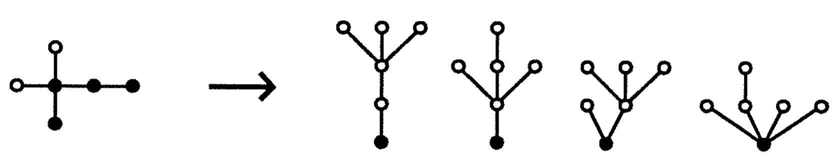
4

Rooted Trees ($v \leq 10$)

with probabilities

A rooted tree differs from an ordinary tree by having one vertex identified as the root (●). The root usually represents the beginning of a process or the top of a hierarchical organization. Each vertex stands at some remove from the root, along a unique procedural path or chain of command. The distance of a vertex from the root is its *depth* (or level), and the depth of the tree is the maximum depth of its vertices. Vertices in a rooted tree are related to each other by terms like parent, child, ancestor, descendant, sibling, etc. But genealogically speaking, rooted trees are usually drawn upside down.

One way of generating rooted trees is by choosing vertices in ordinary trees to act as roots. The tree below generates four different rooted trees, and no other ordinary tree will duplicate any of these four. In general, for every symmetry class of the vertex set of the ordinary tree there is one distinct rooted tree.



As a structure that spans the vertices of a connected graph, a tree can represent a search through the graph. There are two main types of search tree — breadth-first and depth-first — and their algorithms are described below in plain language. Each builds a spanning tree within a graph by appending edges to an initially empty set, choosing new edges that form no cycle with the existing edges. Beyond these similarities, breadth-first has the structure of an outline, while depth-first resembles a drainage system — a river with tributaries — or a spine with ribs.

Breadth-First Search

• choose a vertex to be the root of the search
• Level 1: find its neighbors I, II, III, IV, . . . and append them to the root
• Level 2: find neighbors of I (A, B, C, . . .) that form no cycle, append to I
• find neighbors of II (A′, B′, C′, . . .) that form no cycle, append to II
• repeat for III, IV, . . .
• Level 3: find neighbors of A (1, 2, 3, . . .) that form no cycle, append to A
• find neighbors of B (1′, 2′, 3′, . . .) that form no cycle, append to B
• repeat for C, D, . . .
• repeat for A′, B′, C′, . . .
• repeat for A″, B″, C″, . . .
• • •
• Level 4: find neighbors of 1 (a, b, c, . . .) that form no cycle, append to 1
• repeat for 2, 3, . . .
• repeat for 1′, 2′, 3′, . . .
• • •
continue until all vertices are in the tree

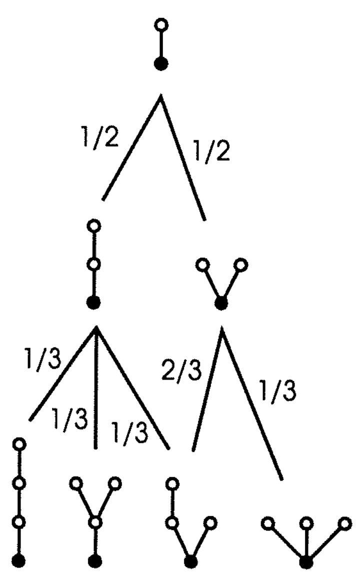
Depth-First Search

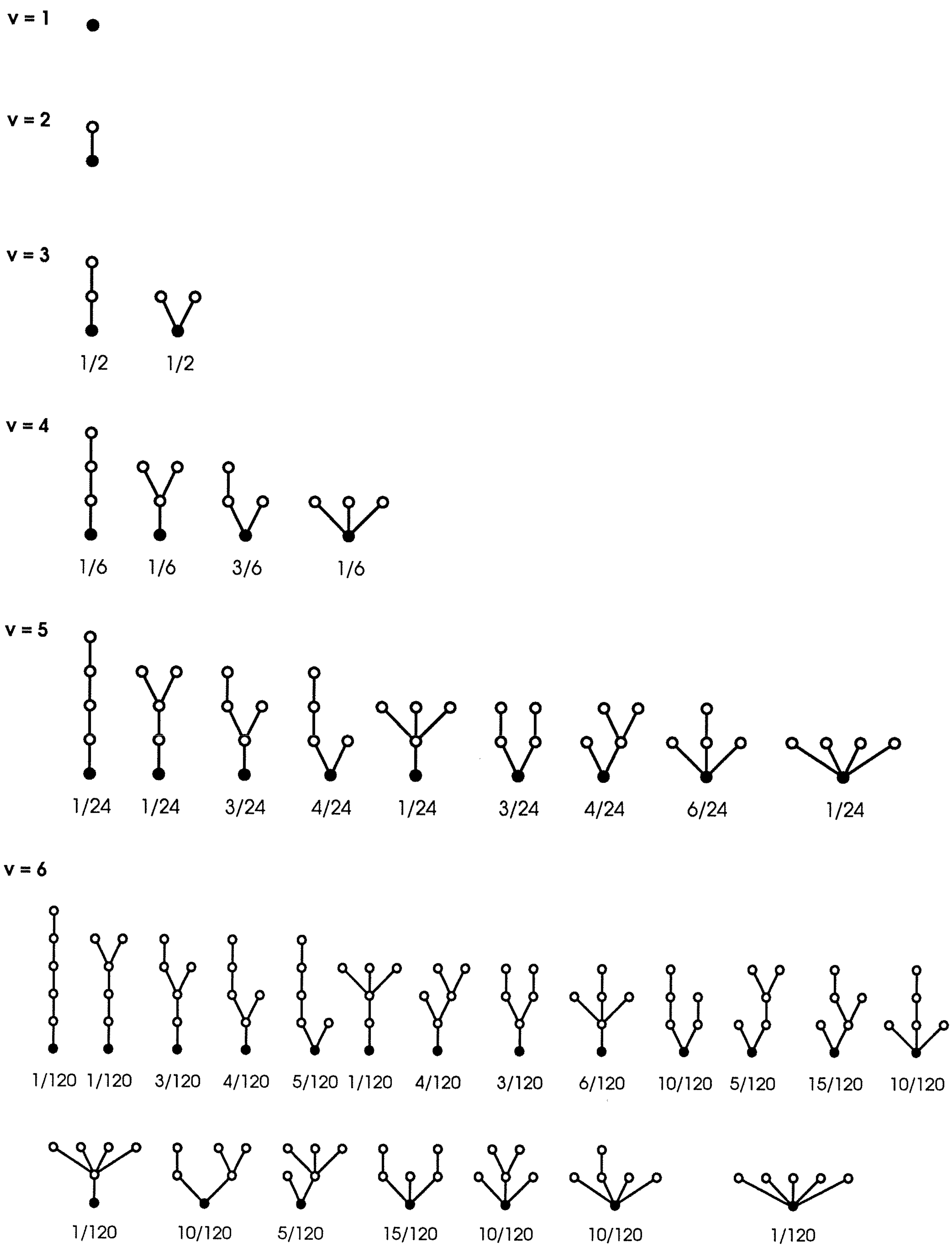
• choose a vertex A_0 to be the root of the search
• Level 1: connect A_0 to a neighbor A_1
• Level 2: connect A_1 to a neighbor A_2 that forms no cycle
• Level 3: connect A_2 to a neighbor A_3 that forms no cycle
• Level 4 through n: repeat until stuck at A_n. Path A_0 through A_n forms the spine.
• backtrack to A_{n-1} and connect to another neighbor B_n, and that to $B_{n+1}, \ldots$ forming no cycles
• continue depth search out of A_{n-1}, backtracking if necessary, until stuck
• backtrack to A_{n-2}, do depth search to $C_{n-1}, C_n, \ldots$ backtracking if necessary
• • •
continue backtracking along the spine to the root, if necessary, until all vertices are in the tree

Notice that breadth-first is the way an outline is constructed, but the order in which an outline is written and read (I, A, 1, a, (1), . . .) is depth-first.

