COURSE TITLE: **Beginning Programming (11.4180000)**
Teacher: Mr. Gary Liu
School: Tri-Cities High School
Address: 2575 Harris Street • East Point, Georgia • 30344
Phone: (404) 669-8200
E-mail: liug1@fultonschools.org

Description: This course introduces students to design, development, support and management of hardware, and software.

Materials: Learn Key and other appropriate software and supplementary materials will be used.

Objectives (BCS-BP-)

The Programming pathway is one of six pathways in the Information Technology Cluster. This cluster builds linkages in IT occupations for entry level, technical, and professional careers related to the design, development, support and management of hardware, software, and multimedia and systems integration services. Careers in Programming and Software Development involve the design, development, implementation and maintenance of computer systems and software, requiring knowledge of computer operating systems, programming languages and software development. People with expertise in programming and software development work with cutting-edge technologies to develop tomorrow's products for use by businesses and consumers. While many of the career opportunities in this area are in software companies, large organizations of other types, such as financial services and business, also offer many opportunities. People with expertise in programming and software development are in high demand, doing work such as creating the software that launches and runs NASA space shuttles. This pathway culminates with an opportunity for students to take the Sun Certified Java Associate offered by Oracle or Microsoft Certified Professional Developer as the end of pathway assessment. **See Appendix for detailed standards.**

Homework Late Work: Homework will be assigned, collected, and graded throughout the duration of the course. Late work will be accepted with a 10 - POINT DEDUCTION per day up to 3 days**. A grade of 50 will be given after three days up to the last day grades are entered for the corresponding progress report** (exception Semester Report - Final Grade). Thereafter, a grade of zero will be given unless special arrangements have been made with me before the deadline**.** Some credit is better than no credit!

Missed Work Due to Absences: Students are expected to be present. However, when applicable, it is the responsibility of the student to complete missed assignments, projects, quizzes, tests etc. within 1 week **A grade of zero will be given after 1 week unless special arrangements have been made with me before the deadline.** Students who have an **EXCUSED ABSENCE** will be allowed to turn in their missing work with no deductions. Students with an **UNEXCUSED ABSENCE** will be allowed to turn in their work with a **10 - POINT DEDUCTION**. Please fill out an absentee sheet and attach it to your missed work before turning it in.

Make-up work can be completed on Thursday's beginning at 4:00pm. If the student misses his/her appointment; he or she can reschedule once, thereafter, the grade becomes a zero.

Recovery: The intent of the recovery policy is to assist students by providing adequate opportunities to master course objectives in order to eliminate preventable failures.

Opportunities designed to allow students to recover from a low or failing **cumulative** grade will be allowed when **all work required to date has been completed** and **the student has demonstrated a legitimate effort to meet all course requirements including attendance**. Students must complete the work before they can recover it. Clearly there is a difference between recovery and make-up work. The recovery policy for this course is as follows:

- ❑ Recovery is for students who are failing, or close to failing (low cumulative grade).
- ❑ Recovery is for a cumulative grade, not for one test, project, etc.
- ❑ Students may recover up to a maximum grade of 70 on recovery work.
- ❑ Recovery must be student initiated.
- ❑ Recovery work will not be granted to change a grade after a semester is over.
- ❑ All recovery work must be completed within ten school days prior to the end of the semester.
- ❑ The recovery assignment is at my discretion.
- ❑ I will establish a reasonable time period for recovery work to be completed during the semester.

Instructional Methodologies:	Class lecture/discussion/demonstration, audio-visual aids, individualized instruction, guest speakers, field trips by grade level (if applicable).
Discipline:	All student handbook rules apply. In addition students are expected to take care of equipment, textbooks, and supplies, as well as, exhibit respect for fellow students and faculty members. Please see guidelines posted in class.
Parent/Teacher Communication:	Communication with parents about academic progress, behavior and attendance will be handled via progress reports, e-mail, phone contact, conference, or a combination of these methods.
Career Opportunities:	Software Engineer, Technology Specialist , Software Developer, Game Designer
Test/Quizzes:	Tests will be based on information in the text that will be covered in class and any other supplementary information given in notes. Students Academic Portfolio will count as one major test grade. Information on the procedures and format of your student portfolio will follow.
Class Works:	Class works includes students' participation, assignments and quizzes. There will be several assignments and small presentations. Each person is expected to do their part.
Work Ethics:	Each student must demonstrate good work ethics in class (i.e. punctuality, not interfering with the learn process of another student, compliance to classroom rules, etc.). Failure to do so will cause your work ethic grades to suffer.

