RUBRIC ASSIGNMENT 2: (i) Maze Searching **OR** (ii) Knight's Tour

Program does not compile. **Assignment receives a grade of zero.**	**0**

Criterion	Good		Satisfactory		Poor		Not Met	
Coding Style	Code is clean, understandable, and well-organized. Good use of indentation, whitespaces, paragraphing and variable names.	**1.5**	Minor issues with consistent indentation, use of whitespace, variable naming, or general organization.	**1**	At least one major issue with indentation, whitespace, variable names, or organization.	**0.5**	Major problems with at three or four of the readability subcategories.	**0**
Documentation	Comments are well written and clearly explain the code. Contains a program header.	**1.5**	One or two places that could benefit from comments are missing them or the code is overly commented.	**1**	Program header missing, complicated lines or sections of code uncommented or lacking meaningful comments.	**0.5**	No program header or comments present.	**0**
Usability	Input and output prompts are intuitive and specify appropriate units.	**1.5**	Input and output prompts are somewhat intuitive and uses units.	**1**	Input and output prompts are confusing, and no use of units.	**0.5**	Input and output prompts are non-existent.	**0**
Reflection document	Clearly describes the Ada design process.	**3**	Somewhat describes the Ada design process.	**2**	Describes the Ada design process in a very limited manner.	**1**	No reflection document.	**0**
Correctness	Program always works correctly. i.e. all test cases are correct.	**6**	Program is partially correct. i.e. all test cases produce results which are partially incorrect.	**4**	Some test cases fail completely. e.g. one passes, the other fails.	**2**	Program does not function at all - may contain logic errors.	**0**

Not Met - Criteria not met.
Poor - Evidence that the student has mastered this learning objective is not provided, unconvincing or incomplete.
Satisfactory - Evidence shows that the student has generally attained this objective. Some of the program structures remain in their legacy state.
Good - Evidence demonstrates that the student has mastered this objective.

RUBRIC ASSIGNMENT 2: (i) Maze Searching **OR** (ii) Knight's Tour

	Good		Satisfactory		Poor	
Effectiveness	Design conveys all important elements, constructs, and behaviors of Ada. It demonstrates a deep understanding of programming in Ada.	**2.5**	Design conveys many key elements, constructs, and behaviors of Ada. Some structures are designed in an unusual manner, but such treatment is documented.	**1.5**	Design minimally conveys key elements, constructs, and behaviors. It shows a superficial understanding of programming in Ada .	**0.5**
Modularization	The program is broken into well thought out elements that are of an appropriate length, scope and independence.	**2.5**	Code elements are generally well planned and executed. Some code is repeated that should be encapsulated.	**1.5**	Code elements exist, but are not well thought out, are used in an arbitrary fashion, or do not improve program clarity.	**0.5**
Efficiency	The code is extremely efficient without sacrificing readability and understanding.	**2.5**	The code is fairly efficient without sacrificing readability and understanding.	**1.5**	The code seems complicated and unnecessarily long.	**0.5**

Requirements	Good		Satisfactory		Poor	
Maze Searching	Program prompts user for filename, and obtains input from a file (3) . Data structure is in the form of an Ada package (3).	**6**	Program obtains input from a file, but some elements are missing, e.g. hardcoded filenames, or no package.	**4**	Limited fulfillment of requirements.	**2**
Knight's Tour	Program prompts for valid input (3). Program results are output to screen and file (3).	**6**	Program obtains input from user and output to file, but some elements are missing, eg static board size.	**4**	Limited fulfillment of requirements.	**2**
Data Structure (BOTH)	Algorithm uses some form of data structure (e.g. stack, queue, list, or recursion)	**5**	Algorithm uses an ad-hoc data structure (i.e. it is not clear from the code - likely patched together)	**2.5**	Algorithm uses no data structure (e.g. stack, queue, list, or recursion)	**1**

DEDUCTIONS	**-2**	**-1**	**0**
Run-time	Program has serious run-time issues and crashes.	Program has some run-time issues and crashes on occasion.	Program has no run-time issues and always functions.
Compiler warnings	Program produces >4 compiler warnings.	Program produces ≤4 compiler warnings.	Program produces no compiler warnings.

GRADE	OVERALL	LATE DEDUCTION: (**1** / **2** / **3** days)		FINAL GRADE
	30 / 32	× **0.9** / **0.8** / **0.7**	=	**30** / **32**

February 2017