Bellman Ford Algorithm

Throughout, when we say *graph* we mean undirected graph.

The Bellman Ford algorithm solves the single source minpath problem for a weighted directed graph G. Weights may be negative, but, as always for minpath problems, no cycle of G may have negative weight. Let n and m the numbers of vertices and arcs of G, respectively. The Bellman Ford algorithm computes, for each vertex i, the least cost path from 0 to i. We call that path $\pi(i)$. The worst case time complexity for the Bellman Ford algorithm is $O(nm)$.

The Main Loop. We assume that the vertices of G are the integers from 0 to $n-1$. After ℓ iterations of the main loop, our data structure consists of

1. An array of arcs, a_j for $0 \leq j < m$. An arc of weight w from vertex s to vertex t is represented by the ordered triple (s, t, w).
2. $V(i)$, the minimum weight of any path from 0 to i which has been found so far, ∞ if no such path has been found. We write $\sigma(i)$ for the least cost path from 0 to i found so far, the current approximation of $\pi(i)$.
3. For any $i > 0$, *back*(i), the next-to-the last vertex on $\sigma(i)$. provided $V[i] < \infty$.
4. For any $i > 0$, *backweight*(i), weight of the arc from *back*(i) to i, provided $V(i) < \infty$.

The structure is initialized by setting $V(0) = 0$ and $V(i) = \infty$ for all $i > 0$. $\sigma(0)$ is the trivial path and $\sigma(i)$ is undefined for $i > 0$.

The largest possible number of edges in $\sigma(i)$ is $n-1$, since the path visits each vertex at most once. In the worst case, the main loop iterates $n-1$ times. If $\sigma(i)$ consists of ℓ edges, that path will be found at either the $(\ell-1)^{\text{st}}$ or the ℓ^{th} iteration of that loop.

The Inner Loop. Each iteration of the main loop contains an inner loop which iterates m times, once for each arc. Consider the j^{th} iteration of that loop. Let $e_j = (s, t, w)$. The iteration uses the relaxation step to try to improve $\sigma(t)$. Let $\tau = \sigma(s)e_j$, the concatenation of the path with the arc. If τ has weight less than $\sigma(t)$, it becomes the new $\sigma(t)$. For all i, $\sigma(i)$ eventually becomes $\pi(i)$, and $V(i)$ becomes the total weight of $\pi(i)$.

Speeding Up Bellman Ford

We give two shortcuts to speed up computation In the example discussed below, We let $n = 100$ and $m = 1000$. The arcs were initialized with the help of a random number generated, and care was taken that there would be no negative cycle. The number of executions of the relaxation routine, arguably the most complex part of the algorithm, was 2020 for the example, far better than the worst case estimate of 100000.

Overall stability. Let $\mathcal{D}(\ell)$ be the data structure after ℓ iterations of the main loop. Bellman Ford computes $\mathcal{D}(\ell)$ determimistically using only the data of $\mathcal{D}(\ell-1)$, and thus, if $\mathcal{D}(\ell) = \mathcal{D}(\ell-1)$, then $\mathcal{D}(\ell+1) = \mathcal{D}(\ell)$. We use a Boolean variable to determine whether any part of $\mathcal{D}$ was changed during the ℓth loop. If not, we can terminate the computation. If $\pi(i)$ contains at most ℓ arcs for each i, the data structure will stabilize after at most ℓ iterations of the main loop. In our example, the maximum number of edges of any $\pi(i)$ is 5, and the data structure stabilizes after 4 iterations of the main loop.

Partial stability. It can happen that some value of V is unchanged during iteration ℓ, but decreases at some later iteration. We can still take advantage of this fact for iteration $\ell + 1$. Let $e_j = (s, t, w)$, and suppose $V(s)$ is unchanged during iteration ℓ. We keep track of which iteration of the main loop during which $V(s)$ changed. If $V(s)$ has not changed during one full iteration of the main loop, it cannot contribute to an improvement of $\sigma(t)$, hence that relaxation step can be skipped.

