AP Computer Science
Game Project 2017

I. Introduction

This project assignment is designed to allow you to develop a Game using the GridWorld GUI and classes. The game is to be one that you or someone you know could and would play. ***You must develop your game on the classroom computer, and it must run on the classroom computers.***

Your game must be unique! You may not produce a game that is the same as another student's.

Our goal is to allow you to exhibit your programming creativity while producing a program that is a working tool for someone. You will be able to choose the subject matter and the operational features of the program. You will create user instructions, design the program, code the program, debug it, and produce the final documents.

It is hoped that the final product will be something you can take with you. That is, something that will summarize everything you know about Java and programming. If done well, you will be able to use it in the future as a reference which will help you in your Computer Science education.

II. Grading

The project will count as a **major test grade**. The grading check sheet lists points allocated for each section. Grading will be done each day. As soon as you have a portion of the project completed, it may be evaluated.

Folder and User Instructions **must be done first**. No exceptions.

Please note that no more than 2 items may be graded in one day. If you wait until the last day and have 7 items to grade, only 2 of these may receive credit. You must pace yourself and earn points on a daily basis. Makeup checks from an absence, including AP Exams, must be made up by the 2nd day back. Makeup checks will only be granted if they are needed, and the absence is excused.

The project is to be presented to your class for the last 5 points of your grade – this 5 points is not counted in the 2 per day count.

The absolute last time to evaluate is Wednesday, May 24, 2017 at 4:30.

III. Sample User Instructions

see example on separate sheet

IV. Grading Check sheet

on separate sheet

Jamie Programmer
AP Computer Science I
May 22, 2015

Game: Collapse
Players: One
Interface: Mouse

Goal:

The game of Collapse begins with a 20 x 20 grid filled halfway with random colored rocks that are blue, green, red, aqua, orange and purple. The goal is to get as many points as possible before any column of rocks reaches the top of the grid, ending the game.

Game Play:

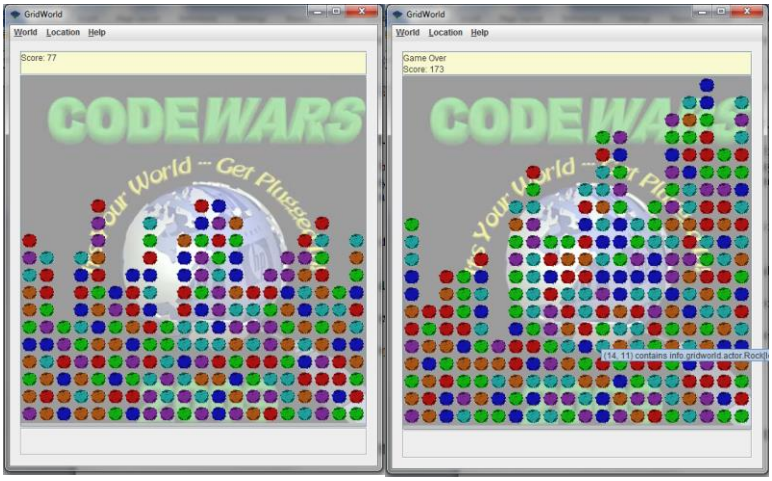
Instructions for game play are displayed in the message area at the top of the grid. The user clicks on "Run" to begin the game. The user clicks on a square to remove it, along with any same-colored rocks that are connected to it horizontally or vertically. Single rocks with no same-colored neighbors cannot be removed. As rocks are removed, the remaining rocks fall down into the empty places below. As the user plays, rocks fall from above, filling up the grid. The new rocks fall more quickly the longer the user plays the game, making it continually more challenging.

Scoring:

The user scores a point for each square removed from the grid. The user's score is continually displayed in the message area at the top of the grid. Game play continues until one or more columns of rocks reaches the top of the grid. The user's final score is compared to the current top 20 scores, and if they qualify in the top 20, they enter their name to be recorded.

Data:

The list of players with the top 20 scores is saved in a file. It is updated each time a player makes a top 20 score, and can be displayed at any time.



** pictures added after game completed – can be sketches initially if you want*

AP Computer Science Game Project

Name _____

Period/Game Title _____

Points		Description	Points	Initial
5	* Folder	Good condition with brads and two pockets. Affix this grading sheet to the front of the folder.	_____	_____
5	* Printed User Instruction Manual	In brads of folder. At least one full page. No more than two.	_____	_____
5	Online Help: Rules of the game	Display at least 4 full paragraphs which describe the game usage. This will be very similar to the User Instructions, but will be "online".	_____	_____
5	GIF's	Appropriate images for game classes shown in grid	_____	_____
5	Step /Run	Let user start the game when ready as appropriate	_____	_____
10	Mouse and/or keyboard interaction	must be clear and easy for user	_____	_____
5	Opening data file	Opens file. Error message if not found.	_____	_____
5	Saves game data	Saves file. Message stating "Game was Saved…"	_____	_____
5	Winner/Calculation	Display points and/or winner at end	_____	_____
5	Choice 1	_____ *Write in this blank what this choice does.*	_____	_____
5	Choice 2	_____ *Write in this blank what this choice does.*	_____	_____
10	Gameplay	Works as intended.	_____	_____
10	Grid/Board	Good user messages within grid. Good size and spacing.	_____	_____
5	State of Memory/UML Diagram(	complete AFTER coding your Classes/see pages 540-543 in textbook) In brads of folder behind User Instructions Must have data members, constructors & methods with parameters	_____	_____

These 3 items must be completed last, but source code and javadocs ARE counted in the 2 per day tally

5	Class presentation:	*(this item only does not count in the 2 per day checks)* Must include a description and demonstration.	_____	_____
5	Source code	*(Must be checked only after folder is turned in.)* Comments in each class with name, class period, date, description of class. Left pocket of folder: class with main and primary game class classes Right pocket of folder: all other classes. Digital copy will be collected at this time also.	_____	_____
5	Javadocs	Document each class and method and produce html and place in Submit.	_____	_____
		Total	_____	_____

* must be graded FIRST

Selection of "choice" options

Choice items are not simply added to get points.
Choice items are added to make the program more useful, realistic, fun.

Example:
In Collapse: a warning could be displayed when a column is nearing the top.

Possible choice items:
Increasing levels of difficulty
Additional computations of average length of runs (#1, #2, # 3) for each player.
Allow the user to use multiple files or allow user to enter file name to open/save to/from.
Make a bar chart of data (one bar for 1s, one for 2s, one for 3s, one for 4s)
Display significant ongoing data, such as health, available tools.
Allow user to choose a level of difficulty.

Since each game is different, and the topic is so broad, you should check with your teacher to be sure your choice option is appropriate before you begin implementing it.