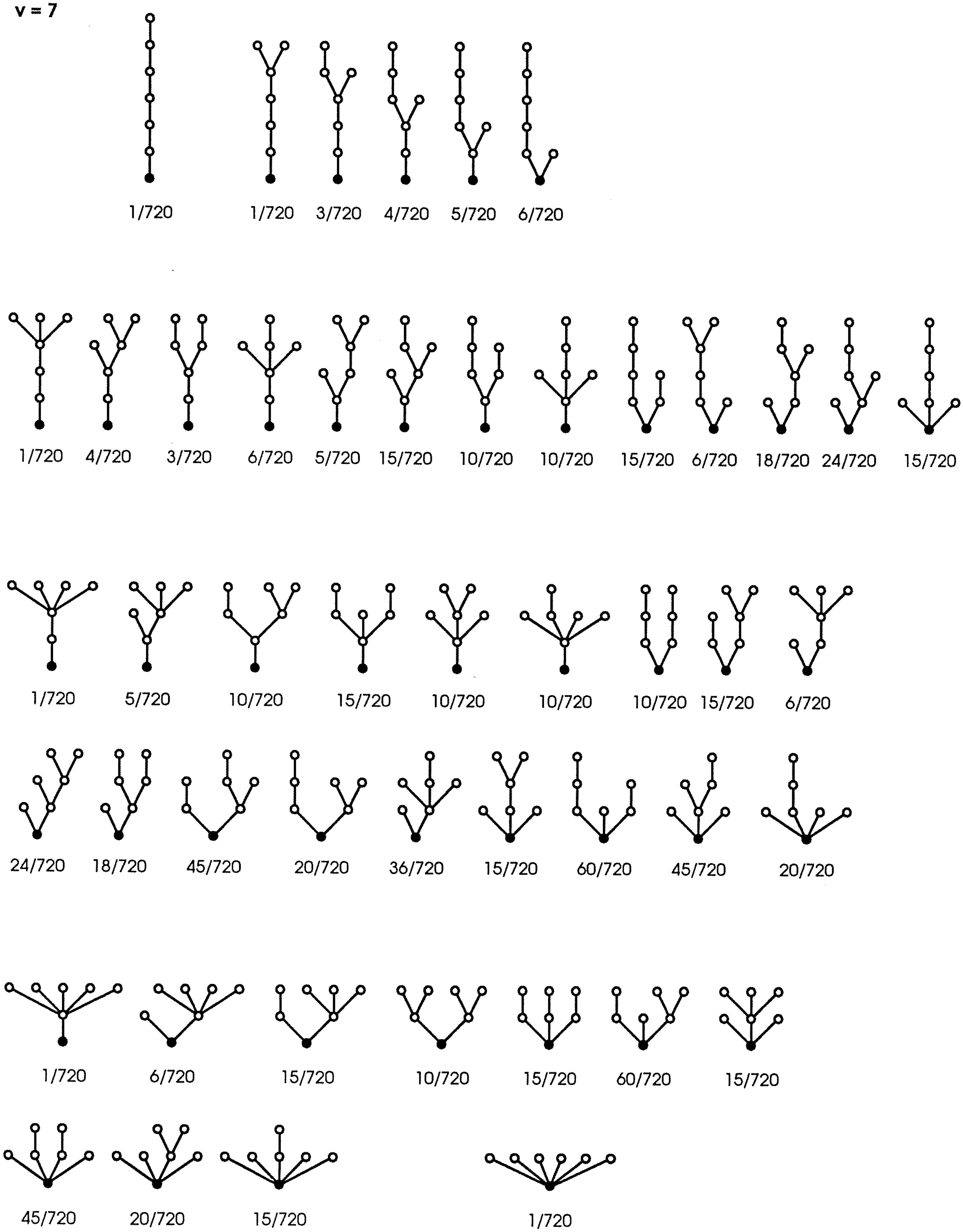
The probabilities of randomly built rooted trees are determined in the same way as probabilities of ordinary trees (Ch 3, p 65) — with one simple difference. The process begins with the root. Then, as before, the rest is leaf addition, with every vertex equally likely to be the attachment point.

For v = 8 and 9 the denominator of probabilities is given once per page.



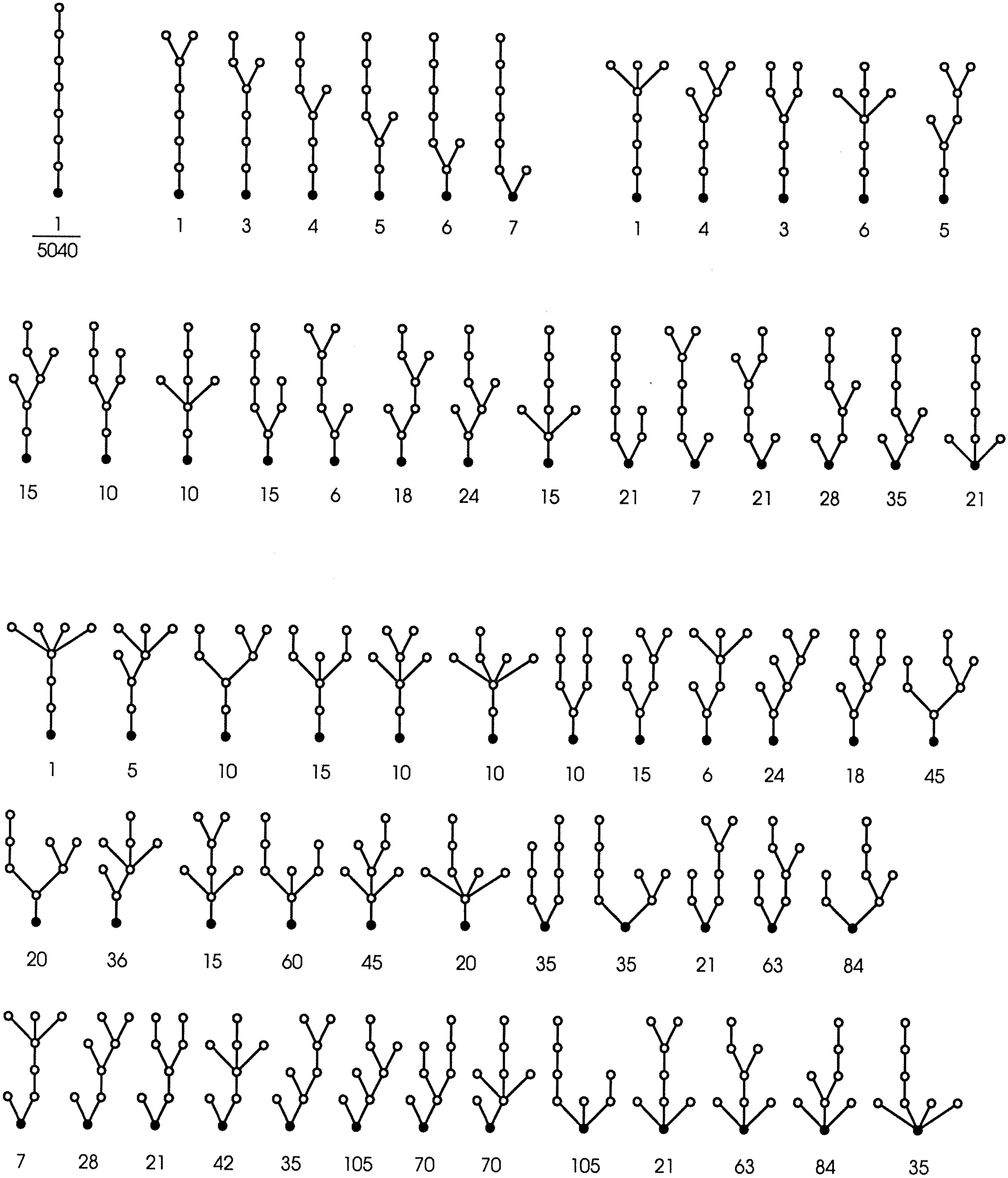

v = 1
v = 2
v = 3
1/2 1/2
v = 4
1/6 1/6 3/6 1/6
v = 5
1/24 1/24 3/24 4/24 1/24 3/24 4/24 6/24 1/24
v = 6
1/120 1/120 3/120 4/120 5/120 1/120 4/120 3/120 6/120 10/120 5/120 15/120 10/120
1/120 10/120 5/120 15/120 10/120 10/120 1/120

