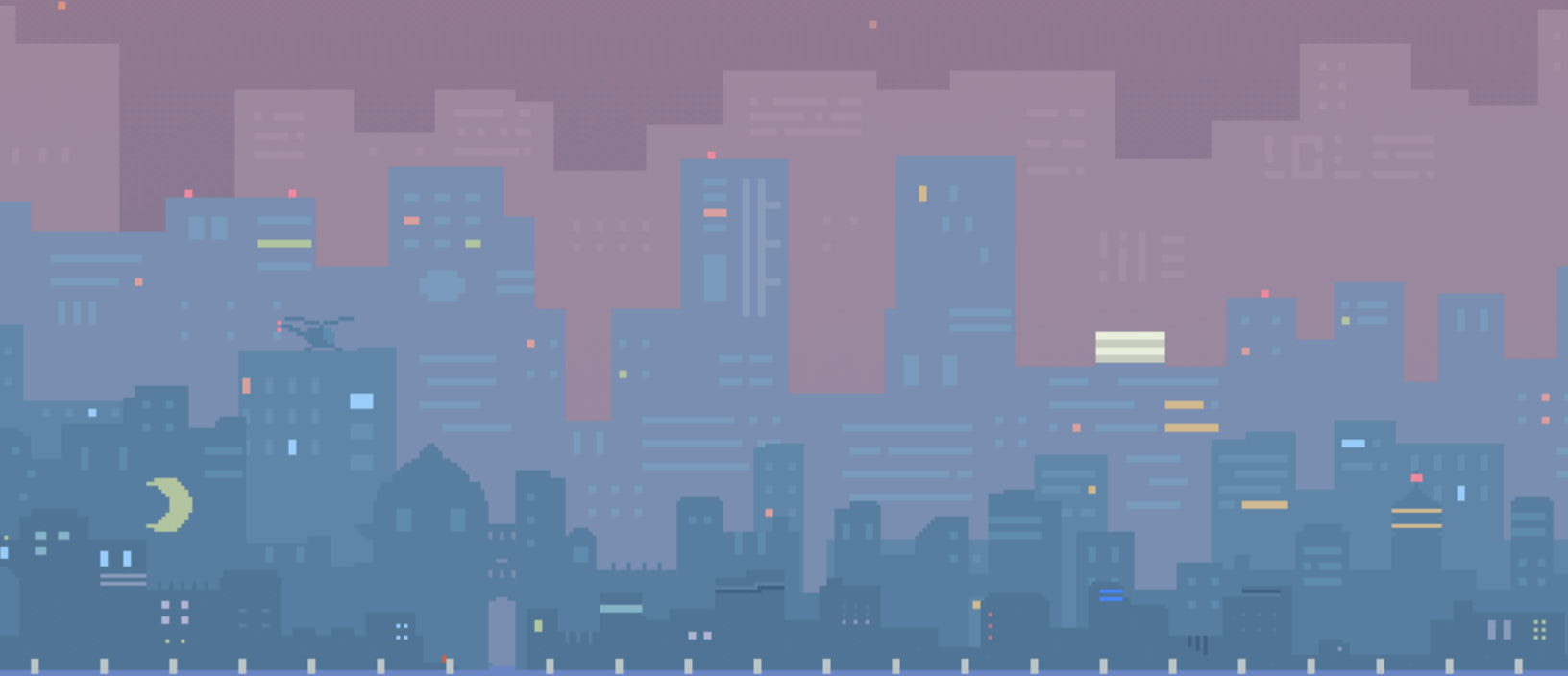


BAD ICE CREAM 4

user's manual



contents

introduction.....	1
booting.....	1
title screen.....	2
how to screen.....	2
credits screen.....	4
level selection.....	4
game screen.....	5
gameplay.....	6
fruit.....	8
monsters.....	9
endgame.....	10
bibliography.....	11

introduction

Bad Ice Cream is a tile-based browser game where players, who control ice-cream characters, complete levels by collecting fruit while avoiding monsters. Some fruit are stationary, while others move about randomly. Likewise, some monsters move very simply, while others may actively chase the player. Players are aided by their ability to shoot and break rows of ice blocks to manipulate fruits or impede monsters. However, if a monster touches the player, they die, and the level must be restarted. Each level is also timed, and the player is required to complete that level before time runs out. If the player has collected all the fruits in a level, has not been touched by a monster, and has completed all tasks within the time limit, they are considered to have won the level. Otherwise, they are considered to have lost the level.

This assignment is a clone of the Bad Ice Cream game, titled Bad Ice Cream 4 The monster, fruit, and tile graphics are taken directly from the game, and are not my own. The code, however, is my own.

booting

Upon booting the program, the user is greeted with a prompted for a password. The password is 100%. Enter 100% into the text field and press the **Check Password!** button to start the game. Anything other than 100% will not cause anything to happen.

title screen

After entering the correct password, the title screen will show, with Bad Ice Cream 4 displayed in the middle of the screen.

