New Mexico State University
Programming Competition
Spring 2024

Problem Set

Problem 1: Max Alphabetizing

Point Value: 2 Points

Bounty: $30

Description:
Given a collection of words, sort them so that the words with the greatest characters appear first. For example, zoo would appear before apple, because the greatest character of zoo, i.e. 'z', is greater than the greatest character of apple, which is 'p'. If two words have the same greatest character, sort them by their second greatest character, third greatest, and so on. Note the second greatest character for a word may be the same as the greatest if there are duplicates. All words will be given only in lower case.

Sample Input 1:
```
can you take me to the zoo to see some animals
```

Sample Output 1:
```
zoo you to to take the some animals see can me
```

Problem 2: Base Conversion

Point Value: 2 Points

Bounty: $30

Description:
Given a positive integer `i`, in base `s`, output it in base `o`. Input will be given as three values `i`, `s`, and `o` separated by spaces. Bases `s` and `o` will be inclusively between 2 and 20, and will be represented in decimal (base 10). Digits greater than 9, should be represented as capitalized alphabetic characters, i.e., 'A', 'B', 'C', etc.. All numbers will be less than 2^16.

Sample Input 1:
```
11101 2 16
```

Sample Output 1:
```
1D
```

Sample Input 2:
```
666 7 14
```

Sample Output 2:
```
1A6
```

Problem 3: Caesar Ciphers

Point Value: 3 Points

Bounty: $30

Description:
Caesar ciphers are an encryption technique that relies on substituting characters of the plaintext with characters that are shifted by some number. For example, a Caesar cipher that shifts by four would replace 'A' with 'E', 'B' with 'F', 'C' with 'G', etc.

Unfortunately, Caesar Ciphers can be easily broken by examining the frequency of the characters appearing in the ciphertext. We know that in the English language that the letter 'E' is the most common letter to appear in most phrases. Given a text encrypted with a Caesar cipher, output its plaintext. You may assume that the letter 'E' will be the most common letter in the plaintext.

Input will be given as a string of capital characters between 'A' and 'Z'. No spaces, numbers, or other non-alphabetic characters will be given.

Sample Input 1:
```
NBLYYVYYMEYYJNLYYMQYYNM
```

Sample Output 1:
```
THREEBEESKEEPTREESWEETS
```

Sample Input 2:
```
FVRMRGURZRNAFBSZRZRF
```

Sample Output 2:
```
SIEZETHEMEANSOFMEMES
```

Problem 4: Igpay Atinlay

Point Value: 3 Points

Bounty: $60

Description:
Pig Latin is a language game, which alters English words by adding a suffix to end of the word and rearranging it. For this problem we will assume the following rules for pig latin:

1. If a word begins with a consonant, move the beginning consonant cluster, i.e. all consonants up until the first vowel, to the end of the word. Then add "ay" to the end of the new word.
2. If a word begins with a vowel, i.e., 'a', 'e', 'i', 'o', or 'u', add "way" to the end of the word.

Given a series of words, translate them to pig latin. All words will be lower case and will be separated by spaces.

Sample Input 1:
```
to be or not to be that is the question
```

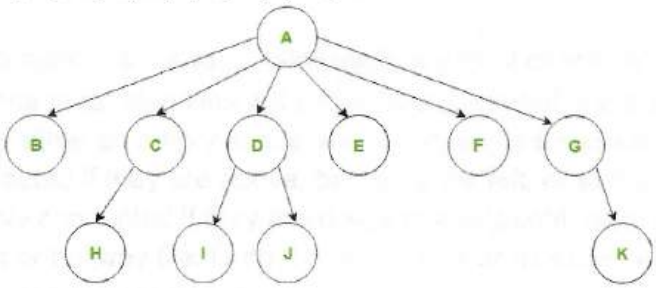
Sample Output 1:
```
otay ebay orway otnay otay ebay atthay isway ethay uestionqay
```

Problem 5: N-ary Preorder

Point Value: 5 Points

Bounty: $60

Description:
N-ary trees are trees in which any particular tree node may have an arbitrary number of children. Given a n-ary tree output its pre-order traversal. Remember that a pre-order traversal is defined as the recursive traversal of a tree where the root node's value is visited followed by the recursive visiting of each of the node's children in order. For example, the preorder traversal of the following tree is A, B, C, H, D, I, J, E, F, G, K.



Input will be given as n, the number of nodes in the tree, followed by n lines of input, one line for each node in the tree. Each line of input for a node will first give the value for the node, followed by the values for each of its children nodes. Note that leaf nodes will have no children. Node values will be given as integers. The first node given will always be the root node. Your program should output the preorder traversal of the tree, visiting the child nodes of each node in the order they are given.