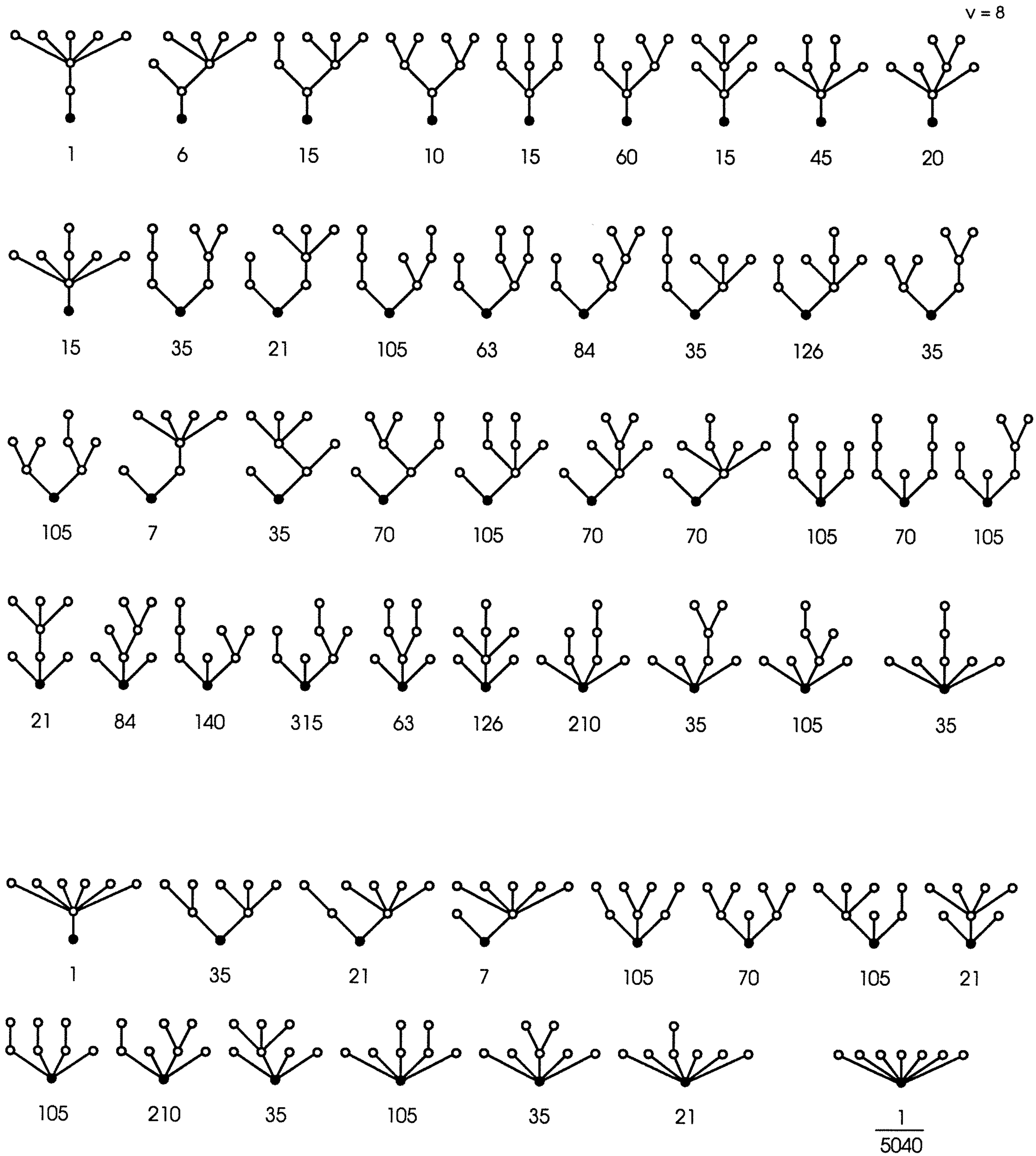
v = 7



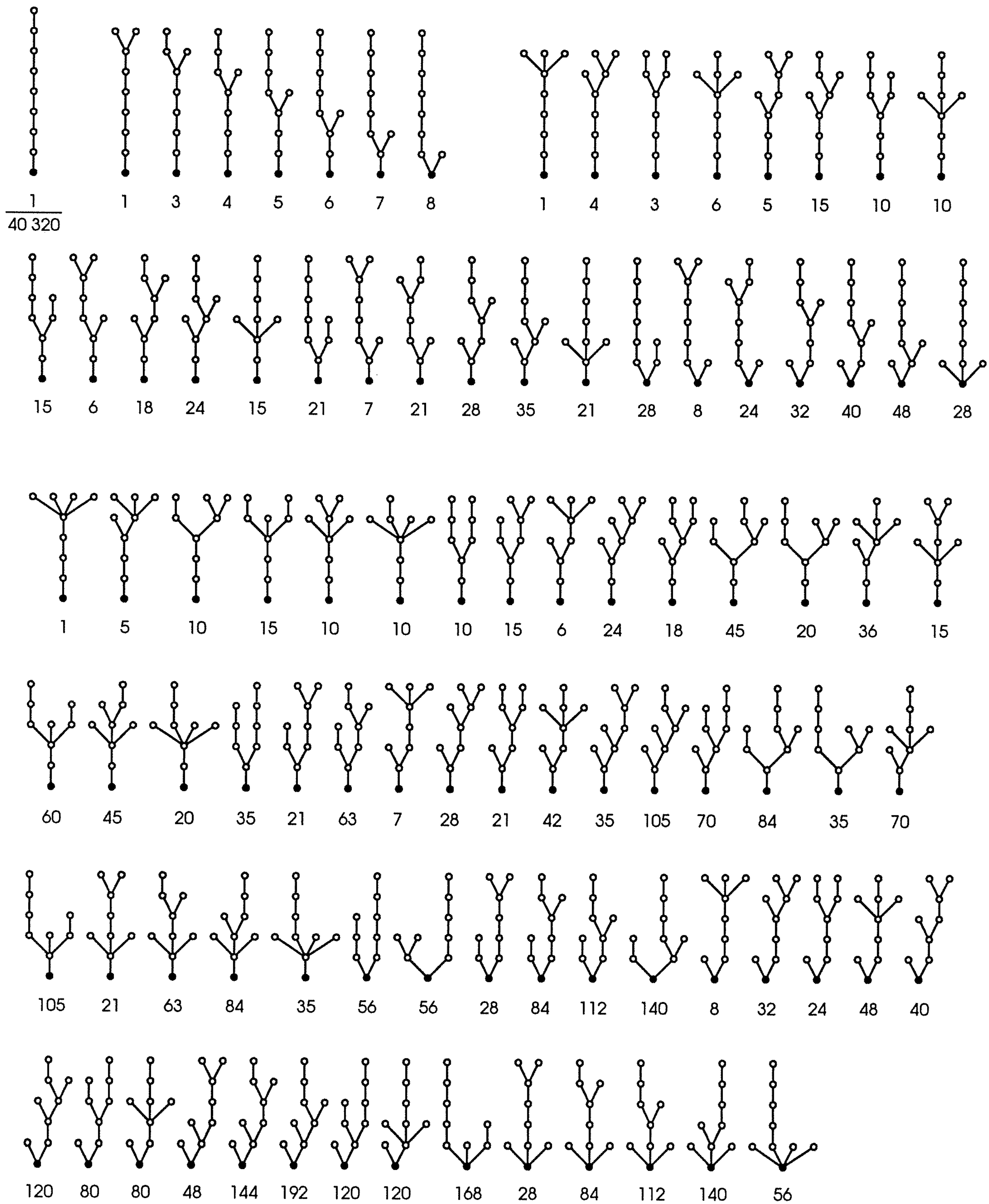
1/720 1/720 3/720 4/720 5/720 6/720
1/720 4/720 3/720 6/720 5/720 15/720 10/720 10/720 15/720 6/720 18/720 24/720 15/720
1/720 5/720 10/720 15/720 10/720 10/720 10/720 15/720 6/720
24/720 18/720 45/720 20/720 36/720 15/720 60/720 45/720 20/720
1/720 6/720 15/720 10/720 15/720 60/720 15/720
45/720 20/720 15/720 1/720

