Test Data for Reverse

Algorithms:

Algorithm 1: Assign each of the first eight register one of the eight bits of 256. Set register 9 to 0. The goal of each register is to store the next number to be outputted whose last bit that is non-zero is the bit assigned to that register. For example, the starting values are: 128, 192, 224, 240, 248, 252, 254, and 255 (and 0, of course). As soon as the value 248 is printed out, it's goal becomes 232. Select a limit on the number of consecutive S-operations. After each P-operation, perform this number of S-operations, getting the register closer to their goal values. For each S-operation, of the registers which are not yet equal to their goal, select the goal which is largest and use the S-operation to get that register closer. If, at any time during the entire program, the goal is not ready by the time it is needed, increase the limit of S-operations and start over.

This algorithm scores 85% of the points.

Algorithm 2: Consider the case of trying to solve each input with only one S-operation. Clearly, register 1 might as well as be initialized to N. The register 2 can be N-2. After printing out N, one S-operation turns register 1 to N-1. Register 3 can be N-5. After printing N-2, S 3 1 makes register 1 N-4. After printing out N-2, S 1 2 turns register 2 into N-3, the next value to output. Continuing this through all the registers, 44 is the largest value of N which can be solved in only one S-operation.

This algorithm scores 90% of the points, if extended to dealing with multiple operations.

Algorithm 3: Perform algorithm 1, but have a fixed limit of 5 operations, instead of determining of the limit for each test case.

This algorithm scores 64% of the points.

Algorithm 4: Initialize the first register to zero, the next seven registers to $\lceil k\ N / 8 \rceil$. Reserve the last for work, but set its initial value to N. At each step, pick the register that is closest to the next value to be printed (the goal), without being over. If it is not equal to the goal, do an S-operation to store its value plus one into the work register. Keep doing S-operations, adding one to the work register, until it reaches the goal value.

This algorithm scores about 50% of the points.

The efficiency of this algorithm varies dramatically based on the exact algorithm used. In general, this class of algorithms is expected to receive about 30% of the points.

Test Data

Test #	Points	N	S limit
1	6	2	0
2	6	8	0
3	6	12	1
4	6	14	1
5	6	26	1
6	6	33	1
7	6	44	1
8	6	50	2
9	6	75	2
10	6	97	~~3~~2
11	6	112	3
12	6	140	4
13	7	173	4
14	7	200	4
15	7	240	4
16	7	255	4

Note that for test case 10, one student gave a better answer than our previously best-known answer.