National University of Computer and Emerging Sciences

Department of Computer Sciences

CS-118
Programming Fundamentals
Project

Course Instructor	Dr. Rabia Maqsood
Instructor	M. Hannan Farooq
Semester	Fall 2020

Objectives:

- All concepts of programming fundamentals
- Control Structure (Selection)
- Control Structure (Repetitive/ Loops)
- Function
- Array (one-dimensional, Multi-dimensional)
- File Handling

Note: Carefully read the following instructions (*Each instruction contains a weightage*)

1. Anyone found in copying the project from any other group or any other source, both the groups will be penalized with zero marks.
2. This game is a two person project so only 2 students are allowed per group. No cross-section is allowed.
3. You must submit the whole project to one member of the group.
4. There must be a block of the comments at the start of code, the block should contain a brief description of the functionality of the code. Also, comment with all functions about their functionality.
5. Proper indentation of code is essential.
6. Write a code in C++ language using Microsoft Visual Studio.
7. First, think about statement problems and then write/draw your logic on a copy.
8. After copy pencil work, code the problem statement on the Microsoft Studio C++ compiler.
9. At the end when you have done your project, submit a **complete project**
10. Submit your file in this format **20F-XXXX_20F-XXXX_CandyCrushGame.zip**.
11. The project is divided into two deliverables (For those who start a project in a group).
12. In the first deliverable, both group members discuss the complete project with each other and divided equal work (50 % work for each member), and submit recorded audio or video link on google classroom or Microsoft teams.
13. In the Final deliverable submit a **complete project file** with **your code copy-pasted on a word file with all possible output screenshots. Both project file and word file must be included in the .zip file.** Make your submission on Google Classroom (Make sure your submission is completed).

Grading Criteria

1. Your program must be stable which means it should do what it is supposed to do. You can achieve these criteria by knowing adequate knowledge of how the game works and by carefully follow and implement all the rules mentioned below.
2. Implement your code with an efficient structure which means the use of control, repetition, and all other features we covered in the course, must be utilized efficiently.
3. Be careful while using functions signature (prototype) and try to avoid using break and continue statements. Use only very essentials global variable otherwise do not a use global variable. Do not use bad programming and hard code techniques.

4. Use proper indentation and comments where they are necessary (use with function definitions, function calls, checks, loops, variables, etc.)

Problem: Candy Crush Game

How to play game

You must use this link, how candy crush game works

https://www.youtube.com/watch?v=6Lsz6FK0uIM

Introduction

Candy Crush Saga is a free-to-play match-three puzzle video game released by King on April 12, 2012, for Facebook; other versions for iOS, Android, Windows Phone, and Windows 10 followed. It is also a variation of their browser game Candy Crush.

In the game, players complete levels by swapping colored pieces of candy on a game board to make a match of three or more of the same color, eliminating those candies from the board and replacing them with new ones, which could potentially create further matches. Matches of four or more candies create unique candies that act as power-ups with larger board-clearing abilities. Boards have various goals that is complete within a fixed number of moves or limited amount of time, such as a certain score or collecting a specific number of a type of candy

Features to implement (Rules for the game)

Following are features/ rules you must carry out in your game in C++ language using all concepts you learn in the course.

This game is menu-driven which contains five options.