v = 8

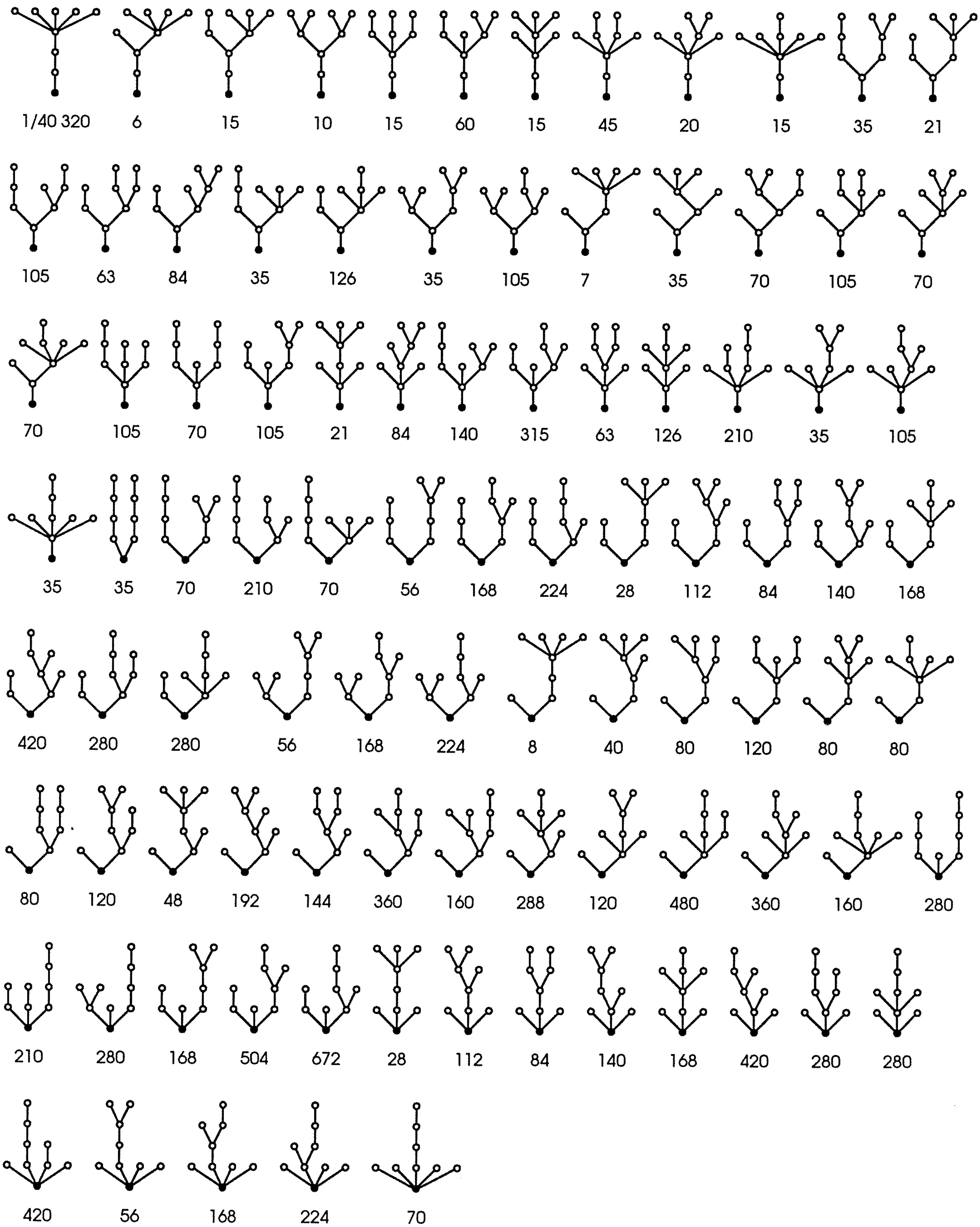

1/5040
1 3 4 5 6 7
1 4 3 6 5
15 10 10 15 6 18 24 15 21 7 21 28 35 21
1 5 10 15 10 10 10 15 6 24 18 45
20 36 15 60 45 20 35 35 21 63 84
7 28 21 42 35 105 70 70 105 21 63 84 35


v = 8
1 6 15 10 15 60 15 45 20
15 35 21 105 63 84 35 126 35
105 7 35 70 105 70 70 105 70 105
21 84 140 315 63 126 210 35 105 35
1 35 21 7 105 70 105 21
105 210 35 105 35 21
1
5040

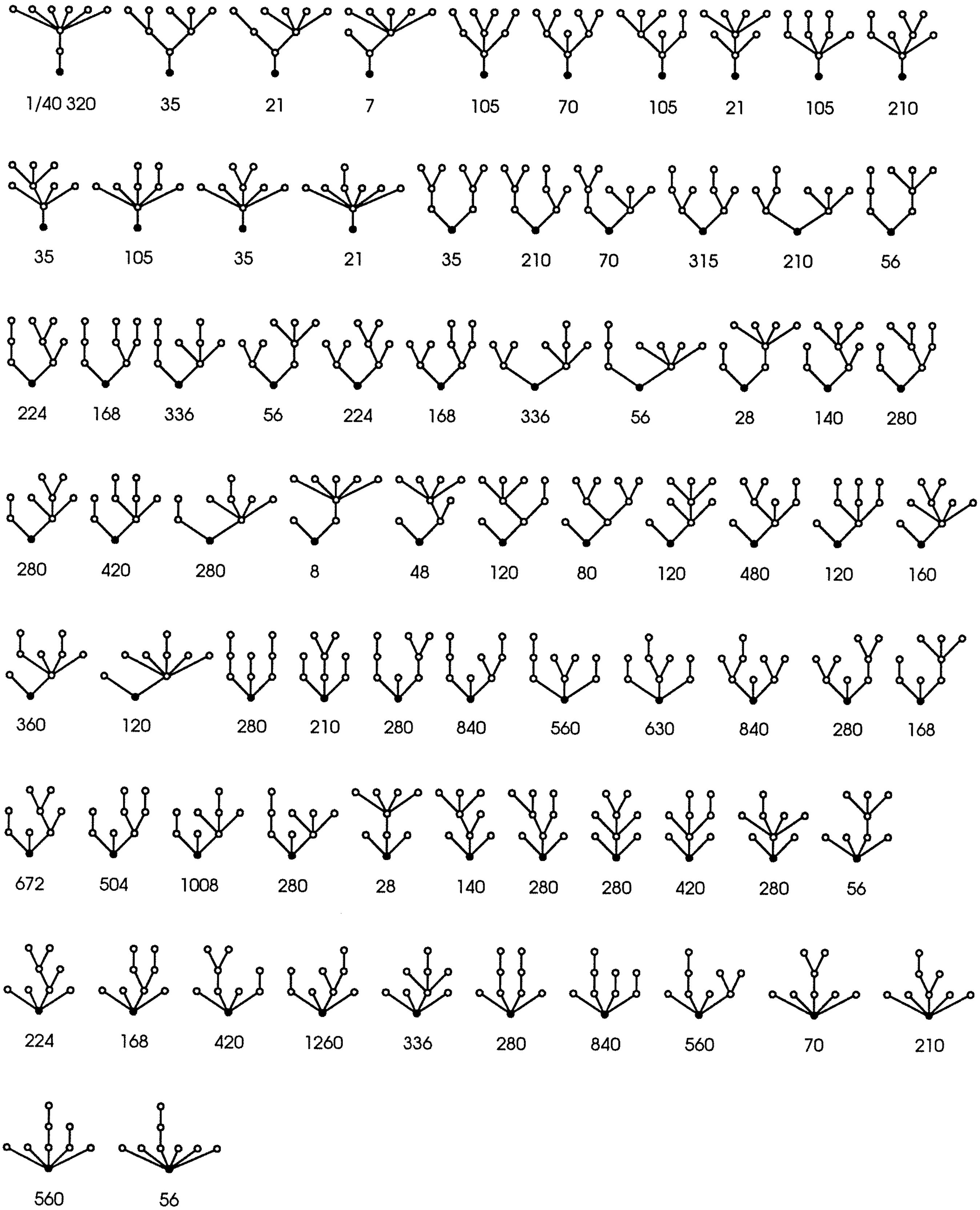
v = 9


1
40 320
1 3 4 5 6 7 8
1 4 3 6 5 15 10 10
15 6 18 24 15 21 7 21 28 35 21 28 8 24 32 40 48 28
1 5 10 15 10 10 10 15 6 24 18 45 20 36 15
60 45 20 35 21 63 7 28 21 42 35 105 70 84 35 70
105 21 63 84 35 56 56 28 84 112 140 8 32 24 48 40
120 80 80 48 144 192 120 120 168 28 84 112 140 56