Grading Evaluation:	Daily Work	45%
	Final Exam	15%
	Project	10%
	Quizzes/Tests	20%
	Work Ethics	10%

Grading Scale:	100-90	A
	89-80	B
	79-70	C
	Below 70	F

Tutorial: Tutorial time is 8:00am on Monday through Thursday. After school by appointment only.

All Tri Cities students are required to have an Academic Portfolio for each of their classes. Academic Portfolios must be maintained in electronic format by following the template provided.

Each student will be expected to bring to class daily his/her notebook, workbook, paper, pencil and any other needed classroom supplies and materials.

The Georgia Department of Education has made available two assessments to ascertain the competence of our career technology students. The assessments are Work-Ready and End-of-Pathway. The Work-Ready assessment measures the skills of Georgia's workers, determines valuable job training opportunities, and assures companies that the state can provide a long-term, qualified labor supply, and is a requirement for some Georgia jobs. The End-of-Pathway assessment measures the level of technical skill attainment of each career pathway completer. The Work-Ready assessment will be administered to all work-based learning students and all seniors currently enrolled in a CTE course. The End-of-Pathway assessments will be administered to all CTE students who have finished or are finishing a pathway. Tests will be administered in October and April.

Student Name _____

I have read this course syllabus and I understand all information presented. In the event that I have questions, I will contact the teacher.

CAREERS

Students will explore careers in the field of computing.

BCS-BP-1. Students will explore careers in computing.

a. Describe the daily tasks and responsibilities of a professional in the field of computing.
b. Compare and contrast the top jobs in computing.
c. Explore careers that combine computing with another field.

Academic Standards:

ELA11W2 The student demonstrates competence in a variety of genres.
ELA11W3 The student uses research and technology to support writing.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

Sample Task

Job shadow a professional in the field of computing

HARDWARE AND SOFTWARE COMPONENTS

Students will demonstrate an understanding of the relationship between hardware and software in program execution.

BCS-BP-2. Students will describe the major parts of a processor and how the processor handles execution of a machine language program.

a. List and describe the function of the major components of the processor (ALU, registers, program counter, etc).
b. Describe the steps in the execution of a simple assembler program.
c. Choose an assembler program that is equivalent to a small high-level program.
d. Solve simple binary arithmetic problems.
e. Explain the interaction between the operating system and the processor.

Academic Standards:

MM3P4 Students will make connections among mathematical ideas and to other disciplines.
ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

BCS-BP-3. Students will explain the process that turns a high-level language program into something a computer can execute.

a. Explain the purpose of a compiler, interpreter, and assembler.
b. Compare and contrast a compiler and an interpreter and specify languages that use each.
c. Compare and contrast high-level and low-level languages for different uses.

BCS-BP-4. Students will explore different representations of images and music in a computer.

a. Explain the difference between bitmapped and vector-based representations of images.
b. Explain the difference between MIDI and sampled sound representations of music.
c. Write a program to modify a bitmapped image and/or sampled sound.

Academic Standards:

MM4A10 Students will understand and use vectors.
ELA11W2 The student demonstrates competence in a variety of genres.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

Sample Tasks:

Read http://computer.howstuffworks.com/microprocessor3.htm on how microprocessors work. Role-play the parts of a processor and the execution of a simple machine language program.

Complete the lessons at http://www97.intel.com/discover/JourneyInside/TJI_DigitalInfo_lesson4/default.aspx on binary numbers and http://www97.intel.com/discover/JourneyInside/TJI_DigitalInfo_lesson5/default.aspx on adding binary numbers and complete the activity on adding binary numbers.

Change all positive values in a sound to the maximum positive value and all negative values in a sound to the maximum negative value.

