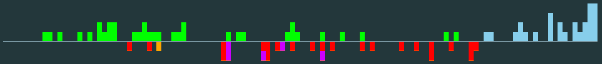
B: Boggle Sort

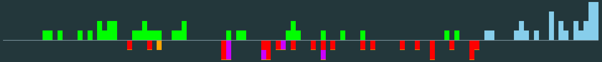
Problem author: Thore Husfeldt



Problem: Turn 16 six-sided dice in given order so that the top-facing sides are alphabetically ordered using as few turns as possible.

B: Boggle Sort

Problem author: Thore Husfeldt

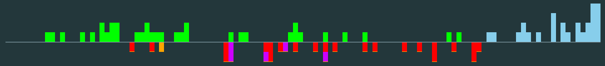


Problem: Turn 16 six-sided dice in given order so that the top-facing sides are alphabetically ordered using as few turns as possible.

Naive solution: There are $6^{16} > 2 \cdot 10^{12}$ possible turns; unoptimised brute-force will not work.

B: Boggle Sort

Problem author: Thore Husfeldt



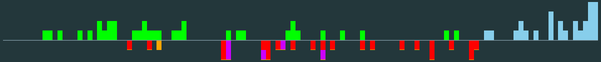
Problem: Turn 16 six-sided dice in given order so that the top-facing sides are alphabetically ordered using as few turns as possible.

Naive solution: There are $6^{16} > 2 \cdot 10^{12}$ possible turns; unoptimised brute-force will not work.

Greedy: Greedily turning the dice to the alphabetically earliest letter does not work; it (helpfully) fails on Sample 1.

B: Boggle Sort

Problem author: Thore Husfeldt

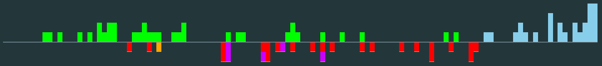


Problem: Turn 16 six-sided dice in given order so that the top-facing sides are alphabetically ordered using as few turns as possible.

Naive solution: There are $6^{16} > 2 \cdot 10^{12}$ possible turns; unoptimised brute-force will not work.

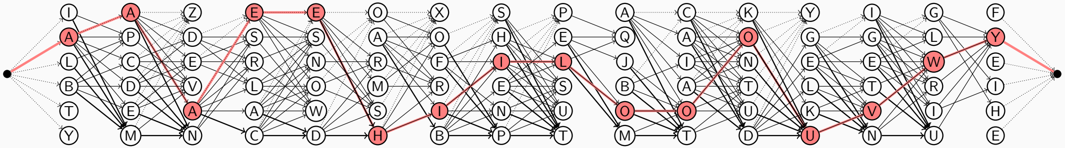
Greedy: Greedily turning the dice to the alphabetically earliest letter does not work; it (helpfully) fails on Sample 1.

Brute-force-y sol.: Among the 4 sideways faces, it is optimal to take the alphabetically earliest face that still fits. Thus, there are really only 3 different choices per die: keep, turn on smallest side, turn bottom up. This gives $3^{16} = 43\,046\,721$ choices, which maybe can be systematically checked.



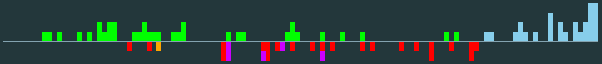
Graph-y solution: Create digraph with vertex set 6×16; connect (r, c) to $(r', c+1)$ if the character in (line l, column c) precedes the character in (line r', column $c+1$) in the alphabet. The weight is 0 if $r = 1$ (dotted), 2 if $r = 6$ (fat), and 1 otherwise. Connect s to $(r, 1)$ and $(r, 16)$ to t. Then a minimum-weight s,t-path is the solution.

Optimal solution for Sample 1:



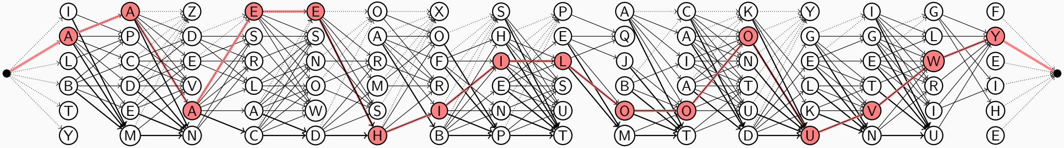
B: Boggle Sort

Problem author: Thore Husfeldt



Graph-y solution: Create digraph with vertex set 6×16; connect (r, c) to $(r', c+1)$ if the character in (line l, column c) precedes the character in (line r', column $c+1$) in the alphabet. The weight is 0 if $r = 1$ (dotted), 2 if $r = 6$ (fat), and 1 otherwise. Connect s to $(r, 1)$ and $(r, 16)$ to t. Then a minimum-weight s, t-path is the solution.

