

## ITEC 120 Lab 14

Created by Dr. Ray

You will need to turn in a lab report for this lab. These are two of the easy problems from the local programming competition. You should consider participating next year, you could win \$100!

### **Problem 1: Golf**

The game of golf is mysterious in many ways. First, it is hard to understand what is so nice about hitting a ball over 200 yards, and then going to look for it. Second, instead of counting in numbers, they use terms like “par”, “birdie” (1 below par) and “bogey” (1 over par), and in rare cases “hole-in-one”, “eagle” (2 below par), “double eagle” (3 below par), “double bogey” (2 over par), and “triple bogey” (3 above par).

John has just joined the golf club and he is new to the names of the scores. He has just completed his first round at the course, and he has filled in his score card, including the 9 that he scored on the par-4 hole 3 after his ball went swimming. Now he needs to be told what each of his scores is called. He also needs to be told that at this course, no one ever reports a score worse than triple bogey, and so he can simply report triple bogey for his 5 over par on hole 3.

Your job, as John's caddy, is to write a program, which, given the par for a hole and the number of strokes that John wrote down, tells John the term to report for his score. Please make sure that if John gets a hole-in-one, it is reported with that name and nothing else.

### **Input**

The first line of input specifies the number  $N$  of holes John played. Each of the next  $N$  lines contains two integers,  $P$  and  $S$ , separated by a single space.  $P$  is the par for the hole and  $S$  is the score John made.  $P$  is either 3, 4, or 5, and  $S$  is any positive integer less than 20). All input will be typed into your program in a command prompt.

### **Output**

For each hole, print on a single line the applicable score, as specified by the above explanation: “Hole-in-one.”, “Double eagle.”, “Eagle.”, “Birdie.”, “Par.”, “Bogey.”, “Double bogey.”, or “Triple bogey.”.

### **Example**

<b>Given this input:</b>	<b>The output would be:</b>
4	Eagle.
5 3	Double bogey.
3 5	Triple bogey.
4 9	Par.
5 5	

## Problem 2: Password Security

Password security is a tricky thing. Users prefer simple passwords that are easy to remember (like *buddy*), but such passwords are often insecure. Some sites use random computer-generated passwords (like *xvtpzyo*), but users have a hard time remembering them and sometimes leave them written on notes stuck to their computer. One potential solution is to generate "pronounceable" passwords that are relatively secure but still easy to remember.

FnordCom is developing such a password generator. You work in the quality control department, and it's your job to test the generator and make sure that the passwords are acceptable. To be acceptable, a password must satisfy these three rules:

- It must contain at least one vowel.
- It cannot contain three consecutive vowels or three consecutive consonants.
- It cannot contain two consecutive occurrences of the same letter, except for 'ee' or 'oo'.

(For the purposes of this problem, the vowels are 'a', 'e', 'i', 'o', and 'u'; all other letters are consonants.) Note that these rules are not perfect; there are many common/pronounceable words that are not acceptable.

### Input

The input consists of one or more potential passwords, one per line, followed by a line containing only the word 'end' that signals the end of the file. Each password is at least one and at most twenty letters long and consists only of lowercase letters.

### Output

For each password, output whether or not it is acceptable, using the precise format shown in the example.

### Example

The following input:	Will produce the following output:
a	<a> is acceptable.
tv	<tv> is not acceptable.
ptoui	<ptoui> is not acceptable.
bontres	<bontres> is not acceptable.
zoggax	<zoggax> is not acceptable.
wiinq	<wiinq> is not acceptable.
eep	<eep> is acceptable.
houctuh	<houctuh> is acceptable.
end	

**Survey questions:**

On a scale of 1 to 10 with 1 being complete disagreement, 5 being neutral, and 10 being strong agreement answer the following questions. *Note:* Every member of your group must answer these questions. Your answers will not be used to change your grades, and will help me and other professors improve ITEC 120 for the next group of students.

I was able to complete the homework assignments without extreme difficulty.

I enjoyed ITEC 120.

I learned a considerable amount of information in ITEC 120.

I enjoyed the practical application weeks (Robots, Sound, Pictures, GUIs).

The Lego Robots helped me learn java programming.

I would recommend using the Lego Robots again next semester.

I would prefer fewer, longer homework assignments in ITEC 120.

The labs helped me learn Java more than the lectures.

The labs were too long and required too much out of class effort.

The lectures helped me learn the material.

**Short answer questions:**

What was the best and worst part of using Lego Robots?

What, in this class, hindered you from learning Java?

Per week, how many hours (on average) do you spend out of class working on homework?

Per week, how many hours (on average) do you spend out of class working on lab work?

Per week, how many hours (on average) do you spend studying for ITEC 120?

Other feedback you want to give about ITEC 120.

What would you tell someone who would take ITEC 120 next semester?

What changes would you make in the course?

What were the best and worst parts of ITEC 120?