



# South Central USA Regional Programming Contest



*Choose Your Own Adventure*

## Introduction:

After reading the book *Tim and Marc Kill Kenny* about fifty zillion times, James decided he'd had it with choose-your-own-adventure stories. No matter what choices he made, it seemed like Kenny always fell down an abandoned mine shaft, got run over by a bus load of nuns, or was messily devoured by stray cats. James eventually found the page with the happy ending (where Kenny saves himself by trapping Tim and Marc between the pizza and the hungry programmers) by flipping through the book, but he can't figure out how to get there by following the rules. Luckily, he owns a C compiler...

## Input:

Input to this problem will consist of a (non-empty) series of up to 100 data sets, each representing a choose-your-own-adventure story. Each data set will be formatted according to the following description, and there will be **no blank lines** separating data sets.

The first line contains a single integer  $n$  indicating the number of data sets.

A single data set has 2 components:

1. *Page Count* - A line containing a single integer  $X$ , where  $1 < X < 100$ , indicating the number of pages in the story.
2. *Page List* - A sequence of  $X$  lines, each of which represents a page from the book. Each line has the following components separated from one another by single spaces:
  - Line type* - A single character indicating what type of line this is. It will represent either a "C" choice page, or an "E" end page. Page 1 is always a choice page.
  - Text* - A string of text surrounded by double quotes. Including the quotes, this component will not exceed 256 characters. The quotes are given for input purposes only and should not be considered part of the text. The text will not contain embedded double quotes.
  - Choices* - Two positive integers from 1 to  $X$  indicating the pages where the reader can go from this page. Only choice pages have this component.
  - Ending Type* - Either the text "HAPPY" or "GRISLY". There will only be one happy ending per story, and only end pages have this component.

## Output:

For each story in the input:

1. Output a single line, "STORY #" where # is 1 for the first story, 2 for the second story, etc.
2. Determine the story that begins on page 1 and ends on the happy ending page. Output the text of this story, printing one "page" of text per line. Note that there is only one such story for each data set.

## Sample Input:

```
2
3
C "Arrived at LSU for the contest" 2 3
E "Was devoured by sidewalk ants" GRISLY
E "Won the contest. Received glory and nachos." HAPPY
5
C "Saw a peanut" 3 5
E "Made peanut butter sandwich" HAPPY
C "Found a hammer" 4 2
E "Hit self on head with hammer, ouch!" GRISLY
E "Ate the peanut, choked on it, and died" GRISLY
```

## Sample Output:

```
STORY 1
Arrived at LSU for the contest
Won the contest. Received glory and nachos.
STORY 2
Saw a peanut
Found a hammer
Made peanut butter sandwich
```

The statements and opinions included in these pages are those of *Hosts of the South Central USA Regional Programming Contest* only. Any statements and opinions included in these pages are not those of *Louisiana State University* or the *LSU* Board of Supervisors.

© 1999, 2000, 2001, 2002, 2003, 2004 [Isaac Traxler](#)