2XAc07eI0>
- "MIT Algorithmic Thinking": <https://www.youtube.com/watch?v=HtSuA80QTyo>
- "TEDx Algorithms": https://www.youtube.com/watch?v=H_aLU-N0dHM
- "Problem... FlowCharts": <https://www.youtube.com/watch?v=hN9xemJYwos>

Programming/Development Guidelines

- Microsoft "Naming Guidelines": <https://docs.microsoft.com/en-us/dotnet/standard/design-guidelines/naming-guidelines>
- Microsoft "Working with Base Types": <https://docs.microsoft.com/en-us/dotnet/standard/base-types/>
- Microsoft Search (UX guidelines): <https://social.msdn.microsoft.com/Search/en-US?query=windows%20ux%20guidelines&ac=5>
- Pete Brown: <http://10rem.net/articles/net-naming-conventions-and-programming-standards---best-practices>
- StackOverflow: <http://stackoverflow.com/questions/181597/what-are-the-naming-guidelines-for-asp-net-controls>
- Joel Spolsky: <http://www.joelonsoftware.com/articles/Wrong.html>

See References folder (e.g. Course folder tree), for more

Highly Recommended

- **Code Complete** - [author's site](#), Amazon: [Author's books](#)
- **Don't Make Me Think** - [author's site](#), Amazon: [Author's books](#)
- **Agile Programming** - https://en.wikipedia.org/wiki/Agile_software_development

Class Schedule (subject to change)

May

Date	DOW	To Do	Details/Reference
6	M	Intros, Learning Community Chapter 1: Intro to Computers & Programming Intro to Visual Studio/Visual Basic Flowcharting	
8	W	Chapter 2: Input, Processing, and Output Data Types Chapter 3: Sub Procedures (Modules)	
13	M	Chapter 5/10: Repeating Structures w/Files Ch 5 – Repetition Structures 171 – 191 DO While/ DO Until Loops Ch 10 – Files: 377-423, Sections 10.1, 10.2, 10.4 along with 423, Skip 398-399 Sec 10.3; 417; Section 10.5 except for 423	Quiz #1 Project #1 handout
15	W	Chapter 5 – Accumulation Chapter 6 - Functions	
20	M	Article Review info - Discuss Chapter 4: Decisions Chapter 7: Validations	
22	W	Chapter 10: Control Breaks	Project 1 Due Quiz #2
27	M	Memorial Day - no class	
29	W	Continued... Midterm Exam	

June

Date	DOW	To Do	Details/Reference
3	M	Chapter 5: For Loops Chapter 8: Arrays	Article Review -DUE Project #2 handout
5	W	Continued...	
10	M	Continued...	Quiz #3
12	W	Chapter 9: Sorting Sorting (229-249) & Binary Searching (365-371) Chapter 15: Windows Forms Forms, properties, events	
17	M	Continued...	Project 2 Due Project #3 handout
19	W	Continued...	
24	M	Continued... College IDEA Survey	Project 3 Due
26	W	Final Exam	Comprehensive

Community Time

- AITP – Friday evenings (5:30pm Student Commons) Fall & Winter
- Other conferences Michigan - Dev Days/Nights, SQL Saturdays

Ethics

Electronic Class Attendance Report (E-CARs) occurs the 3rd week of class. I report participation in class based on active attendance and real effort. Do not ask me to help you commit fraud.

Attendance

I monitor attendance for the college. I do not award points for your attendance. Enhancing your learning... that is the reason you should attend class.

Can you learn without attending class? Some students can; however, your lack of attendance places a high burden on you to learn on-your-own. I have found through personal experience and also proven by students, that less learning takes place when students do not attend class. The amount of knowledge transferred in-class through discussion, demonstrations, and over-the-shoulder guidance is significant.

I do not teach this class via email and the short videos (if available) are not a replacement for your attendance and participation.

Make the decision to attend and participate in class!

Grading Plan

My goal in this course is that you learn the material.

Projects #1-3 have defined due dates (usually 2 weeks after introduction of project). You must turn in each project by the due date, or take a zero for that project. After my review, if I believe the project is incomplete, I will give you another week to advance the project.

It doesn't make sense for you to continue to the next objective when you are struggling with a prior objective. However, the college semesters offer us a defined time period to complete the content; therefore, if you have not finished sections of work early in the semester, it does not make sense for you to take the final exam without the knowledge to do so.

Course Book

See Textbooks & Readings

Commitment to Learning Community

I enjoy learning. I enjoy the entire atmosphere of learning. The new knowledge, the interactions, and the shared ideas. My goal is to foster a learning community.

You can learn by reading a book and then trial-error your way to a solution. Over time you will likely "get it". However, you will find that when you participate in a learning community... the time to "get it", is significantly less.

A quality work environment is a learning community. Sharing amongst peers can help you reach goals, seek new ideas, or spur new learning & work opportunities.

Each student should seek out learning communities - one which each participant can and will interact with others in the community. It is these interactions and sharing of ideas which will enable the greatest amount of learning to occur.

Support

Many of us may not have learned... *how to learn*. Sometimes the course material really stumps us. What is a struggling student to do? The college offers many resources to help struggling students. Our goal is that you succeed. However, it really is up to the student to seek out assistance.

Be Realistic

1. The sooner the better - most learning is cumulative - you need to know first parts, to continue to later parts
2. Missing (or skipping) classes and then trying to catch up is very frustrating, as you realize how difficult it is to recover lost time

Instructor

Speak with your instructor. Really! I would be glad to help you find what it takes to turn the light bulb on as you learn the course material. If you're not "getting it" in class, then let's arrange a time outside of class, so we can figure it out.

Class Network

Network with your classmates. Successful students work together to learn. Work this network!

Student Success Center

A great place to become familiar with, before you need help. The place to go if you need extensive help. Arrange *one-on-one* tutoring, information about "how-to-study", reading assistance, etc.