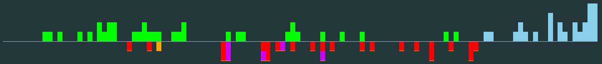
Optimal solution for Sample 1:



Pitfall: There should be no edge from `Q` to `T`, even though `Q<T` (because it should be treated as `QU` and `U>T`). Ignoring this leads (helpfully) to a wrong answer on Sample 1.

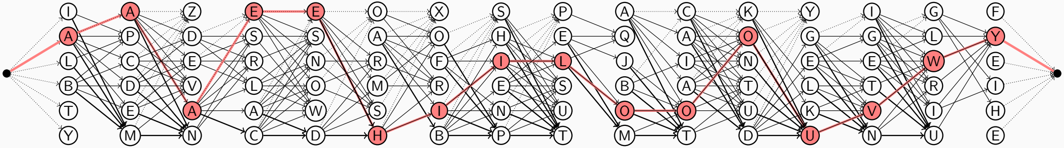
B: Boggle Sort

Problem author: Thore Husfeldt



Graph-y solution: Create digraph with vertex set 6×16; connect (r, c) to $(r', c+1)$ if the character in (line l, column c) precedes the character in (line r', column $c+1$) in the alphabet. The weight is 0 if $r = 1$ (dotted), 2 if $r = 6$ (fat), and 1 otherwise. Connect s to $(r, 1)$ and $(r, 16)$ to t. Then a minimum-weight s, t-path is the solution.

Optimal solution for Sample 1:

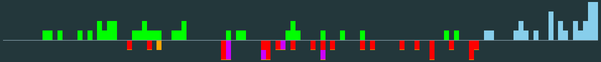


Pitfall: There should be no edge from `Q` to `T`, even though `Q<T` (because it should be treated as `QU` and `U>T`). Ignoring this leads (helpfully) to a wrong answer on Sample 1.

Running time: Using Dijkstra's algorithm, time $\mathcal{O}(rc \log rc)$ for c dice with r faces. But graph is acyclic, so actually $\mathcal{O}(rc)$.

B: Boggle Sort

Problem author: Thore Husfeldt



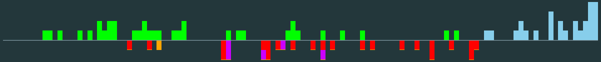
DP: Compute, for $1 \leq i \leq 16$, and each letter x, the smallest number $f(i,x)$ of turns needed to bring the first i dice into nondecreasing order such that the ith die shows x. Then, if x appears on the ith die, we have the general case

$$f(i,x) = \max_{y \leq x} f(i-1,y)$$

where y ranges over all letters appearing on die $(i-1)$. (Remember the `Q`=`QU` pitfall.)

B: Boggle Sort

Problem author: Thore Husfeldt



DP: Compute, for $1 \le i \le 16$, and each letter x, the smallest number $f(i,x)$ of turns needed to bring the first i dice into nondecreasing order such that the ith die shows x. Then, if x appears on the ith die, we have the general case

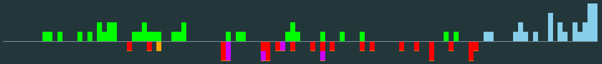
$$f(i,x) = \max_{y \le x} f(i-1, y)$$

where y ranges over all letters appearing on die $(i-1)$. (Remember the `Q`=`QU` pitfall.)

Running time: $\mathcal{O}(rc)$ for c dice with r faces.

B: Boggle Sort

Problem author: Thore Husfeldt



DP: Compute, for $1 \leq i \leq 16$, and each letter x, the smallest number $f(i,x)$ of turns needed to bring the first i dice into nondecreasing order such that the ith die shows x. Then, if x appears on the ith die, we have the general case

$$f(i,x) = \max_{y \leq x} f(i-1,y)$$

where y ranges over all letters appearing on die $(i-1)$. (Remember the $\texttt{Q}$=$\texttt{QU}$ pitfall.)

Running time: $\mathcal{O}(rc)$ for c dice with r faces.

Statistics: 89 submissions, 35 accepted, 26 unknown