Example

We give files showing the list of 1000 arcs of a directed graph of 100 vertices. We then give the final values of $V(i)$ for each vertex i, as will as the path $\pi(i)$, which is shown with alternating vertices and arc weights, where the weights are parenthesized.

93 34 6 34 21 4 85 57 1 61 62 7 41 16 0 50 62 4 99 17 2 73 15 3
51 64 1 63 91 1 37 37 5 28 71 -1 87 56 2 41 70 4 65 11 7 17 61 3
51 12 0 6 38 3 64 89 6 54 4 3 79 41 6 38 69 3 70 56 3 60 49 5
65 14 5 86 83 5 69 35 3 21 93 1 89 9 -1 73 64 2 48 95 5 13 33 3
49 55 4 93 68 2 60 33 7 86 71 1 77 40 1 81 61 -2 23 50 0 54 75 6
42 24 6 19 89 8 69 38 1 76 83 4 33 43 6 56 81 2 66 11 4 12 92 6
2 68 3 2 74 5 18 16 2 77 87 0 73 57 3 25 33 6 96 18 3 53 26 7
80 93 2 48 5 1 29 59 5 60 62 9 19 80 3 2 10 1 26 83 6 40 8 4
38 57 3 31 10 1 5 90 6 91 38 2 21 67 6 71 80 6 95 99 7 88 54 -1
69 32 5 10 73 -1 33 63 2 79 94 9 99 51 2 64 42 1 86 15 3 15 86 4
11 34 6 87 22 4 73 43 9 42 54 5 24 39 5 63 18 4 12 69 3 4 33 5
34 19 -2 35 87 3 69 50 4 2 37 2 89 10 -1 60 4 5 57 29 4 16 92 3
22 5 2 79 61 4 78 47 2 45 82 2 99 51 2 86 53 -1 48 94 6 6 7 4
17 64 2 20 80 3 94 54 1 37 33 7 17 12 1 17 9 1 56 8 7 45 95 3
71 95 7 59 1 2 52 71 0 25 43 7 91 89 7 66 78 5 50 96 3 33 13 3
99 70 5 68 15 0 42 39 4 63 98 3 39 2 4 31 28 3 4 71 -1 83 38 -1
95 40 4 72 26 5 58 25 1 52 45 4 46 39 1 83 3 -1 25 42 3 39 74 7
96 30 6 94 13 3 19 60 0 92 32 3 79 90 6 35 95 4 7 41 4 70 76 1
84 1 2 0 92 3 96 40 8 63 35 5 73 6 6 23 51 2 51 30 8 4 65 -1
2 25 3 91 47 7 84 31 1 19 31 5 28 27 6 19 43 5 23 68 2 39 43 6
88 94 1 80 98 4 18 52 3 98 95 -1 5 79 3 66 98 4 72 78 5 66 97 4