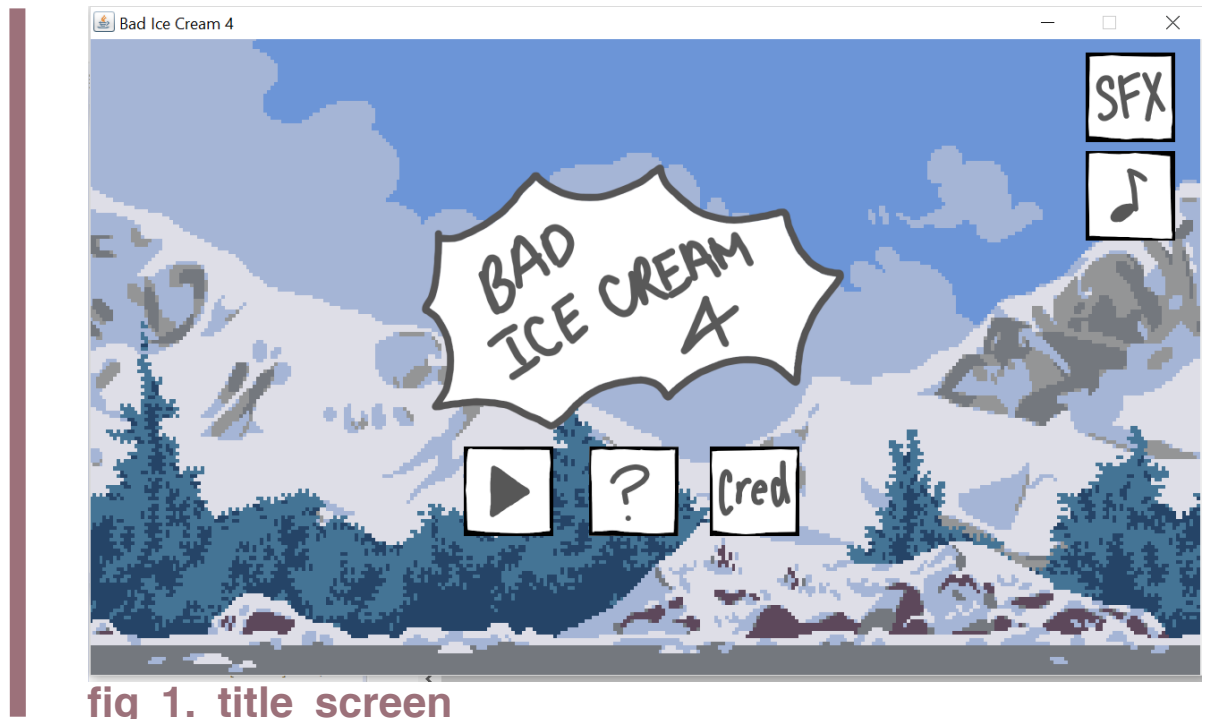


fig 1. title screen

In the top right, there is a music button and an SFX button. They are used to turn music and sound effects, respectively, on and off.

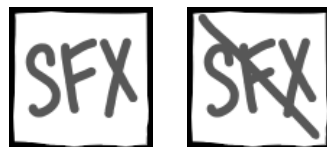


fig 2. SFX on and off



fig 3. music on and off

title screen

Underneath the Bad Ice Cream 4 logo, there are three buttons. The **Play** button on the left takes the user to the **Level Selection** screen. The **?** button in the middle takes the user to a **How To** screen with basic instructions on playing the game. The **CRED** button takes the user to the **Credits Screen**.



fig 3. Play, ?, and CRED buttons

how to screen

The **How To** screen shows a graphic with basic game instructions. The top right corner of the screen contains a left-facing arrow **Back** button in addition to the **SFX** and **Music** buttons. The **Back** button takes the user back to the **Title Screen**.