Reverse a sound.

Write a program that mirrors the left side of a picture onto the right side.

PROBLEM SOLVING

Students will practice the application of problem-solving strategies to develop and increase logical thinking skills.

BCS-BP-5. Students will design algorithms and programming solutions for a variety of computational problems.

a. Apply, test, analyze, and adjust problem-solving algorithms.
b. Design, test, analyze, and adjust coding solutions based on problem-solving algorithms.
c. Analyze and discuss coding solutions for elements of thoroughness and correctness.
d. Describe how recursion can be used to solve a problem.

Academic Standards:

SCSh3 Students will identify and investigate problems scientifically.
MM3P1. Students will solve problems (using appropriate technology).
ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11W2 The student demonstrates competence in a variety of genres.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

PROGRAMMING

Students will use basic programming techniques to design, implement, and solve simple problems using an object-oriented programming language.

BCS-BP-6. Students will design solutions for simple programs using basic programming techniques and constructs.

a. Implement techniques such as conditional statements, iterative statements, and variables to solve simple problems.
b. Utilize basic mathematical expressions to solve simple problems.
c. Use arrays and lists where appropriate.
d. Comment programs to aid program readability.
e. Test and debug simple programs.
f. Analyze and explain simple programs involving fundamental programming constructs.

Academic Standards:

ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11W2 The student demonstrates competence in a variety of genres.

BCS-BP-7. Students will use and develop algorithms to solve simple problems.

a. Develop algorithms to solve simple problems using psuedocode and/or flowcharts.
b. Interpret algorithms expressed in psuedocode and/or flowcharts to code solutions to simple problems.
c. Discuss the importance of algorithms in problem solving.
d. Identify properties of well-written algorithms in solving problems.

e. Interpret algorithms to write code that will implement searching and sorting techniques.

Academic Standards:
ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

BCS-BP-8. **Students will demonstrate knowledge of the relationships between classes.**
a. Utilize Class, Responsibilities, and Collaborator (CRC) in problem analysis.
b. Create UML diagrams to illustrate relationships between classes.
c. Describe and explain the implementation of "is-a" and "has-a" relationships.
d. Describe how using classes implements the ideas of encapsulation and information hiding.

Academic Standards:
ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

DATA STRUCTURES

Students will demonstrate the ability to process one-dimensional arrays, two-dimensional arrays, and lists. Students will choose an appropriate data structure for a given situation.

BCS-BP-9. Students will write programs that process one-dimensional arrays.
a. Calculate the average of the contents of an array.
b. Manipulate each element of an array.
c. Manipulate a section of an array.

Academic Standards:
ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11W2 The student demonstrates competence in a variety of genres.

BCS-BP-10. Students will write programs that process two-dimensional arrays.
a. Manipulate each element of a two-dimensional array.
b. Manipulate a section of a two-dimensional array.
c. Explain the difference between row major and column major two-dimensional arrays.
d. Explain how to process an array in a row-first or column-first manner.

Academic Standards:
MM3A7 Students will understand and apply matrix representations of vertex-edge graphs.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.
ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11W2 The student demonstrates competence in a variety of genres.

BCS-BP-11. Students will write programs that process lists.
a. Manipulate each element of a list.
b. Add items to a list.

Academic Standards:
ELA11W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals a satisfying closure.
ELA11W2 The student demonstrates competence in a variety of genres.

BCS-BP-12. Students will demonstrate knowledge of the basics of stacks and queues.

a. Explain the purpose of a call stack.

b. Describe the purpose of a printer queue.

c. Compare and contrast stacks and queues.

LIMITS OF COMPUTING

Students will determine how programming limits can affect business, and will recognize that there is a time hierarchy in programming.

BCS-BP-13. Students will discuss limits on computing as a result of programming complexity.

a. Determine the number of steps that a program will take to execute.
b. Compare and contrast the speed of different algorithms.
c. Identify the possible impact on a business when a computer programmer leaves a company.

Academic Standards:

ELA11W3 The student uses research and technology to support writing.
ELA11LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

BCS-BP-14. Students will identify time-related limits to computing.

a. Define polynomial time as it relates to computing.
b. Define exponential time as it relates to computing.

