

Bonus assignment 12: Pakjesavond: first come, first serve

Course ‘Imperative Programming’ (IPC031)

1 Assignment

In the mandatory assignment we assumed that St Nicholas never runs out of gifts. As a consequence, the order of selecting persons for whom gifts are selected is irrelevant. Under more realistic circumstances, the number of gifts is bounded. On Brightspace, in “assignment-12-bonus-files.zip” you may find a number of text files:

- “giftstore_bounded.txt”: this file contains information about the bounded gift store. Each line begins with the number of gifts of that type (an integer value, which may be zero), the price of a gift (an integer value, representing the price in cents), followed by the name of the gift (which may contain white-space characters, *e.g.*, “Playstation 5”). This file contains all possible gifts by St Nicholas.
- “Andrew.txt”, “Belle.txt”, “Chris.txt”, “Desiree.txt”, “Edward.txt”, “Fabienne.txt”: these files are wish lists. In a wish list file, the first line is a budget (an integer value, representing the budget in cents). All remaining lines are gift names (a single gift name per line).

As a consequence, the recursive function `gifts` must also keep track of which gifts are removed from the gift store.

Part 1: bounded gifts

Adjust your data structures, functions that read the bounded gift store file, and recursive function `gifts` to be able to deal with the above restriction.

Now that gifts are removed from the gift store, granting the same wish list multiple times in a row may produce different results. If we start with a fresh gift store loaded from “giftstore_bounded.txt”, and grant a wish list five times in a row, we expect the following remaining budgets:

- Andrew: 251, 251, 251, 251, and 251 cents
- Belle: 301, 301, 301, 301, and 351 cents
- Chris: 3, 106, 106, 106, and 106 cents
- Desiree: 3, 5, 5, 7, and 8 cents
- Edward: 3, 3, 3, 3, and 3 cents
- Fabienne: 303, 903, 3904, 3904, and 5103 cents

Note that each person starts with a fresh gift store.

Part 2: queue before St Nicholas

Design and implement a console I/O application that allows the user to enter the name of a person. When a known person’s name has been entered (Andrew, Belle, Chris, Desiree, Edward, Fabienne), the program must use the adjusted `gifts` function to compute and show what that person gets from St. Nicholas. Additionally, these gifts are removed from the gift store. When the name is entered of an unknown person, the program should ask the user if she wants to stop or enter another name. If the user confirms that she stops, the program should terminate. Otherwise, the program should allow the user to enter a new name.

2 Products

As product-to-deliver upload to Brightspace “main.cpp” that you have created with your solution regarding the bonus assignment, and “main_test.cpp” with your adjusted tests.

Deadline

Bonus assignment: Monday December 11, 2023, 15:30h