47 72 0 86 12 2 77 0 3 97 32 5 35 51 2 98 49 1 61 6 6 3 25 5
28 97 3 33 15 -1 81 14 1 97 0 5 8 29 1 13 79 1 16 14 4 75 65 0
30 26 1 16 29 5 52 9 2 15 95 5 28 76 5 13 26 -1 62 86 4 63 0 5
97 16 0 82 44 4 20 26 5 65 94 9 34 46 3 53 62 8 86 42 1 34 7 -1
34 69 0 15 84 4 24 82 0 51 16 1 43 37 0 61 2 4 60 88 9 20 41 6
24 28 1 8 62 3 70 96 1 66 11 2 15 87 5 32 90 1 93 33 5 80 94 6
13 54 0 44 75 2 38 51 2 73 11 3 68 81 4 83 99 4 35 14 6 69 46 4
72 91 -1 11 23 5 57 36 7 39 81 2 14 71 -1 18 96 4 83 64 1 97 0 5
74 35 2 90 57 1 49 29 4 90 92 2 29 49 1 22 40 6 43 55 4 66 25 1
53 60 3 20 57 4 87 83 3 21 26 2 5 27 7 28 69 1 27 98 2 63 21 2
83 16 0 23 82 0 11 35 4 63 8 5 68 95 3 60 68 5 9 73 3 87 54 4
57 81 7 29 96 6 42 80 4 9 55 5 54 66 5 59 81 5 73 1 5 71 13 8
99 22 3 55 61 -2 80 71 2 3 0 4 0 42 1 12 4 6 59 58 1 94 17 1
36 90 0 26 14 5 37 65 3 21 20 3 63 0 4 86 4 4 58 56 4 10 68 3
69 28 2 46 74 5 5 10 4 65 89 7 90 26 3 38 99 4 0 62 4 32 48 0
17 7 6 97 69 5 28 39 4 70 85 3 80 42 4 33 7 3 0 50 2 85 88 1
90 40 7 47 25 0 61 94 7 30 91 0 20 20 5 38 42 3 30 22 3 85 8 5
80 8 7 2 45 5 22 87 0 57 35 5 92 48 1 86 30 2 49 51 4 56 89 7
48 72 6 82 57 3 76 37 0 20 39 6 5 14 6 82 23 8 36 15 1 84 53 2
24 54 0 36 10 2 42 58 5 23 41 1 11 69 1 60 90 5 55 47 3 89 81 4
58 17 2 75 1 1 74 78 0 77 66 -1 69 61 3 33 36 3 6 99 6 16 60 5
70 67 2 86 8 4 77 66 -1 35 93 4 46 19 1 12 96 6 40 17 -2 74 50 1
2 8 3 51 25 2 90 47 4 76 34 5 95 63 1 31 51 0 20 97 1 88 61 -2
74 1 2 69 28 2 52 82 1 82 33 5 77 23 5 90 51 0 60 46 8 47 29 -1
28 49 3 28 89 6 76 63 0 90 84 6 59 36 4 84 71 2 38 0 7 39 90 1

