OkCupid Matching Challenge

OkCupid is an online dating site that has an interesting matching algorithm. You're going to implement it. See the attached PDF for OkCupid's description of their algorithm. Write a program in the language of your choice that reads a set of user profiles from a file (represented in JSON; see included example input file) and writes the top 10 matches for each user profile to stdout (also in JSON, see below), sorted in rank order.

When you've completed the challenge, email me your solution along with the output from your solution.

Important

- The guts of the algorithm is more important to me than the JSON processing (it's not a JSON processing quiz). Consider that when prioritizing if you're short on time or unfamiliar with JSON processing libraries in your language of choice.
- Your primary goal is to produce a quality implementation of the algorithm. Code should be self-documenting and easy to read.
- Being time and space efficient is not as important as clear, understandable code here. But think about how space/time efficient your simple solution is. Think about the data structures and operations you're using.
- There are several optimization opportunities in the implementation. Implement the optimizations if you have time. If you don't have time, feel free to followup with an optimized version or some notes on how you might make the implementation more time or space efficient.

Input Format

- The 'importance' field is in the range [0,4] and is the index into an array that defines the weights as described in the OkCupid doc, e.g.:
  `IMPORTANT_POINTS = [0, 1, 10, 50, 250]`
- Answers are always in the range [0,3]
- The size of the acceptable answer set is between 1 and 3. 0 and 4 are nonsensical.

Example Output Format:

```
{
  "results": [
    {
      "profileId": 0,
      "matches": [
        {
          "profileId": 2,
          "score": 0.87
        },
        {
          "profileId": 1,
```

```
          "score": 0.65
        },
      ]
    },
    {
      "profileId": 1,
      "matches": [
        {
          "profileId": 0,
          "score": 0.65
        },
        {
          "profileId": 2,
          "score": 0.5
        }
      ]
    }
  ]
}
```