

ICS4U Course Feedback

Project Management
<i>Students apply standard project management techniques to plan, develop, produce, and close a student-managed project through the creation of a single modular computer program to master the software development process.</i>

<div>Project Management</div> <div>PLAN = Project Planning, CH1 = Check-in 1, CH2 = Check-in 2, DEV = Project Development and Production</div>						
We are learning to manage the software development process effectively through all of its stages (B1)						
Success Criteria (I can...)	IDQ	B	D	P	C	E
create a software project plan by producing a software scope document and determining tasks, quality standards, deliverables, and schedule.	• PLAN	• PLAN	• PLAN	• PLAN	• PLAN	• PLAN
use an appropriate project management tool to manage project components (i.e. charts, calendar, ...)	• PLAN • CH1 • CH2	• PLAN • CH1 • CH2	• PLAN • CH1 • CH2	• PLAN • CH1 • CH2	• CH1 • CH2	• CH1 • CH2
develop the software product according to the original project plan (i.e. ensuring the software meets end user needs, functions as intended, and can be produced within quality standards, budget, and timelines).	• DEV	• DEV	• DEV	• DEV	• DEV	• DEV
produce the software according to the specifications (i.e. code, test, and prepare for deployment) and create user documentation and training materials.	• DEV	• DEV	• DEV	• DEV	• DEV	• DEV
We are learning to contribute and reflect on standard project management techniques for a student-managed team project (B2)						
Success Criteria (I can...)	IDQ	B	D	P	C	E
contribute, as a team member, to the planning, development, production, and close out of a large software project.	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV
meet project goals and deadlines by managing individual time during a group project.	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV	• PLAN • CH1 • CH2 • DEV
Overall teacher comments:						

ICS4U Course Feedback

Student unit reflection or comments:

ICS4U Course Feedback

Data Structures and Algorithms

Students design, create, test, and analyze complex algorithms and data structure through the creation of computer programs with clear documentation to enhance their logical thinking skills.

IDQ = insufficient data or quality (less than 50%), B = beginning (50-59%), D = developing (60-69%), P = proficient (70-79%), C = comprehensive (80-89%), E = exemplary (90-100%)

Data Structures and Algorithms

RVN = Review and New, DST = Data Structures, EDS = Extended Data Structures, ALG = Algorithms, REC = Recursion (1 = feedback submission, 2 = final submission)

We are learning to work with data types and proper code maintenance techniques (A1, A4)

Success Criteria (I can...)	IDQ	B	D	P	C	E
work independently, using support documentation, to resolve syntax issues during software development.	<ul style="list-style-type: none"> RVN DST 	<ul style="list-style-type: none"> RVN DST 	<ul style="list-style-type: none"> RVN DST 	<ul style="list-style-type: none"> RVN DST 	<ul style="list-style-type: none"> RVN DST 	<ul style="list-style-type: none"> RVN DST
create fully documented program code according to industry standards.	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS
use one-dimensional arrays of compound data types (i.e. objects)	<ul style="list-style-type: none"> EDS ALG 	<ul style="list-style-type: none"> EDS ALG 	<ul style="list-style-type: none"> EDS ALG 	<ul style="list-style-type: none"> EDS ALG 	<ul style="list-style-type: none"> EDS ALG 	<ul style="list-style-type: none"> EDS ALG

We are learning to design and apply modular programming concepts including complex data types (A2, C1)

Success Criteria (I can...)	IDQ	B	D	P	C	E
create a modular program that is divided among multiple files.	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST
modify existing modular program code to enhance the functionality of the program.	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS
decompose a problem into modules, classes, or abstract data types using an object-oriented design methodology (using classes, methods,... using UML, etc.).	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS 	<ul style="list-style-type: none"> DST EDS
apply the principle of reusability in program design (use of inheritance).	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS 	<ul style="list-style-type: none"> EDS

We are learning to design, write and analyze complex algorithms and subprograms (A3, C2)

Success Criteria (I can...)	IDQ	B	D	P	C	E
read from, and write to, an external file (i.e. database, API, text file, binary file, ...).	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST 	<ul style="list-style-type: none"> DST
compare the efficiency of sorting algorithms, using run times and computational complexity analysis.	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG
compare the efficiency of linear and binary searches , using run times and computational complexity analysis.	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG 	<ul style="list-style-type: none"> ALG

ICS4U Course Feedback

Overall teacher comments:

Student unit reflection or comments:

ICS4U Course Feedback

