Contest 7 Results

The problem presented in Contest 7 was the following.

Start with the natural numbers. At level 1, take one number, reject one number, take one, reject one, indefinitely. Level 1 thus yields the odd integers. At level 2, take the output from level 1: take two numbers, reject two numbers, take two, reject two, indefinitely. Repeat this process: at level K, take K numbers, reject K numbers,... What numbers will survive this sieving process?

The winning solution, unquestionably, comes from Tom Duff and Hugh Redelmeier, graduate students in computer science at the University of Toronto, who submitted 1199 terms of the resulting sequence. The first 58 of these terms, and some larger ones, are shown. Mr. Duff writes:

"We notice first that if we want only the first N numbers that the sieve generates, we need only simulate the first N-1 levels of the sieve, since the levels from N onward will all let at least the first N numbers pass. Each level of the K-level sieve may be regarded as a co-routine or asynchronous process which is in a producer-consumer relationship with the levels immediately above and below it. Level 1 of the sieve may be represented by the following piece of programme:

```
Loop:
      for J:=1 until I do
            receive M from level I-1
            send M to level I+1
      end
      for J:=1 until I do
            receive M from level I-1
      end
      goto Loop
```

A whole set of these co-routines, running in parallel, act a little like a bucket brigade. When a member of the bucket brigade receives a number from the previous member, it either passes the number on to the next member, or it discards the number. Of course, the first and last levels of the sieve are a little different, since the first level has no previous level to receive numbers from, and the last one prints the numbers out instead of passing them on.

With a little effort, this algorithm can be programmed to be relatively quick, but no matter how hard you try, its speed will still be at best a linear function of the result. Since the numbers outputted by the K-level sieve increase approximately exponentially, there is little hope of producing significant output of the sieve with any programme that has to simulate the sieve's operation.

A5-209

PC43-15

1	1
2	3
3	9
4	25
5	57
6	145
7	337
8	793
9	1921
10	3849
11	8835
12	18889
13	41473
14	92305
15	203211
16	432699
17	944313
18	2027529
19	4077769
20	8745153
21	18133305
22	37898113
23	80713737
24	169730259
25	358760457
26	750591867
27	1575313473
28	3255787851
29	6751959507
30	14108682265
31	29364255033
32	61173205587
33	126792880201
34	261786645129
35	542760030745
36	1090481316033
37	2254104779211
38	4576926103825
39	9375021033745
40	19362713813451
41	39746772236619
42	81889654169481
43	168348435476475
44	346352979792385
45	710707885619259
46	1454261791467673
47	2985844385599497
48	6112722593126091
49	12525096166152969
50	25453257177612675
51	52120032930644041
52	106216796997067065
53	217388980361779977
54	443667202522163353
55	905860955252113281
56	1851430774702624849
57	3778065321123848833
58	7700902493000163393

```
100     59721045190927381881499710261123

500     806443425214722972351549293423030824213355988901
        3727097631334253455067996456969941417811642485116
        1734818267880537671943571900003441962106674037047
        407233

1000    5313532548926165408921584229151796140386264683541
        4010630692427060681617155340580221063014675818895
        1608129661264218913555204037441755816463307268532
        6457016253865930987542603153162451709294642721995
        3290391722638945523574345904078467012319750764268
        7546266981530584938038605859667128033923096441412
        4937911825

1199    5119095356622829848295226812411528502341478017175
        9639533078954241574659785529231843938218888817594
        2482021288456532073349170622212188875613312551219
        7445952463318340192492856985443353429511600258319
        5017528328736772239016794711566448097482403123988
        7108383447448346850596669456669456665669177556178
        3991820852913232896296356290846110437322072526673
        577069709097224927441932353
```

Some results of the

K-level Sieve Problem

Let us analyze the behaviour of this programme. Each member of the bucket brigade receives a certain number of numbers from the previous member and passes along a certain fraction of them to the next member. When level I passes along its Nth number to level I+1, it will have received

$$2*N - ((N-1) \bmod I) - 1$$

numbers from level I-1. The Kth number outputted by the sieve is just the length of the sequence of numbers that must be inputted to the first level of the sieve to get K numbers out of the Kth level. We can compute that number by iterating on the above formula, like this:

```
N:=K
For I:=K-1 step -1 to 1 do
          N:=2*N-((N-1) mod I)-1
end
```

Thus, a complete programme to print the first KMAX numbers generated by the K-level sieve is:

```
for K:=1 until KMAX do
          N:=K
          for I:=K-1 step -1 until 1 do
                    N:=2*N-((N-1) mod I)-1
          end
          print K, N
end
```

In order to simplify the calculations in the inner loop, the above programme can be transformed by replacing K and N with K+1 and N+1. This yields the following programme:

```
for K:=0 until KMAX-1 do
          N:=K
          for I:=K step -1 until 1 do
                    N:=N*2-N mod I
          end
          print K+1, N+1
end
```

Our programme implements the last algorithm given above in UNIX assembly language. The main problem is that the variable N must be stored to extremely high precision. Our programme represents N as a sequence of base 10000 digits, each of which is stored in a single PDP-11 word."

We think that Duff and Redelmeier's work is a superb piece of computing. Members of our staff wrote eight different programs for the K-level sieve, the best of which yielded 42 terms of the sequence. The problem appears to be intrinsically useless, but is still a splendid exercise in logical analysis that calls for a high level of sophistication to achieve significant output. With 1199 terms of the final sequence now known, the problem passes into the realm of outstanding textbook problems. □