

Alexander Laudino

CSC 161-04

Dr. Farrett

Lab Assignment 8 - FreezeBoil

### **Pseudo-code**

This program will prompt the user to enter a temperature and then will display a list of substances that will freeze at that temperature and those that will boil at that temperature.

#### FreezeBoil

Begin

1. setTemp()
2. showFreezeBoil()

End

These are the methods and variables that the program depends on to prompt the user to enter a temperature, verify if a substance will freeze or boil, and display the results

#### FreezeBoilDriver

double temperature

double ABSOLUTE\_ZERO = -459.67 F

String[] SUBSTANCES = {"Ethyl Alcohol", "Oxygen", "Water"}, arraylist of substance names

double[][] FREEZE\_BOIL\_TEMPS = {{-173, 172}, {-362, -306}, {32, 212}}, arraylist of list of freezing and boiling temperatures for substances

boolean[] freezeBoilBool = boolean[2][3], empty array of true/false values for temperature checks

String[] freezing = String[3], empty array for names of substances that freeze at the inputted temperature

String[] boiling = String[3], empty array for names of substances that boil at the inputted temperature

Begin setTemp()

1. Initialize and declare temp variable
2. Initialize continueInput variable and set to true
3. Begin do-while loop to validate user input
4. Begin try-catch exception handling to validate input is a number
5. Print "Enter a temperature: "

6. Input temp
7. Catch InputMismatchException
8. Print "Try again. (Incorrect input: a number is required.)"
9. Discard current input line so user can enter a new line of input
10. Check if input is in range, temp >= absoluteZero
11. Set continueInput to false to trigger sentinel value and end loop
12. Else
13. Print "Please enter a number greater than " + absoluteZero
14. Continue loop
15. While continueInput = true, continue loop, if continueInput = false, end loop
16. temperature = temp

End

Begin setFreezeBoil()

1. freezeBoilBool[0][0] = isEthylFreezing()
2. freezeBoilBool[0][1] = isOxygenFreezing()
3. freezeBoilBool[0][2] = isWaterFreezing()
4. freezeBoilBool[0][0] = isEthylBoiling()
5. freezeBoilBool[0][1] = isOxygenBoiling()
6. freezeBoilBool[0][2] = isWaterBoiling()

End

Begin showFreezeBoil()

1. Set numFreezing to return value of isFreezing(), number of substances that are freezing
2. Set numBoiling to return value of isBoiling(), number of substances that are boiling
3. Call printFreeze(numFreezing) to display results for freezing substances
4. Call printBoil(numBoiling) to display results for boiling substances

End

Begin printFreeze(int n)

1. If n == 1
2. Print "At that temperature,"
3. For substance in freezing array
4. If substance == null
5. Continue
6. Else
7. Print substance + " "
8. Print "will freeze."

9. Else if n == 2
10. String[] twoSub = String[2], create empty array to hold substance names
11. Initialize counter, t = 0
12. For substance in freezing array
13. If substance == null
14. Continue
15. Else
16. twoSub[t] = substance
17. t++
18. Print "At that temperature, " + twoSub[0] + " and " + twoSub[1] + " will freeze."
19. Else if n == 3
20. Print "At that temperature, " + freezing[0] + ", " + freezing[1] + " and " + freezing[2] + " will freeze."

End

Begin printBoil(int n)

1. If n == 1
2. Print "At that temperature, "
3. For substance in boiling array
4. If substance == null
5. Continue
6. Else
7. Print substance + " "
8. Print "will boil."
9. Else if n == 2
10. String[] twoSub = String[2], create empty array to hold substance names
11. Initialize counter, t = 0
12. For substance in boiling array
13. If substance == null
14. Continue
15. Else
16. twoSub[t] = substance
17. t++
18. Print "At that temperature, " + twoSub[0] + " and " + twoSub[1] + " will boil."
19. Else if n == 3
20. Print "At that temperature, " + boiling[0] + ", " + boiling[1] + " and " + boiling[2] + " will boil."

End

Begin isFreezing()

1. Declare and initialize counter

2. For column = 0, column < freezeBoilBool[0].length, column++
3.     If freezeBoilBool[0][column] == true
4.         boiling[column] = SUBSTANCES[column]
5.         Increment multi = multi + 1
6. Return multi

End

Begin isBoiling()

7. Declare and initialize counter
8. For column = 0, column < freezeBoilBool[1].length, column++
9.     If freezeBoilBool[1][column] == true
10.         boiling[column] = SUBSTANCES[column]
11.         Increment multi = multi + 1
12. Return multi

End

Begin isEthylFreezing()

1. if temperature <= freezeBoilTemps[0][0] == "-173"
2.     return true
3. else
4.     return false

End

Begin isEthylBoiling()

1. if temperature >= freezeBoilTemps[0][1] == "172"
2.     return true
3. else
4.     return false

End

Begin isOxygenFreezing()

1. if temperature <= freezeBoilTemps[1][0] == "-362"
2.     return true
3. Else
4.     return false

End

Begin isOxygenBoiling()

1. if temperature >= freezeBoilTemps[1][1] == "-306"
2.     return true
3. else
4.     return false

End

Begin isWaterFreezing()

1. if temperature <= freezeBoilTemps[2][0] == "32"
2.     return true
3. else
4.     return false

End

Begin isWaterBoiling()

1. if temperature >= freezeBoilTemps[2][1] == "212"
2.     return true
3. else
4.     return false

End

## UML

