Periodic table

1 second, 64 megabytes

Mr. Kemish entered the chemistry exam with absolute confidence. However, the exam, made by teacher Nathapol Jaesorn, was so hard that Mr. Kemish started contemplating his life decisions. While thinking that this exam was supposed to be about gas laws and the rate of flow, he wondered why it was instead filled with hypothetical elements and ridiculously complex numbers that would take forever to calculate—especially since calculators were, of course, not allowed.

Faced with these questions, Mr. Kemish decided not to bother solving them. Instead, he began daydreaming about the periodic table. Soon after, he started hallucinating, and suddenly, he found himself teleported into a gigantic periodic table.

He discovered that he had appeared on element X and needed to move from element X to element Y. He could move one tile at a time, either within the same row or to an adjacent column. Every time he moves from a tile, that tile falls off and can no longer be stepped on again. Each move would cost him one energy point. However, some elements had special properties:

- Moving to a **radioactive** element (listed below) does not consume energy.

- Moving from a **metalloid** element allows diagonal movement, still costing only one energy point. (Only element with atomic number of 5, 14, 32, 33, 51, 52)

- **Alkali metals** (Group 1) and **noble gases** (Group 18) in the same row can be traversed as if they were adjacent tiles.

- The **f-block elements** are ignored entirely, except for **Lutetium (Lu) and Lawrencium (Lr)**, which replace the entire block.

Mr. Kemish want you the programmer to find the least energy needed for a path from element X and element Y

Periodic table

input:

1st row element X in atomic number

2ndrow element Y in atomic number

output:

1st row the least energy needed to traverse from element X to element Y (-1 if X to Y is impossible)

Input	Output
21 40	2

Table vector (C++)

```
std::vector<std::vector<int>> periodicTable = {

  {  1,  -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,  2 },{  3,   4, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,  5,  6,  7,  8,  9, 10 },

  { 11,  12, -1, -1, -1, -1, -1, -1, -1, -1, -1, 13, 14, 15, 16, 17, 18 },{ 19,  20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36 },

  { 37,  38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54 },{ 55, 56, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86 },

  { 87, 88, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118 }

};

std::vector<int> radioactive = {

  43,  61,84,  85,  86,87,  88,  89,  90,  91,  92,  93,  94,  95,  96,  97,  98,  99,100, 101, 102, 103,104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118

};
```

Author:	Jump