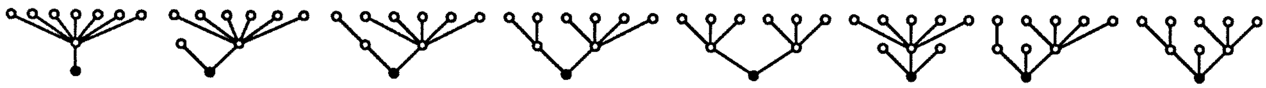
v = 9


1/40 320 6 15 10 15 60 15 45 20 15 35 21
105 63 84 35 126 35 105 7 35 70 105 70
70 105 70 105 21 84 140 315 63 126 210 35 105
35 35 70 210 70 56 168 224 28 112 84 140 168
420 280 280 56 168 224 8 40 80 120 80 80
80 120 48 192 144 360 160 288 120 480 360 160 280
210 280 168 504 672 28 112 84 140 168 420 280 280
420 56 168 224 70

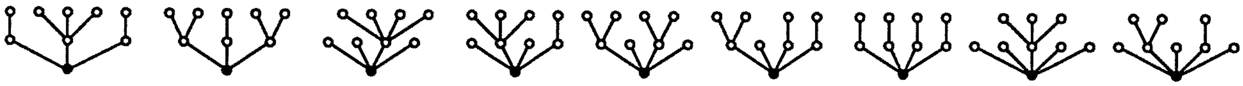
v = 9


1/40 320 35 21 7 105 70 105 21 105 210
35 105 35 21 35 210 70 315 210 56
224 168 336 56 224 168 336 56 28 140 280
280 420 280 8 48 120 80 120 480 120 160
360 120 280 210 280 840 560 630 840 280 168
672 504 1008 280 28 140 280 280 420 280 56
224 168 420 1260 336 280 840 560 70 210
560 56