75 50 3 26 50 4 28 78 6 79 58 5 44 34 5 11 77 3 57 36 6 34 72 -1
95 10 1 32 49 1 67 76 4 1 2 2 63 82 3 27 14 2 33 58 -1 54 69 4
59 27 3 1 13 6 18 60 3 35 92 1 43 12 -1 53 13 7 13 28 0 51 56 2
99 41 4 81 95 0 89 54 2 36 8 4 4 26 0 33 14 3 64 57 1 24 10 5
38 23 7 34 26 0 45 25 3 66 6 5 56 47 3 26 84 5 60 40 7 55 73 1
95 90 1 46 66 1 63 4 2 8 90 2 61 35 6 55 1 0 44 10 0 91 88 5
47 48 -3 81 56 0 29 51 4 34 49 -3 1 64 5 43 72 -1 11 33 4 56 40 4
59 36 6 10 28 2 97 75 6 24 56 1 25 85 8 19 71 0 80 24 5 96 67 8
3 30 9 46 86 1 94 45 2 52 8 5 98 5 0 44 30 2 47 7 2 0 78 5
7 58 3 79 7 5 75 10 3 4 8 2 26 2 2 85 6 -1 23 4 1 37 1 4
10 48 1 69 0 3 50 8 3 56 87 3 28 62 4 83 18 4 23 45 3 61 82 4
7 57 3 11 94 3 44 57 0 37 26 5 60 28 4 41 84 1 82 65 4 33 48 1
44 23 1 10 85 7 83 44 3 12 55 7 42 0 3 20 89 7 50 49 -1 37 42 2
14 24 3 54 9 4 64 54 2 19 16 3 60 99 8 6 63 5 99 5 -3 1 77 5
29 28 5 86 17 -1 79 83 8 89 37 2 47 53 -3 80 24 2 99 84 1 14 43 3
84 42 1 7 96 1 18 25 0 31 63 1 86 94 6 17 35 5 58 82 3 53 15 5
17 14 6 80 80 2 6 64 1 38 23 5 52 42 5 45 25 3 56 63 2 2 80 2
85 91 1 74 44 2 88 61 -3 38 41 5 79 48 4 66 38 3 50 42 3 59 88 5
35 96 4 6 98 5 13 35 1 54 62 8 10 2 3 32 40 4 86 71 2 4 89 6
46 39 4 82 50 5 43 86 3 19 92 4 76 57 0 75 63 1 19 25 4 77 9 2
80 95 3 31 0 4 20 46 6 56 29 3 28 24 1 48 44 9 18 72 5 72 47 5
39 19 3 9 48 4 70 28 -1 55 59 -1 29 31 6 42 88 6 6 68 7 8 69 2
3 87 2 7 11 5 26 3 -1 71 64 5 26 86 3 76 41 2 78 22 4 26 65 3
4 23 4 58 32 5 8 87 -1 30 94 5 51 72 4 92 44 5 80 22 4 49 98 6
28 76 7 87 2 3 38 58 5 58 68 4 46 28 0 74 58 2 14 62 6 56 54 0
15 34 6 60 35 5 46 64 2 55 3 0 33 42 2 83 0 4 16 46 8 37 21 7
3 87 3 42 43 4 77 10 1 86 70 4 93 17 0 28 24 3 66 58 5 66 93 7
86 9 1 35 46 2 95 2 3 16 96 5 92 74 7 6 60 4 3 6 5 17 34 8
34 52 2 93 18 2 4 5 2 13 92 0 44 87 2 61 56 4 26 48 1 50 54 3
96 9 1 11 26 1 2 12 1 52 57 3 46 61 -2 50 27 3 53 71 1 57 84 7
10 11 4 37 13 8 77 61 0 74 24 1 54 26 6 39 78 6 42 25 2 2 27 6
54 80 2 46 90 0 94 0 3 42 90 5 96 19 5 12 93 4 54 48 1 60 87 4
24 29 3 2 83 3 6 37 3 34 36 4 46 30 6 69 25 0 57 73 1 58 85 8
83 39 4 86 52 2 68 76 4 27 30 4 66 37 5 33 71 0 94 17 -2 55 38 -1
7 47 6 77 57 1 93 92 -1 57 78 6 60 47 6 44 26 0 11 92 3 68 77 3
60 71 1 87 79 0 87 86 7 13 63 0 24 9 3 74 66 2 29 27 7 60 71 5
73 35 7 24 3 2 45 15 2 88 3 -2 68 42 1 0 55 3 36 31 1 17 58 4
76 87 0 34 99 1 70 72 1 51 96 3 10 94 6 92 82 4 53 2 5 52 54 1
14 91 -1 38 60 4 28 88 6 45 22 5 37 44 6 65 96 2 47 6 2 97 50 6
2 3 2 50 55 3 5 21 4 91 11 3 88 39 2 23 36 2 58 26 4 65 91 1
20 90 5 22 88 6 57 90 4 95 40 6 40 97 -1 75 41 5 23 29 1 58 52 4
16 10 7 70 27 0 72 0 2 80 74 3 38 31 3 99 79 0 19 19 2 12 46 8
17 69 5 62 28 1 95 96 0 54 66 5 56 38 4 96 18 2 29 8 7 0 7 3
15 78 5 28 90 1 82 8 4 17 22 2 57 69 4 7 23 7 36 79 2 18 27 4
73 56 3 12 8 9 30 76 5 19 56 3 55 39 1 12 8 8 97 65 2 30 72 1
29 61 4 18 31 4 77 56 0 44 69 1 70 51 1 26 23 7 72 30 3 79 94 9

