Finding Partition Numbers via Brute Force

Rashad Chowdhury

4/21/2016

```
#Let g be the generating function of P(n), although we only consider\
    g up to a finite n.
def g(n):
    return prod([sum([x^(i*k) for k in [0..floor(n / i)]]) for i in \
   [1..n]])
```

```
#Does g what is desired. Let's check for g(10)
g10 = g(10); g10
```

```
(x^10 + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)*(x^10 + x^8 + x^6 + x^4 +
x^2 + 1)*(x^10 + x^5 + 1)*(x^10 + 1)*(x^9 + x^6 + x^3 + 1)*(x^9 + 1)*(x^8 + x^4 + 1)*(x^8
+ 1)*(x^7 + 1)*(x^6 + 1)
```

```
#We should get the same output as in class if I expand g(10)
g10.expand()
```

```
x^87 + x^86 + 2*x^85 + 3*x^84 + 5*x^83 + 7*x^82 + 11*x^81 + 15*x^80 + 22*x^79 + 30*x^78 +
42*x^77 + 54*x^76 + 70*x^75 + 91*x^74 + 116*x^73 + 145*x^72 + 181*x^71 + 222*x^70 +
270*x^69 + 325*x^68 + 386*x^67 + 454*x^66 + 529*x^65 + 616*x^64 + 707*x^63 + 805*x^62 +
910*x^61 + 1022*x^60 + 1135*x^59 + 1255*x^58 + 1374*x^57 + 1497*x^56 + 1618*x^55 +
1741*x^54 + 1856*x^53 + 1966*x^52 + 2069*x^51 + 2165*x^50 + 2246*x^49 + 2319*x^48 +
2379*x^47 + 2425*x^46 + 2456*x^45 + 2473*x^44 + 2473*x^43 + 2456*x^42 + 2425*x^41 +
2379*x^40 + 2319*x^39 + 2246*x^38 + 2165*x^37 + 2069*x^36 + 1966*x^35 + 1856*x^34 +
1741*x^33 + 1618*x^32 + 1497*x^31 + 1374*x^30 + 1255*x^29 + 1135*x^28 + 1022*x^27 +
910*x^26 + 805*x^25 + 707*x^24 + 616*x^23 + 529*x^22 + 454*x^21 + 386*x^20 + 325*x^19 +
270*x^18 + 222*x^17 + 181*x^16 + 145*x^15 + 116*x^14 + 91*x^13 + 70*x^12 + 54*x^11 +
42*x^10 + 30*x^9 + 22*x^8 + 15*x^7 + 11*x^6 + 7*x^5 + 5*x^4 + 3*x^3 + 2*x^2 + x + 1
```

```
#Can we find P(n) using g(n)?
g10.coefficients()
```

```
[[1, 0], [1, 1], [2, 2], [3, 3], [5, 4], [7, 5], [11, 6], [15, 7], [22, 8], [30, 9], [42,
10], [54, 11], [70, 12], [91, 13], [116, 14], [145, 15], [181, 16], [222, 17], [270, 18],
[325, 19], [386, 20], [454, 21], [529, 22], [616, 23], [707, 24], [805, 25], [910, 26],
[1022, 27], [1135, 28], [1255, 29], [1374, 30], [1497, 31], [1618, 32], [1741, 33], [1856,
34], [1966, 35], [2069, 36], [2165, 37], [2246, 38], [2319, 39], [2379, 40], [2425, 41],
[2456, 42], [2473, 43], [2473, 44], [2456, 45], [2425, 46], [2379, 47], [2319, 48], [2246,
49], [2165, 50], [2069, 51], [1966, 52], [1856, 53], [1741, 54], [1618, 55], [1497, 56],
```

```
[1374, 57], [1255, 58], [1135, 59], [1022, 60], [910, 61], [805, 62], [707, 63], [616,
64], [529, 65], [454, 66], [386, 67], [325, 68], [270, 69], [222, 70], [181, 71], [145,
72], [116, 73], [91, 74], [70, 75], [54, 76], [42, 77], [30, 78], [22, 79], [15, 80], [11,
81], [7, 82], [5, 83], [3, 84], [2, 85], [1, 86], [1, 87]]
```

```
#Let's define P(n) using the coefficients of g(n)
def P(n):
    Coeffs = g(n).coefficients()[:n+1]
    return [Coeffs[i][0] for i in range(len(Coeffs))]
```

```
#Using P(n) we get the first n numbers of the partition sequence
P(10)
```

```
[1, 1, 2, 3, 5, 7, 11, 15, 22, 30, 42]
```

```
#Let's go higher.
P(100)
```

```
[1, 1, 2, 3, 5, 7, 11, 15, 22, 30, 42, 56, 77, 101, 135, 176, 231, 297, 385, 490, 627,
792, 1002, 1255, 1575, 1958, 2436, 3010, 3718, 4565, 5604, 6842, 8349, 10143, 12310,
14883, 17977, 21637, 26015, 31185, 37338, 44583, 53174, 63261, 75175, 89134, 105558,
124754, 147273, 173525, 204226, 239943, 281589, 329931, 386155, 451276, 526823, 614154,
715220, 831820, 966467, 1121505, 1300156, 1505499, 1741630, 2012558, 2323520, 2679689,
3087735, 3554345, 4087968, 4697205, 5392783, 6185689, 7089500, 8118264, 9289091, 10619863,
12132164, 13848650, 15796476, 18004327, 20506255, 23338469, 26543660, 30167357, 34262962,
38887673, 44108109, 49995925, 56634173, 64112359, 72533807, 82010177, 92669720, 104651419,
118114304, 133230930, 150198136, 169229875, 190569292]
```