Checklist for self-evaluation

Self-evaluation of Moria Mines (Textual adventure game)

Section A. Evaluation of the quality of the program code.

Please write in the box and/or ✓ or X against each of the points in second column below.

Evaluation of the quality of the code:

Criteria	Values	Tick (✓)
What are your test data? And does the program run on these test data?		✓
Did you check that all your identifiers(variables, constants, classes and objects are appropriately	a. Name (e.g variables names are descriptive and should start with lowercase letters and class should start with uppercase etc.) b. Defined/declared (e.g check that variables(local, instance and static) are declared the right place, methods have parameters, return value type and body that reflects the method name) c. Initialized where appropriate value d. Invoked appropriately e. All identifies are used in your program to contribute to fulfill the program specification or have an appropriate role in the	✓

	program f. Access modifiers (private, projected and public) g. Scope and visibility of the identifiers understood	✓
Did you modularize your code so it is easy to understand?	a. Are there same pieces of code that are appropriate for method abstraction (redundant code) b. If a method is too long, it may be good idea to think about method modularity using method abstractions	Swith case ↳ W, S, E, N Lidt redundant kode
Control flow ✓ ✓ ✓ ✓	a. All loop should terminate at some point in the program b. Switch statements should have a default case c. Avoid using multiple exit from a loop. Rethink about your algorithm if you think you need to do this d. Are there too many nested loops/conditions? Rethink about your algorithm if you think you need to do this.	✓
Input/Output	a. Does the program cater for all types of input? b. Are exceptions handles so that the program ends gracefully? c. Does the program run without breaking?	✓
Boolean expressions	a. Are Boolean expression is short and easy to understand with regard to the program logic?	✓
Documentation of the code ✓	a. Is it clear from the comments that what the each segment of code will do? b. Do the codes do what the comments say for each appropriate segment?	✓

	c. Do the comments in the beginning of the methods explain what the method will actually perform? d. Do all the declarations(variable, class, methods) have appropriate comments? e. Are critical algorithms explained in plain language?	
Program layout	a. Indentation style is consistent. b. Code within a bloc (e.g. inside a loop) should be indented c. If a block is nested within another block the inner block's body should be indented relative to the enclosing block. d. Avoid excessive "stairstep" indentation. If problem reduce the number of spaces per indentation or switch to vertical style temporarily.	
Data encapsulation	a. Proper use of visibility modifiers and getters/setters b. Are local variables are visible only within the declared method, constructor, or block c. Access modifiers can be given for instance variables d. Instance variable are declared private e. Instance variables are decla in a class, but outside method, constructor or a block.	
Object oriented design	a. Does each class have distinct role e.g. controller class and entity class	

Section B. Evaluation against the program requirements.
Please write small note against each of the requirements below.

Requirements	Your comments/notes
Is you game to read user input from the console and also output text to it ?	Ja
Does your show all elements used in the program. i.e.: all attributes, methods including parameters and return types and associations including multiplicity and navigation direction.	Ja
Did you create a player class that holds the amount of gold picked up so far	Ja
Did you create a maze of rooms that the player can navigate? Did you populate it?	Ja
Did you create a room class. Does each room object have a text description and some gold that can be picked up? Does the room have four tunnels? Which one is your starting room?	Ja. Det er "StartRoom"
How do you keep track of the rooms?	Separate "Room" objekter
How does your program end? What conditions makes it end of the game?	Haver i slutRoom. Der ifm. 0 HP. Hvis "player" skriver 'exit' i konsollen. Spil loopet afbrydes vha. en boolean
Does the user get a menu of options to choose from once they enter a room? Do you have error-handling based on user input? Are there appropriate message for the wrong input? Do you have a mechanism for user to ask for help?	Ja