# COW – Conditional Branching

## Sample Code

**Level 1**

**Create a class called Goldilocks that has the following methods:**

Name: doYouEnterHouse
Input: boolean doorLocked
Output: boolean enterHouse
Action: Returns whether Goldilocks entered the house. She enters the house if the door is unlocked.

---

Name: howsTheBed
Input: int stiffness
Output: boolean justRight
Action: Returns whether the stiffness of the bed is just right. The method takes in an int variable stiffness that will store how stiff the bed is. Positive numbers indicate that the bed is stiff. Negative numbers indicate that the bed is soft. To be just right, a bed must be neither stiff nor soft (stiffness is zero).

---

Name: howsTheChair
Input: String size
Output: boolean justRight
Action: Returns whether the size of the chair is just right. The method takes in a String variable size that will store a "Small", "Medium", or "Large". To be just right, a chair must be medium.

**Level 2**

Name: howsThePorridge
Input: double temperature
Output: boolean justRight
Action: Returns whether the porridge is the right temperature. To be the right temperature, porridge needs to be between 75 and 99 inclusive.

## Level 3

Name: howsEverything
Input: double temperature, String size, int stiffness
Output: boolean justRight
Action: Returns whether everything is justRight. To be the right temperature, porridge needs to be between 75 and 99 inclusive. To be just right, a chair must be medium ('M'). To be just right, a bed needs to be neither stiff nor soft (0).

## Level 4

Name: whichChairToSitIn
Input: String size1, String size2, String size3
Output: int chair
Action: Returns which chair Goldilocks should sit in (1, 2, or 3). Goldilocks would prefer to sit in the first chair that is a medium chair ("Medium). But if there are no medium chairs then Goldilocks would prefer to sit in the first large chair ("Large"). If there are no medium or large chairs then Goldilocks will sit on the ground. If goldilocks sits on the ground, indicate this by returning a -1.

## Level 5

Name: whichBedToSleepIn
Input: int stiffness1, int stiffness2, int stiffness3
Output: int bed
Action: Returns which bed Goldilocks should sleep in (1, 2, or 3). Goldilocks would prefer to sleep in a bed that is just right (stiffness is 0), which ever just right bed comes first. If none are just right, GoldiLocks would prefer to sleep in which ever one is closer to just right. The one exception to this is that Goldilocks would always prefer to sleep in a soft bed (negative stiffness) as opposed to a stiff bed (positive stiffness), no matter how soft.

## Level 6

Name: whichPorridgeToEat
Input: double temp1, double temp2, double temp3
Output: int porridge
Action: Returns which porridge Goldilocks should eat (1, 2, or 3). In order for porridge to be acceptable, it needs to be between 75 and 99 inclusive. 90 is the best temperature though. Return which temperature (1, 2, or 3) comes closest to 90 degrees. If no temperature falls within the acceptable range, then return a -1. If two porridges tie then return the lower of the two numbers.

**----------End of Sample Code----------**

# Level 1

**Create a class called Grader that has the following method:**

Name: isItPassing
Input: double percentGrade
Output: boolean passing
Action: takes in a percentage grade and returns whether the student is passing. Anything at or above 60 is passing.

---

**Create a class called Cop that has the following method:**

Name: areTheySpeeding
Input: double speed, double speedLimit
Output: boolean speeding
Action: Returns whether they are speeding. Any speed above the limit indicates speeding

---

**Create a class called NumberAnalyzer that has the following method:**

Name: isItEven
Input: int number
Output: boolean isEven
Action: Returns whether the number sent in is even. Hint – use the mod (%) operator.

---

**Create a class called PayRoll that has the following method:**

Name: getBonus
Input: double sales
Output: double bonus
Action: Returns the amount the employee gets as a bonus. If sales are below 300,000 then they do not get a bonus. If their sales is above 300,000 then receive 10% of whatever amount is above 300,000. For example, sales of 500,000 would result in a bonus of 20,000 since they their sales went 200,000 above 300,000 and 10% of 200,000 is 20,000.

---

**Create a class called Hogwarts that has the following method:**

Name: isCauldronAtCorrectTemp
Input: double temp
Output: boolean isNormal
Action: Returns whether the cauldron is at the correct temperature. A cauldron should be at exactly at 200.23 degrees.

# Level 2

**Add the following method in Grader:**

Name: isError
Input: double percentGrade
Output: boolean isInError
Action: takes in a percentage that a student has in a class and returns whether the grade is an error. Any grade below 0 or above 100 is in error.

---

**Add the following method in Cop:**

Name: isItRecklessDriving
Input: double speed, double speedLimit, boolean isOnHighway
Output: boolean isReckless
Action: Returns whether the ticket is reckless driving. It is reckless driving if the speed is 20 mph above the speedLimit on a highway or 10 mph on a road that is not a highway.

