Data Structures

Project #2
CALENDAR SCHEDULING

REQUIREMENTS:

A certain charity recruits twelve volunteers each year to spearhead its fundraising efforts. Each volunteer is assigned one month to take the lead organizing events. As these are *volunteers*, the charity does it best to accommodate everyone's schedule as much as possible. Before the year begins, each volunteer is asked to give his or her top three choices for a month to serve. These preferences are likely to overlap. For example:

	Susan	*John*	*Mary*
first choice	December	December	March
second choice	February	March	May
third choice	April	February	April

Obviously both Susan and John cannot be assigned December; one of them will need to be assigned a different month. The best possible assignment for the three volunteers (ignoring the other nine for the moment) would be

> Susan = February   John = December   Mary = March

as John and Mary get their first choice and Susan receives her second. For twelve volunteers, the task above is humanly unmanageable, and so the charity has contacted you to automate this process for them.

The charity proposes that the following point system be used when determining the desirability of a particular month assignment to its volunteers:

3 points	for each volunteer assigned his or her first choice
2 points	for each volunteer assigned his or her second choice
1 point	for each volunteer assigned his or her third choice
0 points	for each volunteer not assigned any of his or her choices

Thus, each of the 12! possible assignments of months to volunteers will have a desirability between 0 and 36. Your task is to find the best calendar assignment.

SPECIFICATIONS:

Design and implement a program to find the best possible calendar assignment for the charity's volunteers. **Your authored code MUST be a client of the UnsortedType class presented in the textbook.**

INPUT:

- The names of the twelve volunteers followed by their month preferences. The information for each volunteer will be listed on a single line in the file "volunteers.dat" in the following format:

   <name> <first-choice> <second-choice> <third-choice>

   where *<name>* is a string of up to 20 characters, and *<first-choice>*, *<second-choice>*, and *<third-choice>* are integers between 1 and 12.

OUTPUT:

- A listing of the best possible calendar assignments found so far as your program progresses through the permutations.

EXAMPLE OUTPUT:

SCORE: 12

January	Yuen
February	Bill
March	Steve
April	Alice
May	Susan
June	Julio
July	Gary
August	Mike
September	Mary
October	Rich
November	Shilpa
December	Ann

SCORE: 14

January	Steve
February	Julio
March	Shilpa
…	

DELIVERABLES:

- Source code
- Any data files you create and use in your test plan
- a sample run of your program with the best assignment of months for the 12 volunteers
- an implemented test plan