

CPT 111 - ASSIGNMENT 1 RUBRIC

Group Number: _____

Criterion	Poor (1-2)	Fair (3-4)	Moderate (5-6)	Good (7-8)	Excellent (9-10)	Marks	Comments
Program Completeness <ul style="list-style-type: none"> • read from file • working cohesively • clear • have inheritance, composition, friend function, • single array of object used in main • At least one overloading operator • Runnable 	Program work but class run individually. Inheritance or composition is not communicating well. Incomplete requirements fulfilled.	Missing two or more components.	Contain all the specifications. Program is not very cohesive.	Fulfils basic program completeness.	Complete and following program specification. All classes communicating correctly and consistently.		
Comply to OO Programming <ul style="list-style-type: none"> • inheritance & composition is communication correctly • only one or two (max) friend function 	Not fulfilling encapsulation and information hiding. Uncontrol use of friend function	Coding style is not easy to be read. Naming convention is not very consistent.	Program sufficiently easy to read and maintain. Composition is private and used wisely. Inheritance is sensible and correct.	Program easy to read and maintain. Composition is private and used wisely. Inheritance is sensible and correct.	All coding style are easy to read and easy to maintain. Inheritance, composition, friend function and overloading operator.		

Criterion	Poor (1-2)	Fair (3-4)	Moderate (5-6)	Good (7-8)	Excellent (9-10)	Marks	Comments
<ul style="list-style-type: none"> • Encapsulation and information hiding is fulfilled • Only one object • Program easy to read 				Use of overloading operator is correct. Use of friend function is justifiable.	Friend function is justifiable. All variables usage are at where it is supposed to.		
Complexity and Sophistication <ul style="list-style-type: none"> • Program is well thoughts and design • explanation on solving method is described • program correctness • array of object is justifiable • Using multiple file inclusion 	Program lack of depth.	Program resembling procedural programming.	Program portray understanding of advance class processing.	Program is well design and portray understanding of advance class processing.	Program is well thought and portray strong understanding of advance class processing.		
Video Presentation & Content	No clear direction.	Direction of the explanation is somewhat given but not clear.	Describe the problem. Describe the classes' component. Clear test cases.	Problem the program can solve is described. The communication among classes and components are sufficiently covered. Clear test cases.	Clear explanation on the problem and the structure of the classes and how they communicate with each other. Clear test cases.		

Criterion	Poor (1-2)	Fair (3-4)	Moderate (5-6)	Good (7-8)	Excellent (9-10)	Marks	Comments
Report <ul style="list-style-type: none"> • problem to be solved in address clearly • explanation on solving method is described • UML reflecting the program • Sample test cases provided 							

Criterion	Marks	Comments
PENALTY		
Using <code>friend</code> class	- 50%	
Composition is <code>public</code> or <code>protected</code>	- 40%	
Multiple objects in <code>main()</code>	- 60%	
Groups copying each other	- (all marks)	

Criterion	Marks	Comments
Bonus		

TOTAL				
-------	--	--	--	--