59 91 4 9 73 4 81 2 0 37 72 3 20 1 5 11 45 4 15 67 6 43 93 2
80 17 4 7 96 3 91 8 9 93 17 1 67 50 1 54 87 1 72 59 1 46 23 5
42 38 1 89 82 -2 63 14 4 67 21 1 65 65 5 6 58 2 20 25 1 88 79 -2
28 51 0 69 97 3 59 91 2 16 32 4 66 95 2 85 14 3 1 79 3 77 37 -1
18 57 2 62 45 3 93 25 0 35 46 2 49 5 4 53 73 2 39 39 2 91 77 6
64 30 6 55 8 2 51 26 6 77 40 1 81 33 3 89 69 3 46 70 3 94 23 1
75 14 3 4 5 1 95 21 6 69 76 5 29 80 3 99 57 -2 66 38 2 58 79 2
49 25 5 46 72 2 16 47 6 77 51 1 16 98 5 37 68 4 66 49 1 21 48 -2
69 66 1 51 77 4 5 26 6 30 72 -1 29 88 7 43 18 0 61 86 5 1 23 8
70 89 2 18 62 4 24 32 4 20 35 2 47 92 1 0 22 3 25 4 7 50 47 3
78 8 4 4 62 4 51 32 3 47 2 0 43 26 -1 58 98 6 50 98 6 69 98 7
12 76 8 47 78 0 49 8 8 36 64 1 51 15 5 54 14 5 20 10 2 70 68 1
7 71 3 89 40 5 93 4 3 53 51 4 8 52 -2 23 41 2 1 92 5 62 46 4
26 19 1 69 89 7 10 49 -1 1 90 4 42 83 6 12 88 7 11 96 1 19 71 2
91 24 4 30 87 1 36 65 0 52 34 7 89 96 2 96 97 5 96 92 5 99 56 1
58 67 4 22 86 4 74 77 2 71 8 5 66 96 2 80 49 -1 35 38 2 40 34 2
86 30 2 16 82 3 85 92 2 51 14 5 40 41 0 53 64 2 54 30 5 72 62 5
72 97 0 21 89 2 20 76 6 10 92 4 9 29 4 75 80 4 27 20 3 9 74 7
99 80 -1 38 52 3 25 76 7 51 98 5 63 70 5 57 32 4 34 41 1 3 16 4
38 96 2 63 5 1 3 57 3 26 95 2 58 20 3 92 23 6 64 38 3 63 48 -1
7 82 3 28 37 2 33 75 3 59 90 5 11 46 2 95 72 3 31 82 0 96 75 5
58 39 6 43 54 0 60 61 2 70 41 3 97 75 5 20 86 4 1 98 6 38 45 6
23 29 -2 28 25 0 8 35 1 79 30 7 46 90 0 47 13 3 37 10 4 51 83 6
7 36 5 9 26 6 35 49 2 91 77 5 23 38 -2 97 17 4 74 16 -3 65 15 5
49 53 5 56 56 5 1 15 1 97 76 5 38 11 3 8 2 -1 3 77 8 85 75 0
98 1 0 6 18 4 74 67 3 53 23 8 67 77 4 69 26 5 46 64 -1 24 72 1
79 27 6 25 64 6 36 15 0 15 21 7 9 95 4 57 1 4 60 20 3 31 90 3
25 88 6 83 12 -2 43 91 1 32 16 0 97 4 4 63 71 3 99 81 1 51 90 4
31 2 1 87 33 3 70 11 -1 68 46 3 91 89 5 92 73 4 58 70 7 91 85 7
97 37 1 99 88 1 53 71 5 28 59 0 45 81 5 50 1 3 19 92 1 80 37 3