Sample Input 1:
```
3
1 2 3
2
3
```

Sample Output 1:
```
1 2 3
```

Sample Input 2:
```
10
9 7 6 8
7 1 2
6 10 11 13
8 4
1
2
10
11
13
4
```

Sample Output 2:
```
9 7 1 2 6 10 11 13 8 4
```

Problem 6: Connected Rooms and Halls

Point Value: 5 Points

Bounty: $60

Description:
Matthew is developing an algorithm to produce maps for his video game. The maps generated by the algorithm are a collection of tiles on a grid. However, the generation algorithm sometimes creates maps that are not completely connected, meaning there might be hallways or rooms that are not connected to any other. These maps are unusable because there is no way for players to gain access to all of the created spaces.

Input will consist of two numbers `r` and `c`, followed by a grid of characters with `r` rows and `c` columns representing the map. Map tiles will either be a '.' (period character) or 'x' (lowercase x). A period represents either an empty space. An x represents a room or hallway space. Room and hallways are connected if they are above, below, to the left, or to the right of each other. They are not necessarily connected if they are diagonally adjacent. Note that every map will have at least one room or hallway tile. Output `connected` or `disconnected` depending on whether or not the map is fully connected.

Sample Input 1:
```
5 5
.....
.x...
.x.xx
.x.xx
.....
```

Sample Output 1:
```
disconnected
```

Sample Input 2:
```
4 7
.xxxxx.
.x...x.
.x.xxx.
.x...xx
```

Sample Output 2:
```
connected
```

Problem 7: Plane Seating

Point Value: 7 Points

Bounty: $60

Description:
Two bitter rival countries' soccer teams have accidentally booked the same flight to the Olympic games. Additionally, there are several reporters and team support staff on the flight. The flight is totally full, and the flight attendants want to rearrange seating to avoid any incidents mid-flight. Figure out whether or not it is possible to rearrange seating on the flight to satisfy the following rules. Output either 1 or -1 to indicate whether it is possible.

We will use the following notation:
R: a reporter
A: a player from team A
B: a player from team B
S: a team staff member

The following rules apply:

1. Sitting "next to" someone in this challenge does not include seats across the aisle.
2. Players from team A cannot sit directly next to players from team B
3. Team staff can sit next to anyone.
4. Reporters can sit next to any players, but must not sit between players of opposite teams
5. Players from team B do not like each other and must not sit next to one another
6. Reporters cannot sit next to each other, or they will talk too much

The airplane's current seating will be specified as follows:
The airplane will have 2 columns of seats, in 10 or fewer rows. Each column will have 2 or 3 seats per row. An efficient algorithm will pass all our test cases within a minute.

Sample Input 1:
```
3
AA BB
AA BR
BA SS
```

Sample Input 2:
```
3
ABA BRR
AAA BSS
SSS ABB
```

Sample Input 3:
```
4
RR RRR
AB BBB
AA AAA
SS SSS
```

Sample Output 1:
```
-1
```

Sample Output 2:
```
1
```

Sample Output 3:
```
1
```

Problem 8: Advanced Plane Seating

Point Value: 9 Points

Bounty: $120

Description:
Two bitter rival countries' soccer teams have accidentally booked the same flight to the olympic games. Additionally, there are several reporters and team support staff on the flight. The flight is totally full, and the flight attendants want to rearrange seating to avoid any incidents mid-flight. Return the minimum number of seat swaps required to satisfy the 5 rules below. If the task is impossible, return -1.

We will use the following notation:
R: a reporter
A: a player from team A
B: a player from team B
S: a team staff member

The following rules apply:

1. Sitting "next to" someone in this challenge does not include seats across the aisle.
2. Players from team A cannot sit directly next to players from team B
3. Team staff can sit next to anyone.
4. Reporters can sit next to any players, but must not sit between players of opposite teams
5. Players from team B do not like each other and must not sit next to one another
6. Reporters cannot sit next to each other, or they will talk too much

The airplane's current seating will be specified as follows:
The airplane will have 2 columns of seats, in 10 or fewer rows. Each column will have 2 or 3 seats per row. An efficient algorithm will pass all our test cases within a minute.

Sample Input 1:
```
3
AA BB
AA BR
BA SS
```

Sample Output 1:
```
-1
```

Sample Input 2:
```
3
ABA BRR
AAA BSS
SSS ABB
```

Sample Output 2:
```
2
```

Sample Input 3:
```
4
RR RRR
AB BBB
AA AAA
SS SSS
```

Sample Output 3:
```
2
```

Sample Input 4:
```
5
RRR RR
SSS SS
AAA AA
BBB BB
RRR RR
```

Sample Output 4:
```
4
```