v = 9



1 8 28 56 35 28 168 280

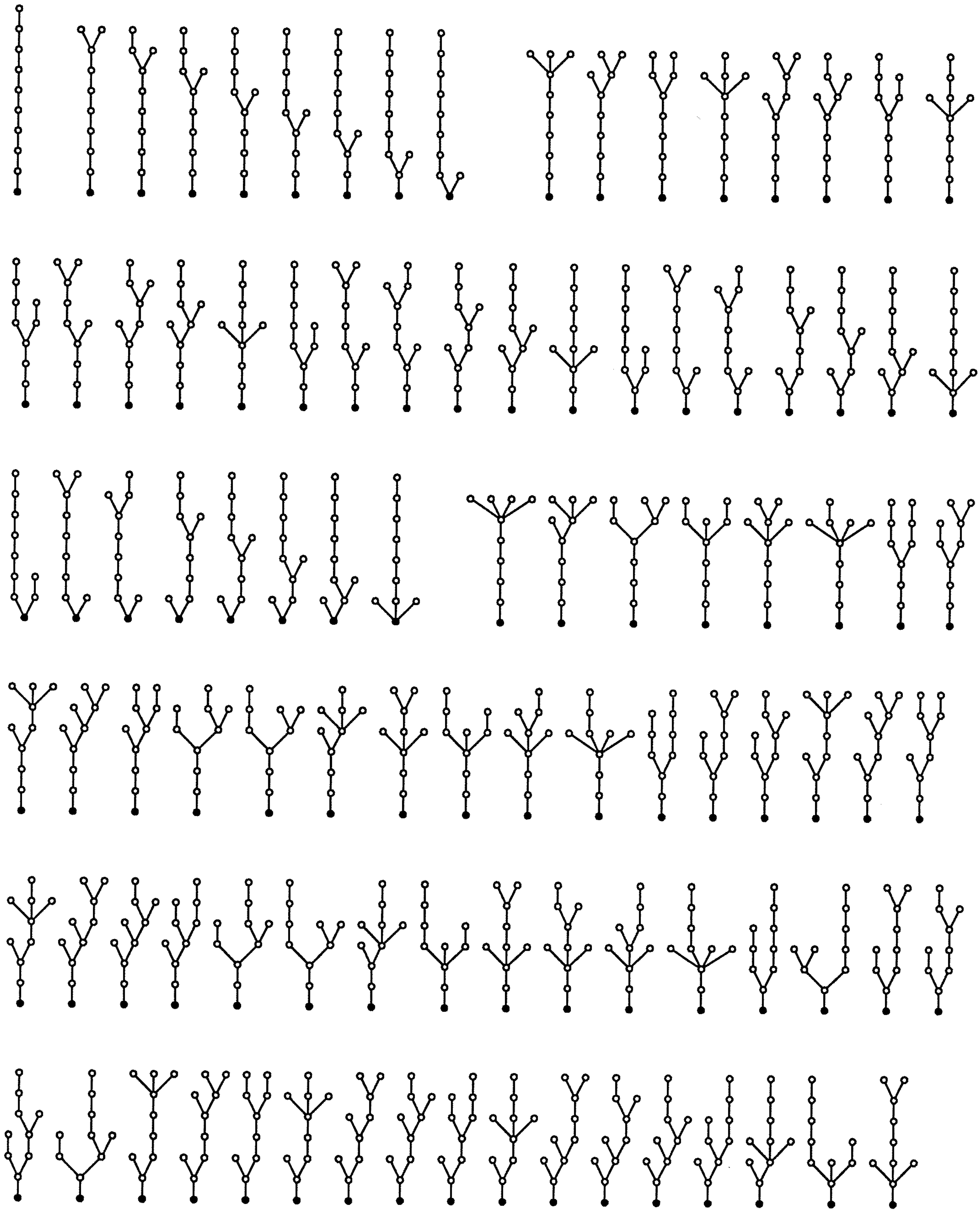


210 280 56 420 280 840 105 70 560

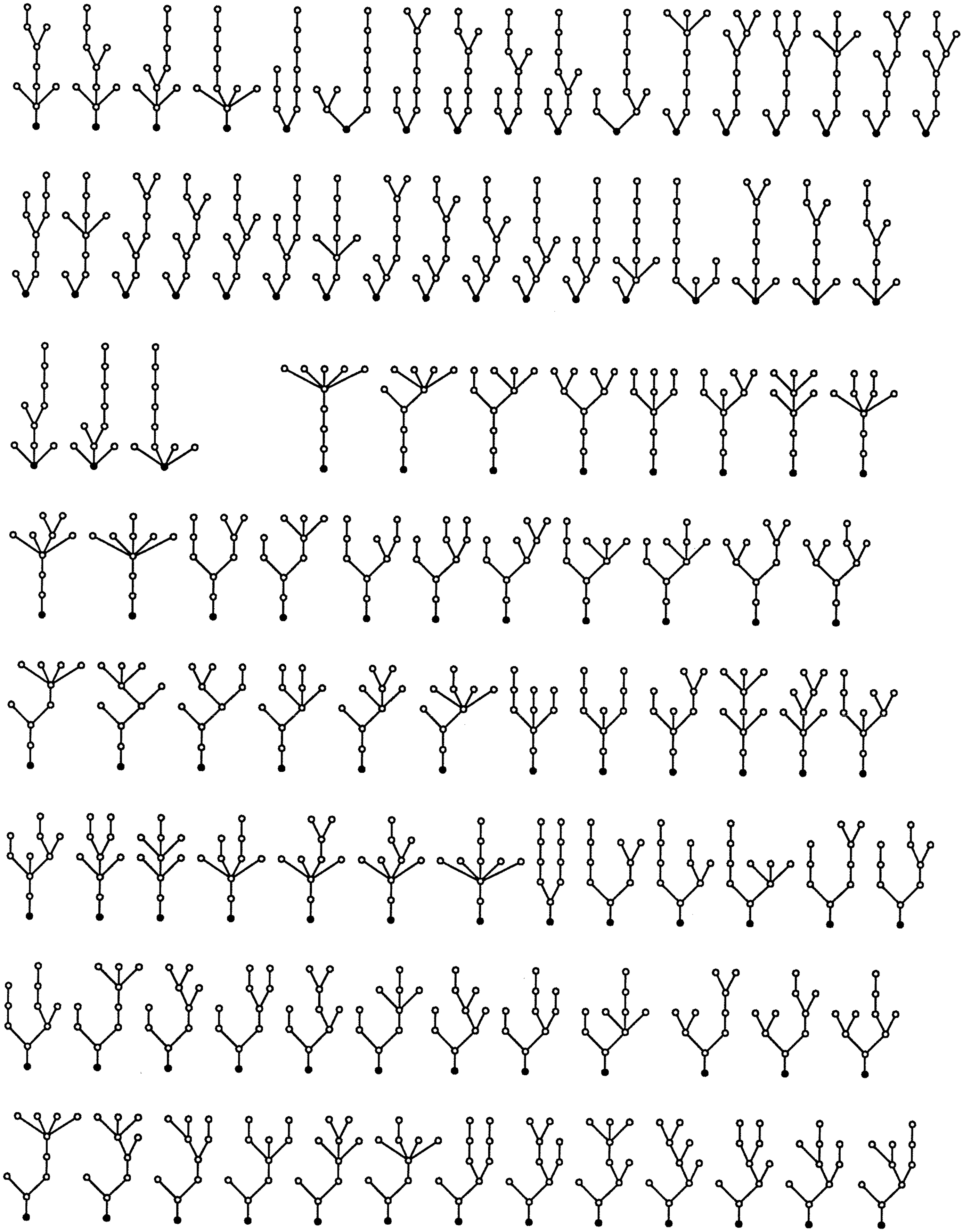
420 56 210 28

$\frac{1}{40\,320}$

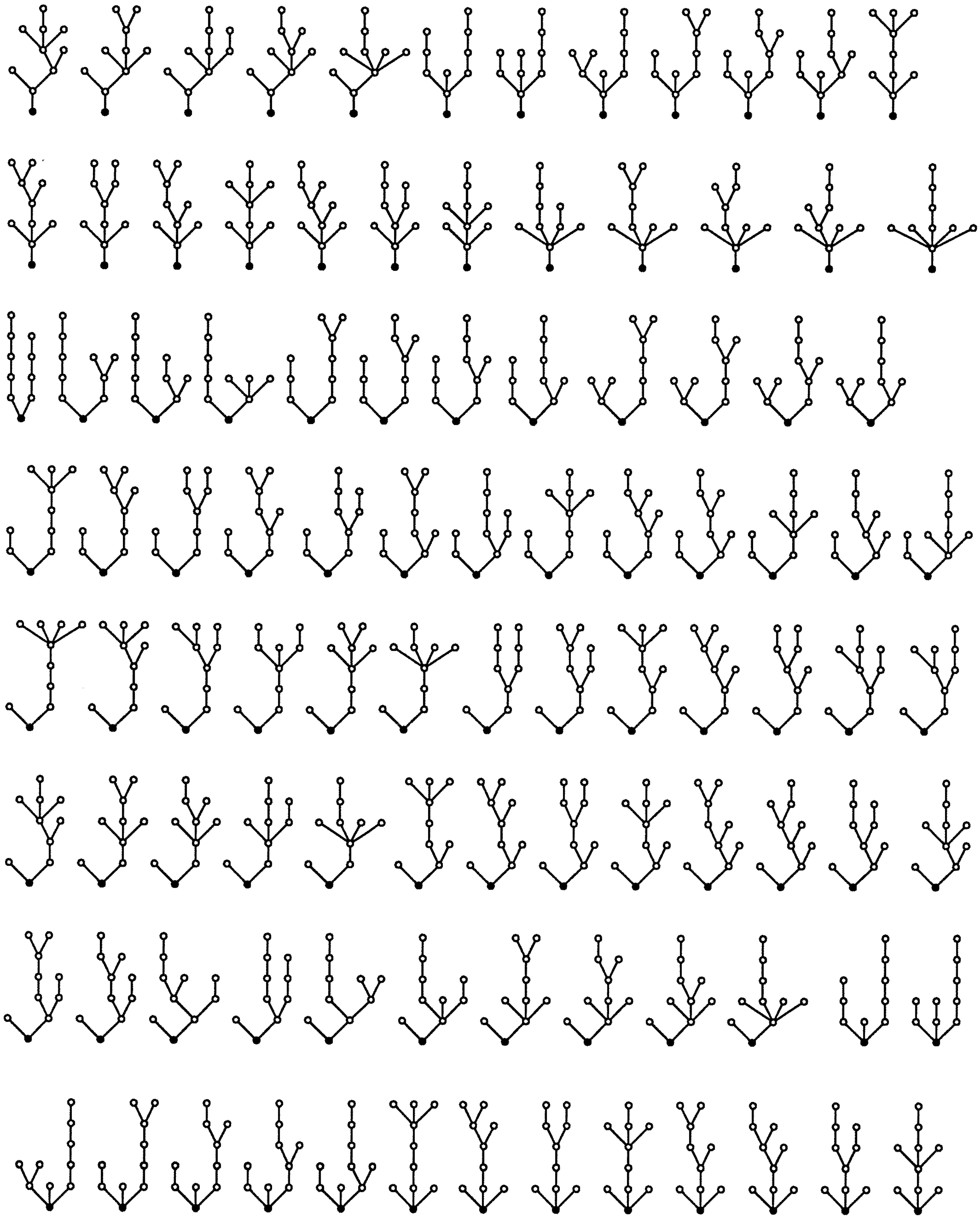
v = 10



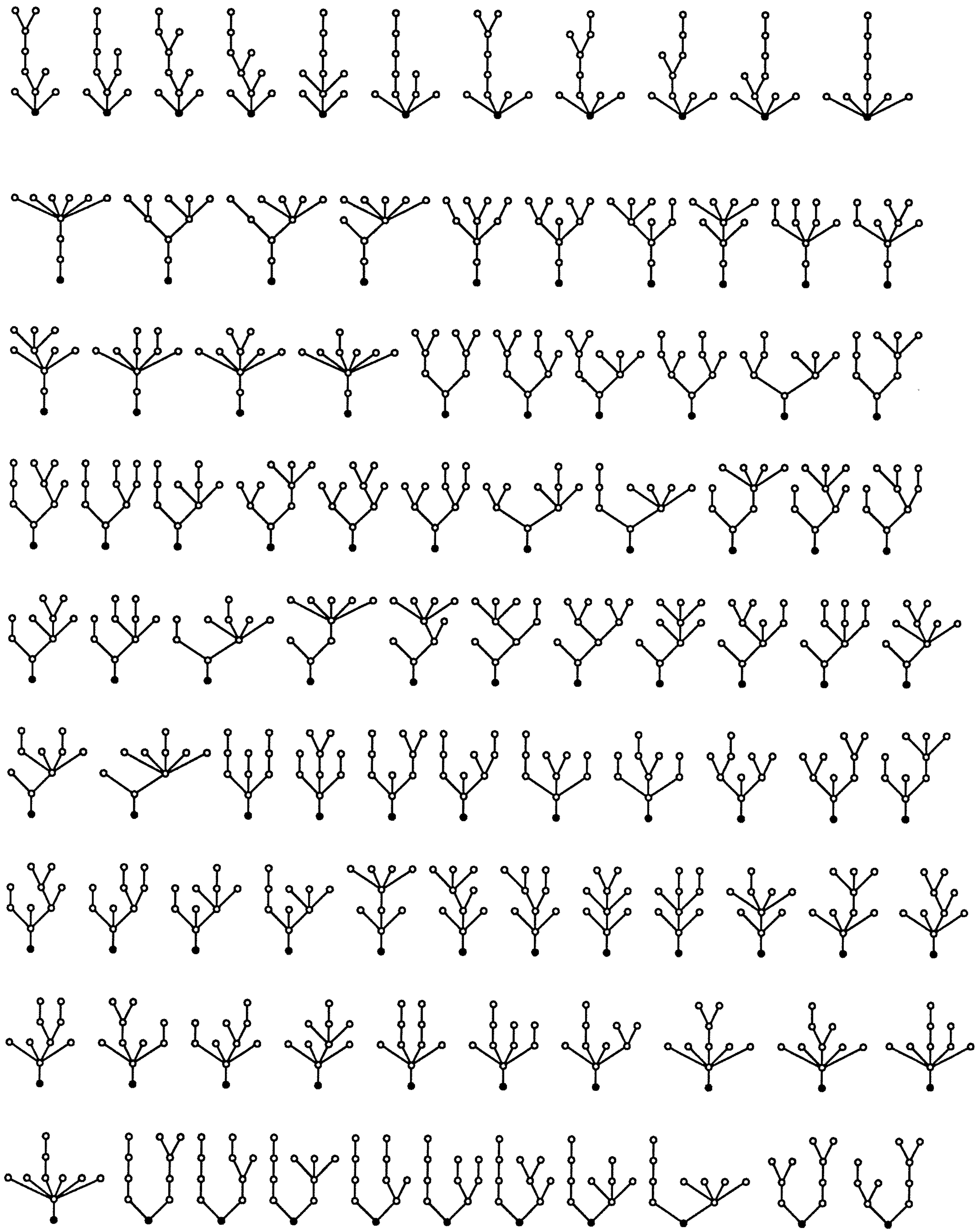
v = 10



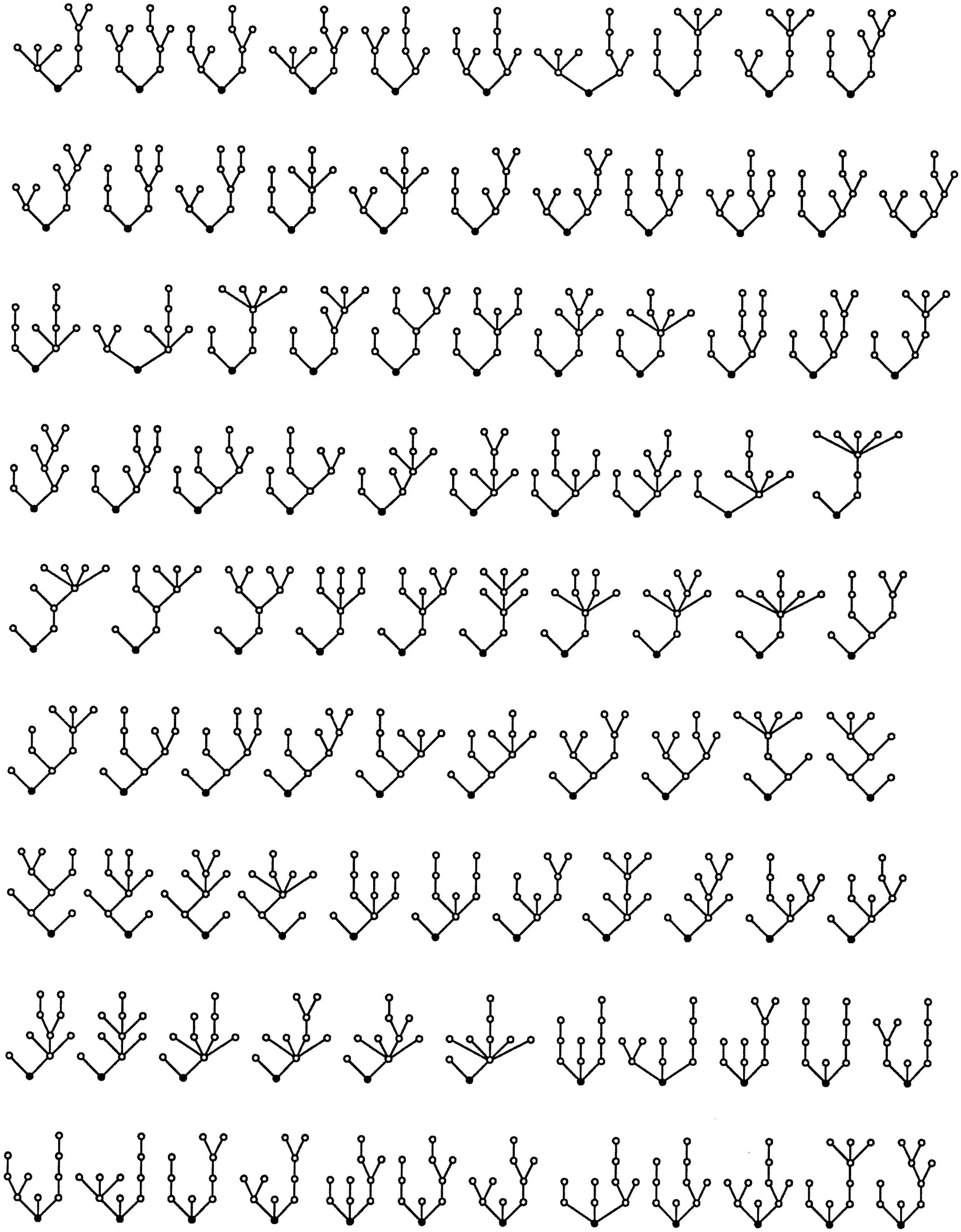
v = 10



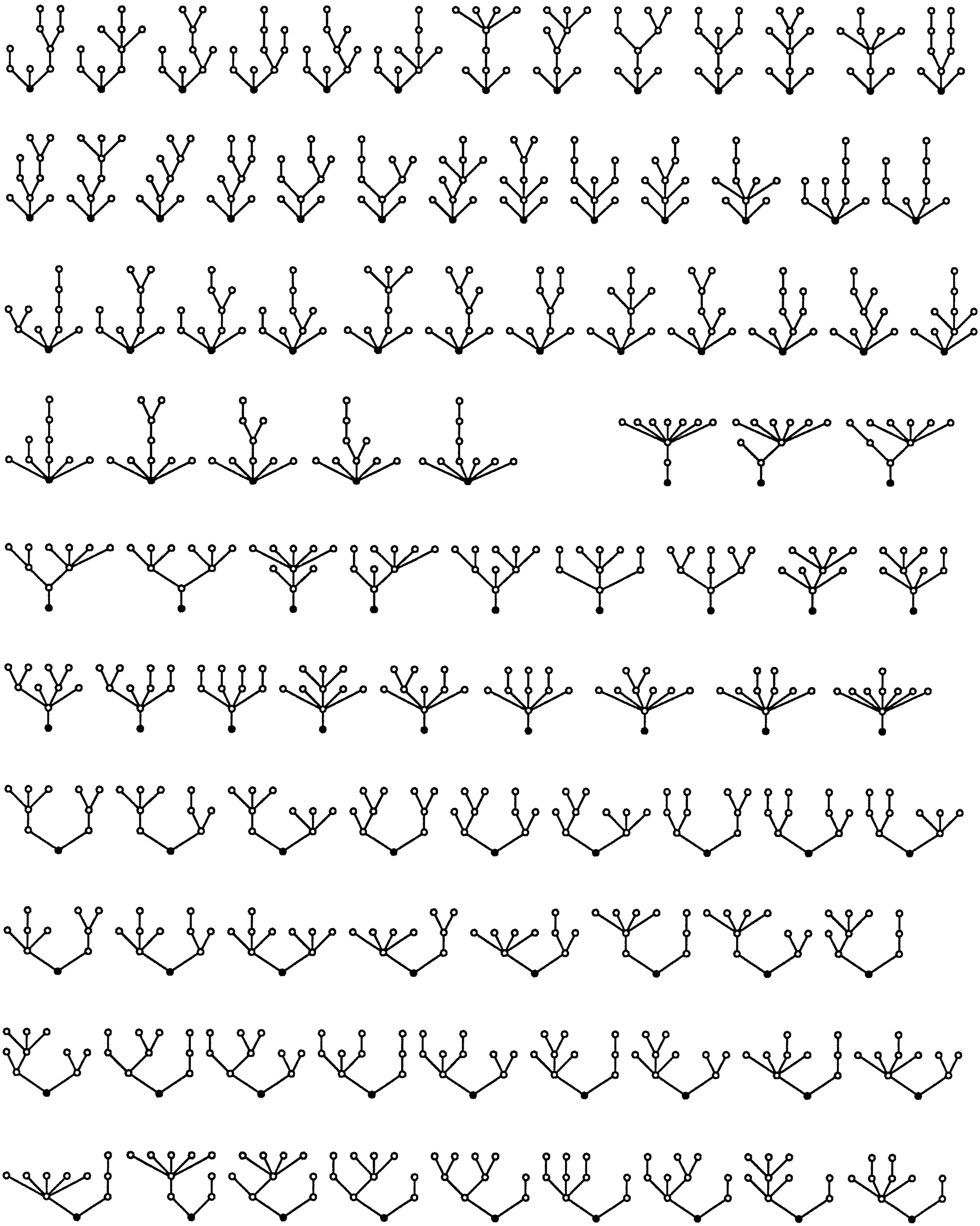
v = 10



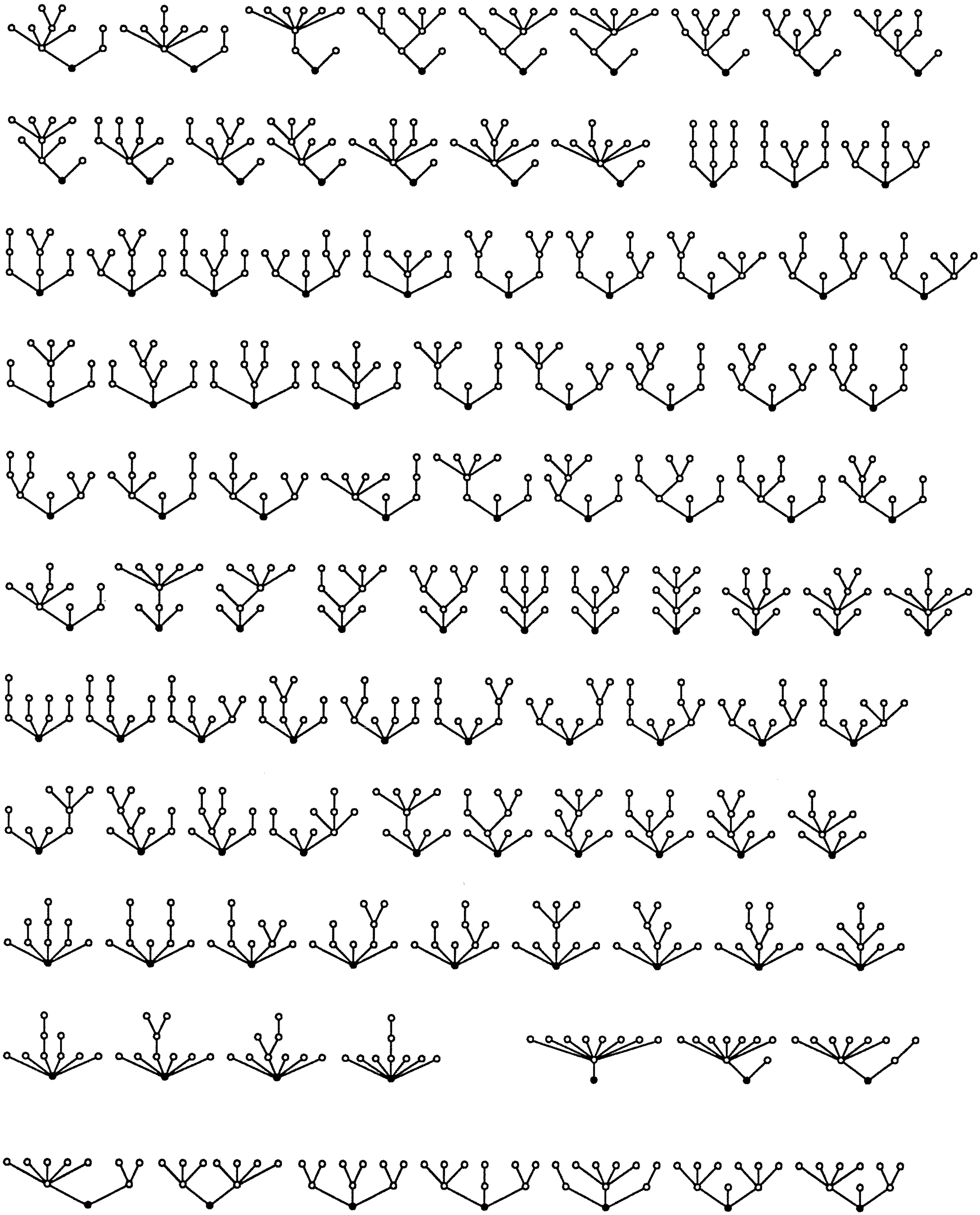
v = 10

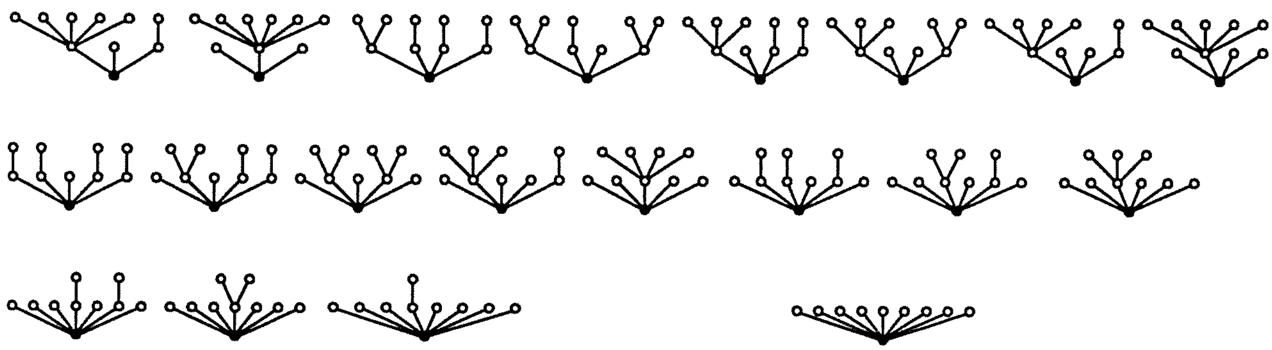


v = 10



v = 10





Numbers of rooted trees by depth **d** or breadth **b**

v	d = 1	2	3	4	5	6	7	8	9	10	11	12	totals	mean d	random mean d
1	0												1	0	0
2	1												1	1	1