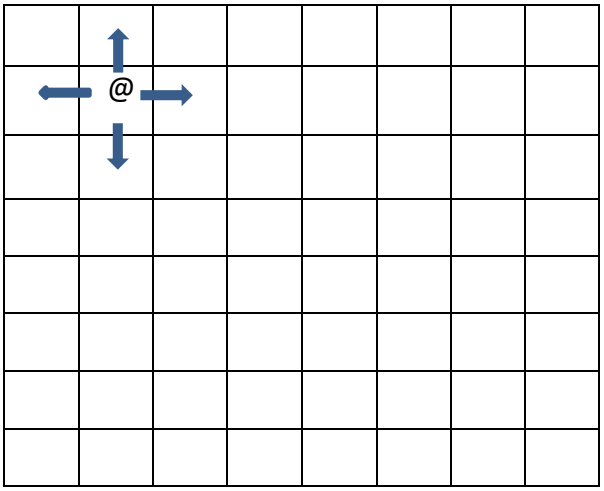
1. **Play Game:**
   - After selecting the play game option, the game gets the player's name by input the player's name from the user.
   - Then you show the player two modes
     1. Easy Mode
     2. Hard Mode
   - After select game mode, you need to implement a countdown timer. For this purpose, you need to use **windows.h** library and function **sleep(value)** use for the timer. You can use the **system("cls")** for clear screen purposes.
   - Your board should look like this board mention below

@	H	$	$	#	@	#	&
&	$	&	#	&	#	&	#
@	H	$	@	$	#	#	$
&	@	H	H	#	@	&	#
$	&	#	#	$	#	#	@
@	&	@	#	H	#	@	$
#	#	&	$	&	@	$	#
@	#	@	#	@	$	#	&

   - **Easy Mode:**
     1. The board has a size of 8x8 that appeared on the console.
     2. Your board contains at least five different kinds of candies. Your board should not contain the same candy on three or more consecutive blocks.

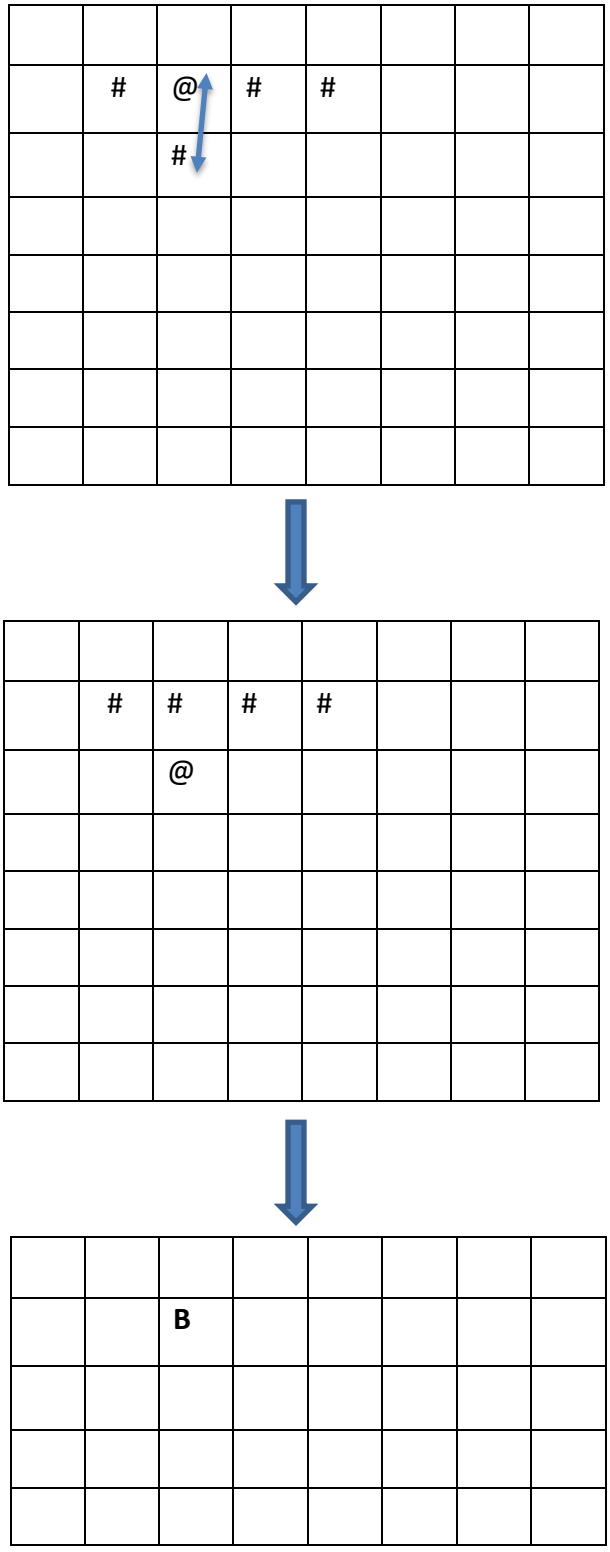
3. Your board contains at least five hurdles. These hurdles are not swappable.
4. When the game starts. Start timer from 60 seconds and when the timer is zero your game is over and display the message “Time’s Up”.