---

**Add the following method in NumberAnalyzer:**

Name: isItASingleDigit
Input: int number
Output: boolean isSingleDigit
Action: Returns whether the number passed in only consists of a single digit.

---

**Add the following method in Hogwarts:**

Name: isGoodProfessor
Input: String teacherName
Output: boolean goodProfessor
Action: takes in the name of a professor at Hogwarts and returns whether they are a good professor. The good professors at Hogwarts include “Dumbledore”, “Flitwick”, “McGonagall”, “Sprout”, “Lupin”, and “Hagrid”.

---

**Add the following method in Calendar:**

Name: isSchoolOpen
Input: boolean isWeekDay, boolean isHoliday, boolean timeForVacation
Output: boolean schoolOpen
Action: Returns whether the school is open. The school is open during weekdays unless it is a holiday, or it is time for vacation.

# Level 3

**Add the following method in Grader:**

Name: getLetter
Input: double percentGrade
Output: String letterGrade
Action: takes in a percentage that a student has in a class and returns the corresponding letter Grade that the student should receive for the class using the following scale:
[0 – 60) F, [60 – 70) D, [70 – 80) C, [80 – 90) B, [90 – 100] A
You can assume that percentGrade will always fall between 0 and 100.

---

**Add the following method in Cop:**

Name: whatsTheFine
Input: double speed, double speedLimit, boolean isOnHighway
Output: double cost
Action: Returns how much the speeding ticket will be. If the driver was not speeding then a 0 is returned. If the drive was driving over the limit but not recklessly then the fine will be 120 plus 10 for each mph over the limit). If the driver was driving recklessly then the fine will be 5000. You should make use of isItRecklessDriving and areTheySpeeding to program this method.

---

**Add the following method in PayRoll:**

Name: whatIsMyBaseSalary
Input: String degree
Output: double salary
Action: Returns the base salary of the employee given the following chart:

| | |
|---|---|
| Doctorate | 100000 |
| Masters | 60000 |
| Bachelors | 40000 |
| Associates | 25000 |
| No Degree | 15000 |

**Add the following method in Hogwarts:**

Name: whatHogwartsHouse
Input: int dedication, int intelligence, int bravery, int ambition
Output: String house
Action: Returns the Hogwarts House that the student will get into given ratings in four categories. A person is placed in a house depending on which of their categories are the highest. Here are the houses along with their corresponding category:
"Gryffindor" – bravery
"Hufflepuff" – dedication
"Slytherin" – ambition
"Ravenclaw" – intelligence
You are guaranteed that no two categories will be equal.

---

**Add the following methods in Calendar:**

Name: getNumberOfMonth
Input: String month
Output: int numberOfMonth
Action: Returns the number of the month. For example, "January" returns 1 and "May" returns 5.

# Level 4

**Add the following methods in NumberAnalyzer:**

Name: whatTypeOfNumberIsIt
Input: int number
Output: String description
Action: Returns a description of the number passed in. The description returned should be "even positive", "odd positive", "even negative", "odd negative", or "zero".

---

**Add the following method in PayRoll:**

Name: whatIsMySalary
Input: String degree, int yearsExperience, double sales
Output: double salary
Action: Returns the salary of the employee. The salary should be equal to the base pay plus their bonus plus 5% of the base pay for each year of experience that the employee has. So, an employee with a "Bachelors" with 5 years of experience and 500,000 in sales should have a salary of 70,000 (70,000 = 40,000 + 2,000 * 5 + 20,000). You should make use of the whatIsMyBaseSalary and getBonus methods to write this method.

---

**Add the following method in Calendar:**

Name: whatSeason
Input: String month, int day
Output: String season
Action: Returns what season it is (winter, spring, summer, or fall). Use the following for the start of the seasons:
spring - March 19, summer - June 20, fall - September 22, winter - December 21

**Add the following methods to Hogwarts:**

Name: whatDoYouTurnInto
Input: int doorNumber, String drinkColor
Output: String animal
Action: takes in the door the user went through and what color potion you drank and returns what animal you were polymorphed into. The doorNumber is going to be either 1, 2, or 3. The colors are going to be either "pink", "green", or "blue". The resulting animal can be found in the following table:

| Door Number | Drink Color | Ending |
|---|---|---|
| 1 | pink | "Cat" |
| | green | "Toad" |
| | blue | "Owl" |
| 2 | pink | "Rat" |
| | green | "Snake" |
| | blue | "Griffin" |
| 3 | pink | "Unicorn" |
| | green | "Dragon" |
| | blue | "Giant Squid" |

Name: whereToApparate
Input: boolean overseas, boolean onCoast, boolean inBigCity,
Output: String destination
Action: You are looking for someone and are given clues about where they have gone. Uses the boolean value clues passed in to return where to apparate to using the following options: (London, Honk Kong, Nassau, Lucerne, Chicago, Los Angeles, Nags Head, Blacksburg)

# Level 5

**Add the following method in Grader:**

Name: getModifier
Input: double percentGrade
Output: String modifier
Action: takes in a percentage that a student has in a class and returns the modifier that should be appended to the letter. Any percent that ends in a 7, 8, or 9 should return a "+" unless it is under 60. Any percent that ends in a 0, 1, or 2 should return a "–" unless it is less than 60 or equal to 100. A 100 should return a "+". Anything else should return an empty String "".

