ECOO 2012
Programming Contest
Solutions & Notes

Regional Competition (Round 2)

April 28, 2012

Problem 1: Prime Time

Recommended Approach

Every number is a product of two numbers: The message code number in the range 0 to 3030, and the prime number key in the range 100000 to 500000. The easiest thing to do is simply search one of the message numbers sequentially for the first factor you find starting at 100000.

It may also be possible in some cases to find the GCF (greatest common factor) of two of the message numbers using the Euclidean algorithm or some other method. But this method may not find the right key. For example if the two message numbers you select are both even, the GCF will actually be double the key.

Solution to DATA11.txt

```
REMEMBER WHEN YOU WERE YOUNG? YOU SHONE LIKE THE SUN...

NOW THERES A LOOK IN YOUR EYE, LIKE BLACK HOLES IN THE SKY.

SHINE ON YOU CRAZY DIAMOND!

YOU WERE CAUGHT IN THE CROSSFIRE OF CHILDHOOD AND STARDOM, BLOWN ON
THE STEEL BREEZE.

COME ON YOU RAVER, YOU SEER OF VISIONS. COME ON YOU PAINTER, YOU
PIPER, YOU PRISM AND SHINE!
```

Solution to DATA12.txt

```
ZIGGY PLAYED GUITAR, JAMMING GOOD WITH WIERD AND GILLY AND THE SPIDERS
FROM MARS.

HE PLAYED IT LEFT HAND, BUT MADE IT TOO FAR...

BECAME THE SPECIAL MAN! THEN WE WERE ZIGGYS BAND.

ZIGGY PLAYED FOR TIME, JIVING US THAT WE WERE VOODOO. THE KIDS WERE
JUST CRASS!

WHEN THE KIDS HAD KILLED THE MAN I HAD TO BREAK UP THE BAND...
```

Problem 2: Password Strength

Notes

There are no hidden traps here, but very few test cases are given so you have to interpret the criteria and test your solution carefully before submitting. The hardest part is probably finding the longest sequences. This can be done by looping through the string, keeping track of the last character seen, and keeping count of the current and longest sequence seen so far.

Solution to DATA21.txt

```
Very Weak (score = 4)
Very Weak (score = 3)
Good (score = 44)
Good (score = 57)
Weak (score = 33)
Very Strong (score = 93)
Strong (score = 78)
Very Strong (score = 100)
Very Weak (score = 15)
Strong (score = 80)
```

Solution to DATA22.txt

```
Very Weak (score = 14)
Good (score = 41)
Very Weak (score = 5)
Very Strong (score = 92)
Very Strong (score = 100)
Very Weak (score = 3)
Very Weak (score = 4)
Strong (score = 75)
Weak (score = 28)
Very Weak (score = 20)
```

Problem 3: Airport Radar

Recommended Approach

Compute the end point of the flight using trigonometry functions ($x = r \cdot cos(\theta),\ y = r \cdot sin(\theta)$). Then compute equation of the line of flight path ($y = mx$, where the slope is the tangent of the angle of flight). Then, for each tower:

1. Use length of line segment formula to check if the origin or end point of the flight is within radar range. If so, count the tower.
2. If not, compute the equation of the line perpendicular to the flight path passing through the radar tower point, then compute the intersection point. If it is on the line segment (between the start and end point), compute the distance to see if it's in radar range. If it is, count the tower.

This procedure may have special cases for horizontal and vertical lines, depending on the programming environment.

Simulation Approach

Compute the end point of the flight as above, then get the slope of the plane and divide rise and run by a big number to get small x and y increments for simulation. Simulate the flight of the plane along its path, checking each tower that has not seen the plane yet at each step to see if it can see it at this point (using length of a line segment formula). Count all the towers that see the plane at some point during the simulation.

Solution to DATA31.txt

```
The jet will appear on 15 radar screens.
The jet will appear on 9 radar screens.
The jet will appear on 2 radar screens.
The jet will appear on 5 radar screens.
The jet will appear on 1 radar screens.
```

Solution to DATA32.txt

```
The jet will appear on 1 radar screens.
The jet will appear on 8 radar screens.
The jet will appear on 8 radar screens.
The jet will appear on 4 radar screens.
The jet will appear on 26 radar screens.
```

Problem 4: Lo Shudoku

Exhaustive Search

An exhaustive search of all possible moves would be difficult to write and will not finish in time, so another strategy is needed. Perhaps a breadth first search would work in some cases, but still in the worst case would still have a bad running time and memory requirements. As a ball park estimate, in the worst case you have to consider 8 swaps for each square, giving you 8^9 combinations (more than 130 million) to try.

Observations

If you can find a fast enough way to transform a 3x3 square into one of the magic squares, you can just run it 8 times – once for each of the eight possible magic squares – and look for the smallest required number of moves.

There is an upper bound of 9 moves for transforming any 3x3 square that has no blanks into any given magic square. This is because every swap can be used to put at least one number in its correct spot. In fact, there is no faster approach when there are no blanks than to simply cycle through all nine spots and swap the correct piece into place if necessary.

It actually does not matter to the number of moves required whether you fill in the blanks at the beginning or at the end.

Final Strategy

Try each of the 8 magic squares. For each one, you transform the given 3x3 square by first swapping into place as many pieces as you can, and then by filling in the blanks at the end with their correct numbers. Whichever of the 8 squares requires the least moves gives you the answer.

Solution to DATA41.txt

```
446         999         888         485         000
554         999         888         769         080
553         999         888         564         000
---         ---         ---         ---         ---
```

Solution to DATA42.txt

```
188         565         000         988         544
797         336         078         988         564
882         554         778         878         444
---         ---         ---         ---         ---
```