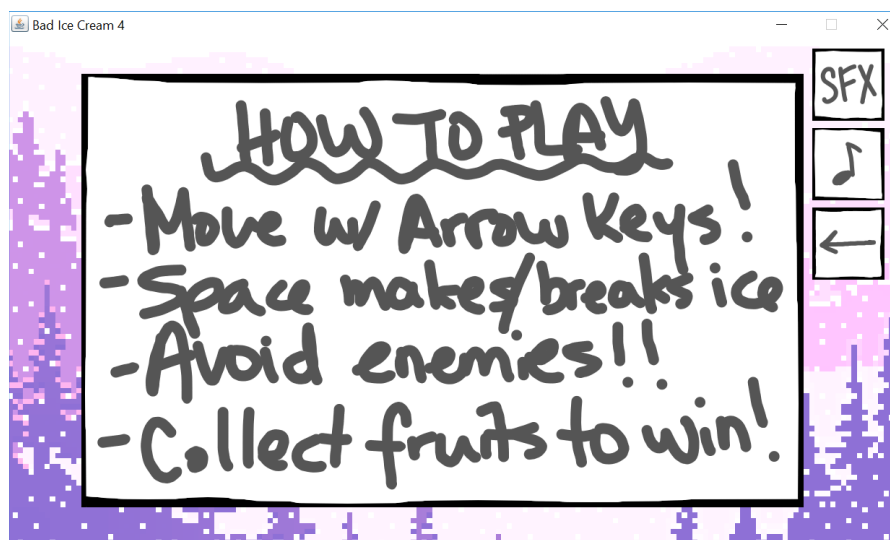


fig 4. How To Screen

credits

The Credits Screen shows a credits graphic. Like the How To Screen, the top right contains a left-facing arrow Back button in addition to the SFX and Music buttons. The Back button takes the user back to the Title Screen.

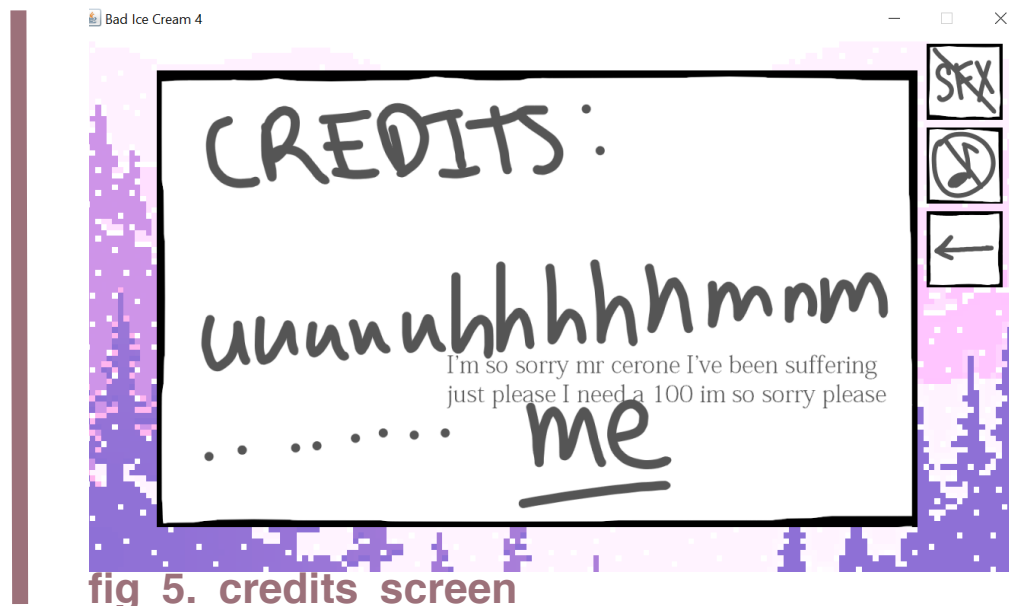


fig 5. credits screen

level selection

The Level Selection Screen contains 10 numbered buttons that take the user to the respective levels they indicate, as well as a field at the bottom for a secret code. When the user enters a code into the field and presses the SEARCH button, the program will search through its list of secret codes, looking for a match with what the user has entered. If what the user has entered is a secret code, they will be taken to a secret 11th level.

Valid secret codes are: cerone i\$ awesome, ICs\$u roXXX, or PIS Assign 1 0 0.

level selection

The **Level Selection Screen** also includes the **SFX** and **Music** buttons, as well as a back button which takes them back to the title screen



game screen

The **Game Screen** features a **timer** in the top left corner that indicates how long the player has left to finish the level.

Directly underneath the **timer**, there is a list of all the types of fruit the player needs to collect. After all of the first type of fruit has been collected, its icon will shrink and become grayscale. The next fruit on the list, if there is one, will then appear in the level and become collectable. This process repeats until all fruits have been collected and the player has won by extension. If the player touches a monster or the timer runs out, the level will immediately end.

game screen

In the top-right corner, there is an **SFX** button, Music button, and a **Back** button, which takes the user back to the **Level Selection** screen. There is also a **Replay** button, which allows the user to restart the current level.



fig 7. game screen

gameplay

Each level features a grid framework of ice blocks, fruits, and monsters on top of a unique animated GIF background. The player moves around using the **Arrow** keys. Tapping an **arrow** key briefly will not cause the player to move, but will cause the player to face that direction. Note, however, that the character will still appear to be facing towards the player. Users must keep in mind what direction they are facing.

gameplay

Players can also shoot or destroy rows of ice blocks using the **space** key. If the player is not directly facing an ice block, pressing **space** will cause a row of ice blocks to be formed. Formation will continue in the direction being faced until another ice block is reached, a monster happens to be in the way, or the level's boundaries have been reached. Any fruit caught along the way will be trapped in the ice. Fruit cannot be collected while trapped in ice.