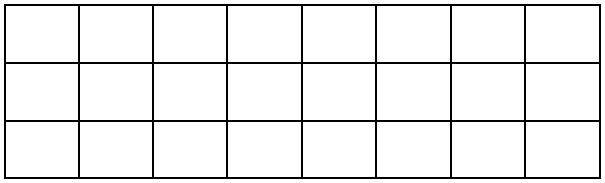
- **Hard Mode:**
  1. A board having a size of 12x12 appeared on the console.
  2. Your board contains at least eight different kinds of candies. Your board should not contain the same candy on three or more consecutive blocks.
  3. Your board contains at least 10 hurdles. These hurdles are not swappable.
  4. When the game starts. Start timer from 40 seconds and when the timer is zero your game is over and display the message “Time’s up”.
  5.
- After selecting the candy block to swap, the player must enter any directional key (top, bottom, right and, left) for swapping. Candies are only swapped with other candies in the neighborhood (top, bottom, right and, left) if three or more same candies exist on consecutive blocks.



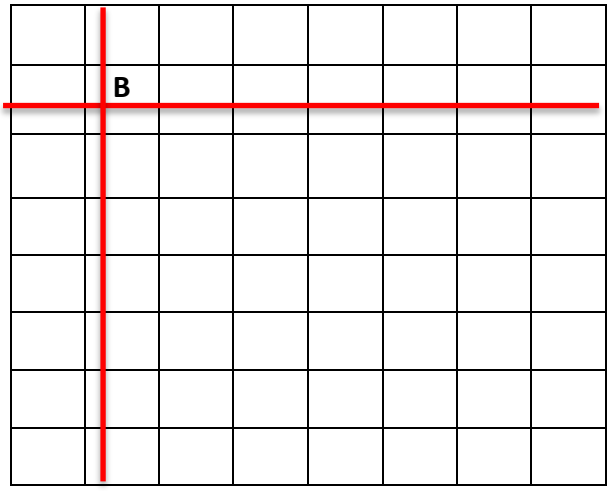
- If three candies are matched then the player will get 10 points with the message displayed on the console “Sweet” and new candies are placed in the position of matched candies.
- If more than three candies match then there will be a bomb candy appear on the position of a swappable block (as mention in the video) onboard

and other blocks filled with new candies then the player will get 20 points with a message displayed on the console “Delicious”. Some graphical representation is mentioned below





- If the bomb block swaps with candy then the whole row and column of that swapped block will be removed and that row and column block contain new candies then the player will get 30 points with a message displayed on console “Divine”.



- If not a single switch/swap available onboard then shuffle the whole board again with candies.
- There is a total of 15 moves when the player starts the game. After each valid move, the move count will be decreased by one until no move left. On zero move game will be over.

2. **Credits:** This section contains who created the game.
3. **Instructions/ Rules:** How to play the game with all rules to play a game.
4. **Score:** The score section will contain the score of all games played by any user. Whenever the users check the score list from the main menu it should display the top three highest scores of all times with their names.
5. **Exit:** Your game simply exit.

Your game provides a choice to the user/player to select an option from 1 to 5 to proceed with the game.

Proper code indentation will hold extra marks!

Best of luck

You are done with your Project, submit it on Google Classroom at the given time.