---

**Add the following method in NumberAnalyzer:**

Name: howManyAreTheSame
Input: int num1, int num2, int num3, int num4
Output: int numSame
Action: Returns how many numbers are the same (0, 2, or 3, 4). If there are two sets of pairs then it returns only a 2.

---

**Adjust the following method in PayRoll:**

Name: taxAmount
Input: double money
Output: double - the amount of tax that the person has to pay on their income
Action: Takes in the person's income and returns the amount that they get charged in taxes. Do not assume that they get charged the same tax rate on all of their income. Tax rates should follow a step wise pattern as they do in real life.
Use the 2014 tax brackets for a single filer.

| | |
|---|---|
| 10% | $0 to $9,075 |
| 15% | $9,076 to $36,900 |
| 25% | $36,901 to $89,350 |
| 28% | $89,351 to $186,350 |
| 33% | $186,351to $405,100 |
| 35% | $405,101 to $406,750 |
| 39.60% | $406,751+ |

**Add the following method in Calendar:**

Name: whatZodiacSign
Input: String month, int day
Output: String zodiacSign
Action: Returns what the zodiac sign is for the given month and day. Use the following chart to determine what zodiac sign it is. Do not use one you found on the Internet.

Aquarius: January 20 - February 18
Pisces: February 19 - March 20
Aries: March 21 - April 19
Taurus: April 20 - May 20
Gemini: May 21 - June 20
Cancer: June 21 - July 22
Leo: July 23 - August 22
Virgo: August 23 - September 22
Libra: September 23 - October 22
Scorpio: October 23 - November 21
Sagittarius: November 22 - December 21
Capricorn: December 22 - January 19

---

**Add the following method in Hogwarts:**

Name: whatToPrepare
Input: String day, double timeOfDay, boolean isVegetarian
Output: String food
Action: This method tells the House Elves of Hogwarts what food to prepare. It takes in the day of the week ("Monday", "Tuesday', "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"), time of day (0 - 24), and whether they are a vegetarian. Any time during the morning (before 11) the answer is "cereal" on a weekday and "biscuits with gravy" on the weekend. During the remainder of the day, the answer is "chicken with rice" on weekdays and "steak and potatoes" on the weekend. If they are a vegetarian, replace "biscuits with gravy" with "waffles", "chicken with rice" with "beans with rice", and "steak and potatoes" with "salad".

# Level 6

**Add the following methods in Grader:**

Name: adjustForHonors
Input: double percentGrade
Output: double adjustedPercent
Action: takes in a percentage that a student has in a class and returns an adjusted score. The percent returned should be 10 percent higher but capped at 100. For example, with isHonors true, an 88 becomes a 98 and a 92 becomes a 100. If the percentGrade is less than 60 then the grade does not change so a 48 stays at a 48.

Name: adjustForCurve
Input: double percentGrade
Output: double adjustedPercent
Action: takes in a percentage score. It returns an adjusted score so that the number of points from 100 is multiplied by .75. So a 60% would become a 70% (-40 * .75 = -30) and a 75% would become an 81.25% (-25 * .75 = -18.75).

Name: giveLetterGrade
Input: double percentGrade, boolean isHonors, boolean isPassFail, boolean isCurved
Output: String letterGrade
Action: takes in a percentage that a student has in a class and returns the corresponding letter Grade that the student should receive for the class. For any percentGrade that isError indicates is in error should return "Error". Any pass/fail class should return a "P" or an "F" depending on whether or not ***isItPassing*** says it is passing. Otherwise modify it first by using ***adjustForCurve***. Then modify it using ***adjustForHonors***. Then return the combination of the two strings returned from ***getLetter*** and ***getModifier***.

---

**Add the following method in NumberAnalyzer:**

Name: getMostCommonNumber
Input: int num1, int num2, int num3, int num4, int num5
Output: int commonNumber
Action: Returns the number that shows up the most. If there is a tie, then return the average of the most common numbers.

**Add the following method in PayRoll:**

Name: whatIsMySalaryAfterTax
Input: String degree, int yearsExperience, double sales
Output: double salary
Action: Use the whatIsMySalary method and taxAmount method to determine what the employees salary is after tax.

---

**Add the following method in Calendar:**

Name: isALeapYear
Input: int year
Output: boolean isALeapYear
Action: Returns whether the given year is a leap year. Note – leap years are not always every four year.