9 47 5 94 39 2 3 88 5 93 87 0 25 41 6 12 21 7 32 67 7 15 69 3
44 89 1 14 21 3 9 30 4 75 76 4 31 80 5 71 25 3 4 3 -2 57 15 3
72 47 3 17 14 3 63 58 1 53 24 5 81 33 5 8 61 -1 74 92 2 28 15 0

```
V[0] = 0: 0
V[1] = 3: 0 (3) 55 (0) 1
V[2] = 4: 0 (2) 50 (3) 8 (-1) 2
V[3] = 3: 0 (1) 42 (4) 43 (-1) 26 (-1) 3
V[4] = 7: 0 (2) 50 (3) 8 (-2) 52 (1) 54 (3) 4
V[5] = 3: 0 (1) 42 (1) 38 (4) 99 (-3) 5
V[6] = 7: 0 (2) 50 (3) 47 (2) 6
V[7] = 3: 0 (3) 7
V[8] = 5: 0 (2) 50 (3) 8
V[9] = 5: 0 (2) 50 (3) 8 (-2) 52 (2) 9
V[10] = 5: 0 (2) 50 (3) 8 (-1) 2 (1) 10
V[11] = 5: 0 (1) 42 (1) 38 (3) 11
V[12] = 4: 0 (1) 42 (4) 43 (-1) 12
V[13] = 8: 0 (2) 50 (3) 47 (3) 13
V[14] = 7: 0 (2) 50 (3) 27 (2) 14
V[15] = 4: 0 (3) 55 (0) 1 (1) 15
V[16] = 4: 0 (1) 42 (4) 43 (-1) 72 (0) 97 (0) 16
V[17] = 5: 0 (3) 55 (-1) 59 (1) 58 (2) 17
V[18] = 5: 0 (1) 42 (4) 43 (0) 18
V[19] = 5: 0 (1) 42 (4) 43 (-1) 26 (1) 19
V[20] = 6: 0 (3) 55 (-1) 59 (1) 58 (3) 20
V[21] = 7: 0 (1) 42 (1) 38 (4) 99 (-3) 5 (4) 21
V[22] = 3: 0 (3) 22
V[23] = 7: 0 (1) 42 (1) 38 (5) 23
V[24] = 6: 0 (4) 62 (1) 28 (1) 24
V[25] = 3: 0 (1) 42 (2) 25
V[26] = 4: 0 (1) 42 (4) 43 (-1) 26
V[27] = 5: 0 (2) 50 (3) 27
V[28] = 5: 0 (4) 62 (1) 28
V[29] = 4: 0 (2) 50 (3) 47 (-1) 29
V[30] = 7: 0 (1) 42 (4) 43 (-1) 72 (3) 30
V[31] = 5: 0 (1) 42 (1) 38 (3) 31
V[32] = 6: 0 (3) 92 (3) 32
V[33] = 6: 0 (3) 22 (0) 87 (3) 33
V[34] = 10: 0 (2) 50 (3) 8 (-2) 52 (7) 34
V[35] = 6: 0 (2) 50 (3) 8 (1) 35
V[36] = 6: 0 (3) 55 (-1) 59 (4) 36
V[37] = 5: 0 (1) 42 (4) 43 (0) 37
V[38] = 2: 0 (1) 42 (1) 38
V[39] = 4: 0 (3) 55 (1) 39
V[40] = 9: 0 (3) 22 (6) 40
V[41] = 7: 0 (3) 7 (4) 41
V[42] = 1: 0 (1) 42
V[43] = 5: 0 (1) 42 (4) 43
V[44] = 8: 0 (3) 92 (5) 44
V[45] = 7: 0 (4) 62 (3) 45
```

```
V[46] = 7: 0 (1) 42 (1) 38 (3) 11 (2) 46
V[47] = 5: 0 (2) 50 (3) 47
V[48] = 2: 0 (2) 50 (3) 47 (-3) 48
V[49] = 1: 0 (2) 50 (-1) 49