Sample Tasks:

Research *The Human Genome Project* and discuss its efforts to overcome computing limits.

CTAE Foundation Skills

The Foundation Skills for Career, Technical and Agricultural Education (CTAE) are critical competencies that students pursuing any career pathway should exhibit to be successful. As core standards for all career pathways in all program concentrations, these skills link career, technical and agricultural education to the state's academic performance standards.

The CTAE Foundation Skills are aligned to the foundation of the U. S. Department of Education's 16 Career Clusters. Endorsed by the National Career Technical Education Foundation (NCTEF) and the National Association of State Directors of Career Technical Education Consortium (NASDCTEc), the foundation skills were developed from an analysis of all pathways in the sixteen occupational areas. These standards were identified and validated by a national advisory group of employers, secondary and postsecondary educators, labor associations, and other stakeholders. The Knowledge and Skills provide learners a broad foundation for managing lifelong learning and career transitions in a rapidly changing economy.

CTAE-FS-1 Technical Skills: Learners achieve technical content skills necessary to pursue the full range of careers for all pathways in the program concentration.

CTAE-FS-2 Academic Foundations: Learners achieve state academic standards at or above grade level.

CTAE-FS-3 Communications: Learners use various communication skills in expressing and interpreting information.

CTAE-FS-4 Problem Solving and Critical Thinking: Learners define and solve problems, and use problem-solving and improvement methods and tools.

CTAE-FS-5 Information Technology Applications: Learners use multiple information technology devices to access, organize, process, transmit, and communicate information.

CTAE-FS-6 Systems: Learners understand a variety of organizational structures and functions.

CTAE-FS-7 Safety, Health and Environment: Learners employ safety, health and environmental management systems in corporations and comprehend their importance to organizational performance and regulatory compliance.

CTAE-FS-8 Leadership and Teamwork: Learners apply leadership and

teamwork skills in collaborating with others to accomplish organizational goals and objectives.

CTAE-FS-9 Ethics and Legal Responsibilities: Learners commit to work ethics, behavior, and legal responsibilities in the workplace.

CTAE-FS-10 Career Development: Learners plan and manage academic-career plans and employment relations.

CTAE-FS-11 Entrepreneurship: Learners demonstrate understanding of concepts, processes, and behaviors associated with successful entrepreneurial performance

Beginning Programming Pacing Guide

Weeks	Standards	Assessment
3 Weeks	**CAREERS** Students will explore careers in the field of computing. **BCS-BP-1. Students will explore careers in computing.** a. Describe the daily tasks and responsibilities of a professional in the field of computing. b. Compare and contrast the top jobs in computing. c. Explore careers that combine computing with another field.	Project 1: Career and Education
2 weeks	**BCS-BP-2. Students will describe the major parts of a processor and how the processor handles execution of a machine language program.** a. List and describe the function of the major components of the processor (ALU, registers, program counter, etc). b. Describe the steps in the execution of a simple assembler program. c. Choose an assembler program that is equivalent to a small high-level program. d. Solve simple binary arithmetic problems. e. Explain the interaction between the operating system and the processor.	Project 2: Simulation of Processing
1 Week	**BCS-BP-3. Students will explain the process that turns a high-level language program into something a computer can execute.** a. Explain the purpose of a compiler, interpreter, and assembler. b. Compare and contrast a compiler and an interpreter and specify languages that use each. c. Compare and contrast high-level and low-level languages for different uses.	Project 3: A research of programming languages
3 Weeks	**BCS-BP-4. Students will explore different representations of images and music in a computer.** a. Explain the difference between bitmapped and vector-based representations of images. b. Explain the difference between MIDI and sampled sound representations of music. c. Write a program to modify a bitmapped image and/or sampled sound.	Project 4: Images, Software, and Graphic Design Project 5: sound file library
2 Weeks	**BCS-BP-5. Students will design algorithms and programming solutions for a variety of computational problems.** a. Apply, test, analyze, and adjust problem-solving algorithms. b. Design, test, analyze, and adjust coding solutions based on problem-solving algorithms. c. Analyze and discuss coding solutions for elements of thoroughness and correctness. d. Describe how recursion can be used to solve a problem.	Project 6: Object, Methods, and Sprites.
2 Weeks	**BCS-BP-6. Students will design solutions for simple programs using basic programming techniques and constructs.**	Project 7:

