Scan A5315

J. A. Reeds,
J. E. Knuth
NJAS

3 pages
emails
1 sequence

f91

5315

```
From reeds Thu May 16 20:47:17 EDT 1991
Status: R

a[1]=1
a[2]=2
```

JA Reeds
DE Knuth
NJAS
email

```
a[3]=8
a[4]=42
a[5]=262
a[6]=1828
a[7]=13820
a[8]=110954
a[9]=933458
a[10]=8152860
a[11]=73424650
a[12]=678390116
a[13]=6405031050
a[14]=61606881612
a[15]=602188541928
a[16]=5969806669034
```

>From gauss!arpa!SAIL.Stanford.EDU!DEK Sun Jan 29 00:06:50 1989
Message-ID: <4gbTS@SAIL.Stanford.EDU>
Date: 28 Jan 89 2107 PST
From: Don Knuth <DEK@SAIL.Stanford.EDU>
Subject: meander numbers
To: reeds%gauss@RESEARCH.ATT.COM, las@RESEARCH.ATT.COM
CC: VRP@SAIL.Stanford.EDU

Vaughan Pratt couldn't resist computing those numbers on his SUN workstation, using a recurrence I thought of (an improvement of Koehler's approach)... The recurrence involves nC[n] values to compute the meanders that cross 2n times, where C[n] is the nth Catalan number. (Koehler's approach had n^2C[n] for the strip-of-stamps problem.) Here are Vaughan's results, which agree with yours up to n=14 (the largest value you sent me):

[the first experiments, Thursday morning 26 Jan]

```
n  A[n]       wall clock secs   max f(alpha,k)
11      73424650        8          4210
12      678390116       32         12198
13      6405031050      117        37378
```

[to go further meant going from 16 bits to 32, with paging onto disk; so he implemented a caching scheme, which slowed down the calculations:]

```
n  A[n]    wall clock secs  cache hits    cache accesses

6       1828       0       638             848
7       13820      1       2487            3279
8       110954     0       9658            12675
9       933458     1       37469           48915
10      8152860    6       148974          199360
11      73424650        20      595854     818026
12      678390116       85      2543840    3811940
13      6405031050      454     12649463        22757106
14      61606881612     2358    61337863        124032186
15      602188541928    12624   308321165       681140900
```

[then on Friday he had the value of A[16] but deleted it accidentally, so he has to compute it all again! We'll send A[16] soon. It looks like A[17] will be the limit of this particular approach; we need about 4^n units of memory as well as time. Vaughan can save the memory when computing

A[17], because he won't have to store the values that would otherwise be used to make A[18]. I suppose the 50% hit rate in the cache can be improved somehow, but still the numbers need to be stored somewhere...]

```
>From gauss!arpa!SAIL.Stanford.EDU!DEK Mon Jan 30 23:00:32 1989
Message-ID: <1$hZH1@SAIL.Stanford.EDU>
Date: 30 Jan 89  1149 PST
From: Don Knuth <DEK@SAIL.Stanford.EDU>
Subject: A16
To:   reeds%gauss@RESEARCH.ATT.COM, las@RESEARCH.ATT.COM

 28-Jan-89  2221   coraki!pratt@Sun.COM    A16
Received: from Sun.COM by SAIL.Stanford.EDU with TCP; 28 Jan 89  22:21:26 PST
Received: from sun.Sun.COM (sun-bb.sun.com) by Sun.COM (4.1/SMI-4.0)
   id AA05027; Sat, 28 Jan 89 22:22:47 PST
Received: from coraki.UUCP by sun.Sun.COM (4.0/SMI-4.0)
   id AA06143; Sat, 28 Jan 89 22:20:36 PST
Received: by  (4.0/SMI-4.0Beta)
   id AA13341; Sat, 28 Jan 89 22:20:19 PST
Date: Sat, 28 Jan 89 22:20:19 PST
From: Vaughan Pratt <coraki!pratt@Sun.COM>
Message-Id: <8901290620.AA13341@>
To: dke@sail.stanford.edu
Subject: A16
Cc: coraki!pratt@Sun.COM
```

```
 n          An        wall secs   cache hits        cache calls

13      6405031050       455     12649463         22757106
14      61606881612      2388    61337863         124032186
15      602188541928     13013   308321165        681140900
16      5969806669034    74844   1725643824       4147489672
```