V[50] = 2: 0 (2) 50
V[51] = 4: 0 (1) 42 (1) 38 (2) 51
V[52] = 3: 0 (2) 50 (3) 8 (-2) 52
V[53] = 2: 0 (2) 50 (3) 47 (-3) 53
V[54] = 4: 0 (2) 50 (3) 8 (-2) 52 (1) 54
V[55] = 3: 0 (3) 55
V[56] = 5: 0 (3) 55 (-2) 61 (4) 56
V[57] = 4: 0 (1) 42 (1) 38 (4) 99 (-2) 57
V[58] = 3: 0 (3) 55 (-1) 59 (1) 58
V[59] = 2: 0 (3) 55 (-1) 59
V[60] = 5: 0 (1) 42 (4) 43 (-1) 26 (1) 19 (0) 60
V[61] = 1: 0 (3) 55 (-2) 61
V[62] = 4: 0 (4) 62
V[63] = 6: 0 (1) 42 (1) 38 (3) 31 (1) 63
V[64] = 4: 0 (2) 50 (3) 47 (-3) 53 (2) 64
V[65] = 6: 0 (2) 50 (3) 8 (-2) 52 (1) 54 (3) 4 (-1) 65
V[66] = 6: 0 (1) 42 (1) 38 (3) 69 (1) 66
V[67] = 7: 0 (3) 55 (-1) 59 (1) 58 (4) 67
V[68] = 7: 0 (2) 50 (3) 8 (-1) 2 (3) 68
V[69] = 5: 0 (1) 42 (1) 38 (3) 69
V[70] = 10: 0 (3) 55 (-2) 61 (5) 86 (4) 70
V[71] = 3: 0 (2) 50 (3) 8 (-2) 52 (0) 71
V[72] = 4: 0 (1) 42 (4) 43 (-1) 72
V[73] = 4: 0 (2) 50 (3) 8 (-1) 2 (1) 10 (-1) 73
V[74] = 8: 0 (1) 42 (4) 80 (3) 74
V[75] = 9: 0 (1) 42 (1) 38 (2) 96 (5) 75
V[76] = 9: 0 (1) 42 (4) 43 (-1) 72 (0) 97 (5) 76
V[77] = 8: 0 (1) 42 (1) 38 (3) 11 (3) 77
V[78] = 5: 0 (5) 78
V[79] = 3: 0 (3) 22 (0) 87 (0) 79
V[80] = 5: 0 (1) 42 (4) 80
V[81] = 6: 0 (3) 55 (1) 39 (2) 81
V[82] = 4: 0 (2) 50 (3) 8 (-2) 52 (1) 82
V[83] = 6: 0 (3) 22 (0) 87 (3) 83
V[84] = 7: 0 (1) 42 (1) 38 (4) 99 (1) 84
V[85] = 10: 0 (1) 42 (4) 43 (-1) 72 (-1) 91 (7) 85
V[86] = 6: 0 (3) 55 (-2) 61 (5) 86
V[87] = 3: 0 (3) 22 (0) 87
V[88] = 7: 0 (1) 42 (6) 88
V[89] = 8: 0 (1) 42 (4) 43 (-1) 72 (-1) 91 (5) 89
V[90] = 5: 0 (3) 55 (1) 39 (1) 90
V[91] = 3: 0 (1) 42 (4) 43 (-1) 72 (-1) 91
```

```
V[92] = 3: 0 (3) 92
V[93] = 7: 0 (1) 42 (4) 43 (2) 93
V[94] = 8: 0 (1) 42 (6) 88 (1) 94
V[95] = 6: 0 (1) 42 (4) 43 (-1) 26 (2) 95
V[96] = 4: 0 (1) 42 (1) 38 (2) 96
V[97] = 4: 0 (1) 42 (4) 43 (-1) 72 (0) 97
V[98] = 7: 0 (2) 50 (-1) 49 (6) 98
V[99] = 6: 0 (1) 42 (1) 38 (4) 99
relax executed 2120 times.
```