	a. Implement techniques such as conditional statements, iterative statements, and variables to solve simple problems. b. Utilize basic mathematical expressions to solve simple problems. c. Use arrays and lists where appropriate. d. Comment programs to aid program readability. e. Test and debug simple programs. f. Analyze and explain simple programs involving fundamental programming constructs.	Algorithm and Game Design Process
5 Weeks	**BCS-BP-5. Students will design algorithms and programming** solutions for a variety of computational problems. a. Apply, test, analyze, and adjust problem-solving algorithms. b. Design, test, analyze, and adjust coding solutions based on problem-solving algorithms. c. Analyze and discuss coding solutions for elements of thoroughness and correctness. d. Describe how recursion can be used to solve a problem. **BCS-BP-6. Students will design solutions for simple programs** using basic programming techniques and constructs. a. Implement techniques such as conditional statements, iterative statements, and variables to solve simple problems. b. Utilize basic mathematical expressions to solve simple problems. c. Use arrays and lists where appropriate. d. Comment programs to aid program readability. e. Test and debug simple programs. f. Analyze and explain simple programs involving fundamental programming constructs.	Final Project: A Game Design Project of Value-Added CS Model
2 weeks	**BCS-BP-6. Students will design solutions for simple programs using basic programming techniques and constructs.** a. Implement techniques such as conditional statements, iterative statements, and variables to solve simple problems. b. Utilize basic mathematical expressions to solve simple problems. c. Use arrays and lists where appropriate. d. Comment programs to aid program readability. e. Test and debug simple programs. f. Analyze and explain simple programs involving fundamental programming constructs.	Project 8: anatomy and review of past projects
4 weeks	**BCS-BP-7. Students will use and develop algorithms to solve simple problems.** a. Develop algorithms to solve simple problems using psuedocode and/or flowcharts. b. Interpret algorithms expressed in psuedocode and/or flowcharts to code solutions to simple problems. c. Discuss the importance of algorithms in problem solving. d. Identify properties of well-written algorithms in solving problems.	Project 9: algorithm and canvas Project 10: an animation

	e. Interpret algorithms to write code that will implement searching and sorting techniques.	
4 weeks	**BCS-BP-8**. **Students will demonstrate knowledge of the relationships between classes.** a. Utilize Class, Responsibilities, and Collaborator (CRC) in problem analysis. b. Create UML diagrams to illustrate relationships between classes. c. Describe and explain the implementation of "is-a" and "has-a" relationships. d. Describe how using classes implements the ideas of encapsulation and information hiding.	Project 11: asset creation and class/object management
3 weeks	**BCS-BP-9. Students will write programs that process one-dimensional arrays.** a. Calculate the average of the contents of an array. b. Manipulate each element of an array. c. Manipulate a section of an array. **BCS-BP-10. Students will write programs that process two-dimensional arrays.** a. Manipulate each element of a two-dimensional array. b. Manipulate a section of a two-dimensional array. c. Explain the difference between row major and column major two-dimensional arrays. d. Explain how to process an array in a row-first or column-first manner.	Project 12: data and array
4 weeks	**BCS-BP-12. Students will demonstrate knowledge of the basics of stacks and queues.** a. Explain the purpose of a call stack. b. Describe the purpose of a printer queue. c. Compare and contrast stacks and queues. **BCS-BP-13. Students will discuss limits on computing as a result of programming complexity.** a. Determine the number of steps that a program will take to execute. b. Compare and contrast the speed of different algorithms. c. Identify the possible impact on a business when a computer programmer leaves a company. **BCS-BP-14. Students will identify time-related limits to computing.** a. Define polynomial time as it relates to computing. b. Define exponential time as it relates to computing.	Project 14: queues and order Project Final: Large project management and computer limitation