

CSE 142, Winter 2008
Programming Assignment #7: Personality Test (20 points)
Due: Tuesday, March 4, 2008, 4:00 PM

This assignment focuses on arrays and file processing. Turn in a file named `PersonalityTest.java`. You will also need input file `personality.txt` from the course web site. The assignment involves processing data from a personality test. There is a link on the course web site where you can take the personality test yourself. Student answers will be included in a data file distributed to the whole class.

Background Information:

The Keirsey Temperament Sorter (<http://www.keirsey.com/>) is a personality test that involves answering 70 questions. Each question has two answer choices, which we will refer to as the "A" and "B" answer. The person taking the test is allowed to leave a question blank, in which case the answer will be recorded with a dash ('-').

The Keirsey test measures four independent dimensions of personality:

1. Extrovert versus Introvert (E vs I): what energizes you
2. Sensation versus intuition (S vs N): what you focus on
3. Thinking versus Feeling (T vs F): how you interpret what you focus on
4. Judging versus Perceiving (J vs P): how you approach life

Individuals are categorized as being on one side or the other for each dimension. The corresponding letters are put together to form a personality type. For example, if you are an Extrovert, intuitive, Thinking, Perceiving person then you are referred to as an ENTP. The "A" answers correspond to E, S, T, and J (the left-hand choices above). The "B" answers correspond to I, N, F, and P (the right-hand choices above). For each dimension, we determine a percentage of B answers the user gave for that dimension between 0 and 100, to indicate whether the person is closer to the "A" or "B" side.

Suppose that someone's answers are as follows (These are the answers given by "Betty Boop" later in this document).

Dimension	# of "A" answers	# of "B" answers	% of "B" answers	Result
Extrovert/Introvert	1	9	90%	I
Sensing/iNtuition	17	3	15%	S
Thinking/Feeling	18	2	10%	T
Judging/Perceiving	18	2	10%	J

We add up how many of each type of answer we got for each dimension. Then we compute the percentage of B answers for each dimension. Then we assign letters based on which side the person ends up on for each dimension. In the Extrovert/Introvert dimension, for example, Betty gave 9 "B" answers out of 10 total (90%), which means she is on the B side, which is "Introvert" or I. The overall percentages are (90, 15, 10, 10) which works out to a personality type of ISTJ.

Mechanics of the Personality Test:

Suppose that "Betty Boop" gave the following answers for the 70 questions, in order from 1 to 70:

BABAAAAABAAAAAABAAAAABAAAAABAAAAABAAAAABAAAAABAAAAABAAAAABAAAAABAAAAABAAAAABAAAAABAAAAA

The questions are organized into 10 groups of 7 questions, with the following repeating pattern in each group:

1. The first one question in each group is an Introvert/Extrovert question (questions 1, 8, 15, 22, etc).
2. The next two questions are for Sensing/iNtuition (questions 2 and 3, 9 and 10, 16 and 17, 23 and 24, etc).
3. The next two questions are for Thinking/Feeling (questions 4 and 5, 11 and 12, 18 and 19, 25 and 26, etc).
4. The next two questions are for Judging/Perceiving (questions 6 and 7, 13 and 14, 20 and 21, 27 and 28, etc).

In other words, if we consider the I/E to be dimension 1, the S/N to be dimension 2, the T/F to be dimension 3, and the J/P to be dimension 4, the map of questions to their respective dimensions would look like this (*spaces added for emphasis*):

1223344	1223344	1223344	1223344	1223344	1223344	1223344	1223344	1223344	1223344
BABAAAA	BAAAAAA	ABAAAAAB	BAAAAAA	BAAAAABA	BAABAAA	BABABAA	BAAAAAA	BAAAAAA	BAAAAAA

Notice that there are half as many Introvert/Extrovert questions as there are for the other three dimensions.

Program Behavior:

The personality data consists of line pairs, one per person. The first line has the person's name, and the second has the person's 70 answers ('A', 'B' or '-'). The A or B can be upper or lowercase. A dash is a question that was skipped.

```
Betty Boop
BABAAAABAAAAAABAAAABAAAAAABAAAAABABAABAAAABABABAABAAAAAABAAAAAABAAAAA
Bugs Bunny
aabaabbabbbbaaaabaaaabaaaababbbbaabaaaabaabbbbabaaaabaabaaaaabbaaaabb
Han Solo
BA-ABABBB-bbbaababaaaabbaaabbbaabbabABBAABABBAABABAAAABBABAAAABBABAAB
```

Your output begins with an **introduction of your own creation**, such as an explanation of the personality test and/or a fact about your own personality type. Next your program asks for the input file name to process. You may assume that the user will type a name of a file that exists and can be read. Then the program prompts the user for an output file name; if this file already exists, your program will overwrite its contents.

Each pair of lines from the input file is turned into a group of lines in the output file with the name, count of As and Bs for each dimension, % Bs for each dimension (rounded to the nearest percent), and personality type. **If the person has the same number of As and Bs for a dimension, give them an "X"** (as with Han Solo). You may assume that the input file has no errors and that nobody has skipped all questions for a dimension. Reproduce the following output format:

Log of execution (user input underlined):

```
<< your introduction message here >>
```

```
Input file name: personality.txt
Output file name: output.txt
```

Resulting output file output.txt:

```
Betty Boop
Answers: 1A-9B 17A-3B 18A-2B 18A-2B
Percent B: [90, 15, 10, 10]
Type: ISTJ
```

```
Bugs Bunny
Answers: 8A-2B 11A-9B 17A-3B 9A-11B
Percent B: [20, 45, 15, 55]
Type: ESTP
```

```
Han Solo
Answers: 2A-8B 9A-9B 11A-9B 15A-5B
Percent B: [80, 50, 45, 25]
Type: IXTJ
```

Implementation Guidelines:

Use a `PrintStream` for file output, as described in the book (6.4).

Use arrays to store data from the input file and transform it from one form to another. The transformation of the data into a personality type can be summarized by the following (using Han Solo's data):

<pre>Answers: "BA-ABABBB- bbbaababaaaabbaaabbbaabbabABBAABABBAABABAAAABBABAAAABBABAAB"</pre>	<pre>↓ <u>What is computed</u> A count: {2, 9, 11, 15} B count: {8, 9, 9, 5} ↓ B percent: {80, 50, 45, 25} ↓ Type: "IXTJ"</pre>	<pre><u>Resulting output</u> Answers: 2A-8B 9A-9B 11A-9B 15A-5B Percent B: [80, 50, 45, 25] Type: IXTJ</pre>
---	--	--

Stylistic Guidelines:

Use the following two **class constants** in your program:

- the **number of dimensions** in the test, as an `int` (with a value of 4)
It will not be possible to change this constant and have the program adjust, but it makes the code more readable.
- a **DEBUG flag**, as a `boolean`
If this flag is set to `true`, your program will not prompt the user for input. It will instead use `personality.txt` as the input file name, and will write all the personality output to the console (`System.out`) instead of to a file.
(In fact, we suggest you initially write your program to behave this way, and only later add the prompting code.)

We will grade your method structure strictly. Write **at least four nontrivial methods besides main**. Your `main` should be a concise summary of the program, and no one method should perform too much of the overall task. Use parameters and returns, including passing and returning arrays. We will also check strictly for redundancy. Remove redundant code using `if/else` factoring and methods. For reference, our solution is 120-150 lines long and has 7 methods besides `main`.

You are limited to features in Chapters 1 through 7. Follow past style guidelines about indentation, names, variables, types, line lengths, and comments (at the beginning of your program, on each method, and complex sections of code).