If the player is directly facing an ice block, pressing **space** will cause a row of ice blocks to be destroyed. Destruction will continue until a non-ice tile is reached, or the level's boundaries are reached. Fruit trapped in ice will then be released and become collectable.



fig 8. space is pressed twice while player faces down

fruit

There are many different types of fruits that may appear, some of which have unique properties.

Lemons, oranges, peaches, watermelon, and grapes are all stationary fruit. They do not move.



fig 9. the stationary fruits

Strawberry move about randomly at a moderate pace.

Kiwi move about very quickly and randomly.

Chili move about randomly and melt adjacent ice blocks on all sides. They can only remain trapped in an ice block for two seconds before breaking free.

Avocado change position randomly every second.



fig 10. the dynamic fruits

monsters

There are many different types of monsters that may appear, each with its own unique set of properties.

Green Trolls move about slowly. They move in a straight line until they meet an ice block. If the troll was originally facing right, it will try to turn right. Otherwise, it will try to turn left. If that is not possible, it will try to go up. If that is still not possible, it will stay still. Similarly, the order of precedence for a troll that was originally going up is left, right, down, then no movement. That of a troll originally going left is down, up, right, then no movement. That of a troll originally going right is up, down, left, no movement.

Cows move about slightly faster, albeit completely randomly.

Eggs move about randomly, but can jump over ice blocks, though only one ice block. If there are two or more in a row, an **Egg** cannot jump.

Orange Squids move about randomly, and can destroy an adjacent ice block in the direction they're facing.

Yellow Cows always follow the player.

Blue Cows alternate between moving randomly and burrowing underground and following the player every five seconds. When they are burrowed underground, a yellow dot is used to denote their location to the player.



fig 11. monsters

endgame

When the user has won, a display in the middle of the screen indicates to the player a win. If the user is not on the last level, a right-facing Next arrow appears below which allows the user to move onto the next level. The user may also restart the level, or go back to the Level Selection Screen.

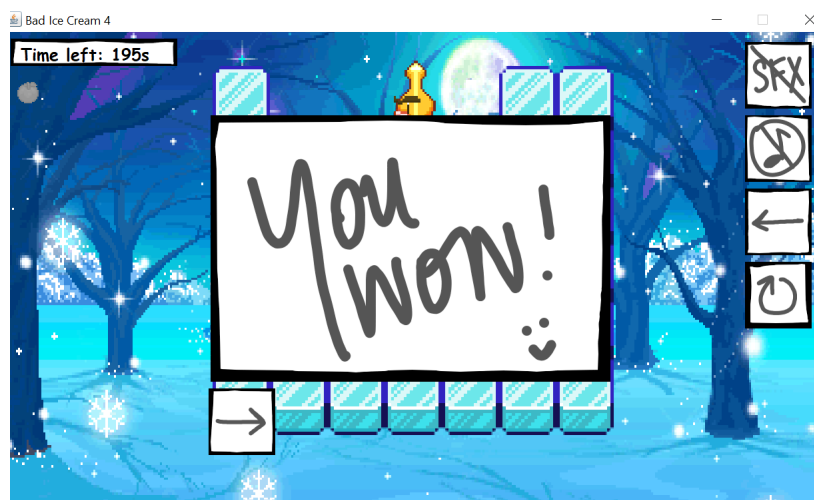


fig 12. win screen

If the user loses, a display in the middle of the screen indicates to the player a loss. The user may restart the level, or go back to the Level Selection Screen.



fig 13. loss screen

bibliography

resize image in java. (n.d.). Retrieved from <https://stackoverflow.com/questions/5895829/resizing-image-in-java>

Convert color image into black and white using java created from scratch. (n.d.). Retrieved from <https://stackoverflow.com/questions/41054525/convert-color-image-into-black-and-white-using-java-created-from-scratch>

Playing Sounds in Java. (n.d.). Retrieved from <https://www.cs.cmu.edu/~illah/CLASSDOCS/javasound.pdf>

How to Play an Audio file using Java. (n.d.). Retrieved from <https://www.geeksforgeeks.org/play-audio-file-using-java/>

Java – Multithreading. (n.d.). Retrieved from https://www.tutorialspoint.com/java/java_multithreading.htm

Multithreading in Java (n.d.). Retrieved from <https://www.javatpoint.com/multithreading-in-java>

Bad Ice Cream – Nitrome Wiki. (n.d.). Retrieved from http://nitrome.wikia.com/wiki/Bad_Ice-Cream