

Feature table

Feature	Mark	Justification
Classes	5	5+ distinct classes with different responsibilities
Inheritance	2.5	2 layer inheritance, e.g. GameObject → Asteroid
Polymorphism	10	Dynamic polymorphism implemented in Game.h to update object positions
Abstract classes	5	GameObject class is abstract
Display object state	5	Object states displayed on-screen using various SFML objects
Change object state	5	Object states changed via user input
Store/recall data	5	Multiple object instances stored and recalled using std::vector, scores stored in and recalled from leaderboard.txt
Write/read data	5	Program writes to and reads from leaderboard.txt
User menu	5	Intuitive and functional user menu
I/O	5	Program requires user input and returns necessary visual output
Comments	5	Frequent, informative, succinct comments across all files
Code style	5	Variables are consistently and sensibly named, code is well organised and correctly indented
Testing	10	Structured unit tests used for a considerable portion of the code
Makefile	5	Makefile builds successfully with multiple targets
Error handling	5	No unexpected behaviour or errors present, program handles unexpected input
Input validation	5	Program only accepts valid input
Plan similarity	5	Core functionality and features unchanged
Dynamic allocation	5	Multiple uses of new and delete, e.g. in Game.h
Total mark	97.5	