3	1	1											2	1.5	1.5
4	1	2	1										4	2	2
5	1	4	3	1									9	2.444	2.375
6	1	6	8	4	1								20	2.9	2.658
7	1	10	18	13	5	1							48	3.292	2.957
8	1	14	38	36	19	6	1						115	3.696	3.203
9	1	21	76	93	61	26	7	1					286	4.059	3.424
10	1	29	147	225	180	94	34	8	1				719	4.416	
11	1	41	277	528	498	308	136	43	9	1			1 842*	4.751	
12	1	55	509	1198	1323	941	487	188	53	10	1		4 766*	5.076	
13	1	76	924	2666	3405	2744	1615	728	251	64	11	1	12 486*	5.385	

*computed by Allen Schwenk, University of Michigan

v	b = 1	2	3	4	5	6	7	8	9	totals	mean b
1	0									1	0
2	1									1	1
3	1	1								2	1.5
4	1	2	1							4	2
5	1	4	3	1						9	2.444
6	1	6	8	4	1					20	2.9
7	1	9	18	14	5	1				48	3.333
8	1	12	35	39	21	6	1			115	3.774
9	1	16	62	97	72	30	7	1		286	4.234
10	1	20	103	212	214	120	40	8	1	719	4.647

The 286 rooted trees on 9 vertices
partitioned by depth and breadth

breadth (number of terminal vertices)

depth	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1								1
2				1	4	9	7	
3			3	20	32	21		
4		1	16	41	35			
5		3	23	35				
6		5	21					
7		7						
8	1							

The 719 rooted trees on 10 vertices
partitioned by depth and breadth

breadth

depth	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1									1
2					2	7	12	8	
3			1	13	48	57	28		
4			11	64	94	56			
5		2	29	79	70				
6		4	34	56					
7		6	28						
8		8							
9	1								