MIDI Olympiad in Informatics 2007
Vilnius University, Lithuania

Problems explained

By Daumilas Ardickas and Adomas Paltanavičius.

Full problem statements, contest standings and reference solutions available at http://adomas.org/midi2007/

A (Divisors)

Number $n = {p_1}^{a_1} \times {p_2}^{a_2} \times \cdots$ (where p_i are primes) has $(a_1 + 1) \times (a_2 + 1) \times \cdots$ divisors. If we are seeking the *n* with *d* divisors, we must minimize p_1, p_2, ... so we pick the first primes (2, 3, 5 etc.) and factorize *d* in every possible way. We assign the factors to$(a_1 + 1)$, $(a_2 + 1)$, ... in descending order.

B (Personal code)

The task was pretty straight-forward. However, there were some pitfalls that should have been considered:

- Initial digit can't be zero.
- Month must be between 1 and 12.
- Number of days in month must be valid. Account for leap year.
- Birth date must be no later than April 28 2007.

C (UTF-8)

This task was quite easy as well. Some things that required attention:

- Unexpected end of file.
- Invalid starting byte, i.e. not matching one of given four cases.
- Overlong sequences. In UTF-8, every Unicode character can in theory be encoded in several ways. However, given there is unique valid representation, one must check for overlong sequences, e.g. 11110**000** 10**000000** 10**000000** 10**000000** whereas U+0000 should be encoded as 0**0000000**. Bugs of this type have lead to many exploits in internet software some time ago.

D (Scissors)

To win a game, we have to cut the rectangle into 1×1, or to chop in odd number of moves, so we have to cut a rectangle into two rectangles such that:

- Other player can't chop them in odd number of moves, or
- He can chop one of them in odd number of moves, but the maximal odd number of moves he can achieve is lesser than minimal number of moves of the other rectangle.

Therefore, it is optimal to cut a rectangle in odd number of moves and to maximize this number or, in case that is not possible, to minimize even number of moves. In the end, we can solve this problem dinamically, trying every horizontal and vertical cut.

E (Reflections)

Between two reflections, the ray runs clockwise always the same arch a, $(0 < a < 2\pi)$. If we want it to come back to the same point reflected n times, the ray must go around the circle integer number of times: $(n+1) \times a = 2k \times \pi$, where k and $n+1$ are relative primes, so we have to count how many relative primes to $n+1$ not greater than $n+1$ are there, and that is Euler's function $\varphi(n+1)$. Next, if $n = {p_1}^{a_1} \times {p_2}^{a_2} \times \cdots$ (where p_i are primes), $\varphi(n) = (p_1 - 1) \times {p_1}^{a_1 - 1} \times (p_2 - 1) \times {p_2}^{a_2 - 1} \times \cdots$. If we want to have $A \leq n \leq B$, we just have to sum $\varphi(n+1)$.

F (Untitled)

Input files contained several English cardinals, each in range from “zero” to “nine” encoded by shifting every letter by one position right. For example, “two” became “uxp”, and “four” became “gpvs”. After finding this, it is pretty apparent that output file contains the product of given cardinals.