TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Department of Computer Systems
Study Centre for Computer Systems

Sigita Ševceva 204434IV

IAX0583 Programming I

TASK: Sale of TV sets

Homework II

Supervisor Risto Heinsar

Engineer

Tallinn 2020

Declaration of originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication. All works and major viewpoints of the other authors, data from other sources of literature and elsewhere used for writing this paper have been referenced.

Name: S. Ševceva

Date: 24.11.2020

Table of contents

Declaration of originality **2**
Table of contents **3**
Task description **4**
Program description **5**
Screenshots **7**

Task description

Sale of TV-sets

We have 2 arrays. The first one contains the names of the manufacturers and the second one contains the prices. The maximum size of arrays is 15. The user enters the discount in percentages (between 10 and 60).

Your task is to calculate and display the discount prices for each manufacturer. You may create a third array for the discount prices.

Implement at least these functions:

- reading the arrays
- reading the discount percentage (you must check that the discount is within permitted limits)
- displaying the manufacturer names and prices (this function gets called twice, once for original prices, then for discount prices)
- calculating new prices with the formula: new price = old price * (1 - (discount percentage / 100.0))

Example:

The code should work when presented with either of the following files. It should also work with any other file generated by the same principle. The first number(s) in the file will be array member count(s), followed by that exact number of array members. If there is any additional input, it will be after the array members.

021

022

```
Old prices

Manufacturer                    Price
Panasonic                        9999
Philips                          6999
Samsung                          3999
Sharp                            5799
Thomson                          3299

New prices

Manufacturer                    Price
Panasonic                        7499
Philips                          5249
Samsung                          2999
Sharp                            4349
Thomson                          2474
```

Program description

The C language program is compiled using standard C libraries `<stdio.h>` and `<stdlib.h>.` The program is designed show TV-set manufacturer products and their respective prices, as its purpose is to calculate discounts upon the user or standard input redirection stream defined discount percentage and show the newly generated prices with the respective brand names in the form of a table. Input data of brand names and prices will be stored in two arrays (a character string array `brandArray` for brand names, the other for prices `priceArray` as floating point numbers) where, by command line function stdin, it would redirect input from a plain text data file containing different product names and prices. A maximum limit of up to 20-character length for each entry of the array is set by defining a `MAX_ENTRY_LEN`. A third array (`discountArray`) is defined to hold the discount applied prices. For preprocessing, three other macros have been defined: `MIN_DISC` of the minimum possible amount of discount i.e. 10 percent, `MAX_DISC` with the amount of 60 percent, as well as a table layout customizing macro `F_LEN` of size 13, that are later used to align the table elements. After macros, four user created function prototypes are declared that will be discussed later in the description.

The program workflow starts with the main function and reading in the first integer input which is taken as the number of elements of the product array, the value is stored after which two initial arrays are defined for brand names and prices with defined length of the arrays. Next, `ReadArrays()` function is executed to read the aforementioned values. Next, the table of old prices is displayed by using `ShowBrandsAndPrices()` function. Next, the third array `discountArr()` is defined with the same length as the previous arrays. A loop counter `i` is introduced and an integer variable `discount` for storing the discount percentage input. This value is taken in from the stdin using `GetIntInRange()` function, and stored by assigning it to `discount`. Then, discounted prices are calculated by `DiscountCalc()` function. A `for loop` is executed to store each calculated value to the discount array. Finally, the new prices are displayed calling the same function as for the old prices, changing one input parameter. At the end, the discount amount is displayed as in the examples.

`ReadArray()` function is void and reads in two types of arrays, a string array and integer array, repeats until the end of the length of the array. Three parameters are taken in:

string array, integer array and length of type integer. A loop counter `i` is introduced to loop through the arrays. The reading is done by `scanf` function in both cases.

`GetIntInRange()` function of type integer asks the user for an integer in between the minimum and maximum values, which are passed as arguments. One additional variable of type integer is used (`num`) which is later returned. This function is used in to get the discount percentage amount between the allowed 10 `(MIN_DISC)` and 60 `(MAX_DISC)`. The function returns an integer if it is in the specified limits, if not – the function brings a warning message to the user which is implemented in a `do-while` loop.

`DiscountCalc()` function of type int calculates the new price according to the calculation formula set in the task. Its parameters are the previously determined discount and the regular price; it returns the new price of type integer as in the examples. It has additional variables: an integer value of the `newPrice`, an integer value of one and floating value of 100.0 (these can also be defined as macros). This function is then used in a `for` loop to calculate the new price for each of the regular `priceArray` elements.

`ShowBrandsAndPrices()` function of type void takes in three parameters – the array of brand names, the array of prices, and the length of the array. A loop counter `i` is introduced to loop through the arrays. It firstly prints out the header of the table by preprocessing macro F_LEN size 13 for layout, and then loops through each element until the array length to display the table of products and prices.

Regarding any special situation handling, all of the input values (products and prices) will always be shown regarding of their count only in case the total number of their amount is provided at the beginning of the stream redirection input document. But for this reason, the program will return garbage values if the specified number of products is larger than the products themselves; therefore, this number can only be either smaller or equal to the number of products.

To avoid random text input and table layout disruption I have set the maximum length of each product/brand name to be of 20 characters (19 + \0).

In case when the document specifies a discount amount outside of the set limits of 10-60, the warning message will appear; however, in order for it to not loop over and crash the program, the loop will be broken which is due to the `break` statement, and the program will only show the old prices.

Screenshots

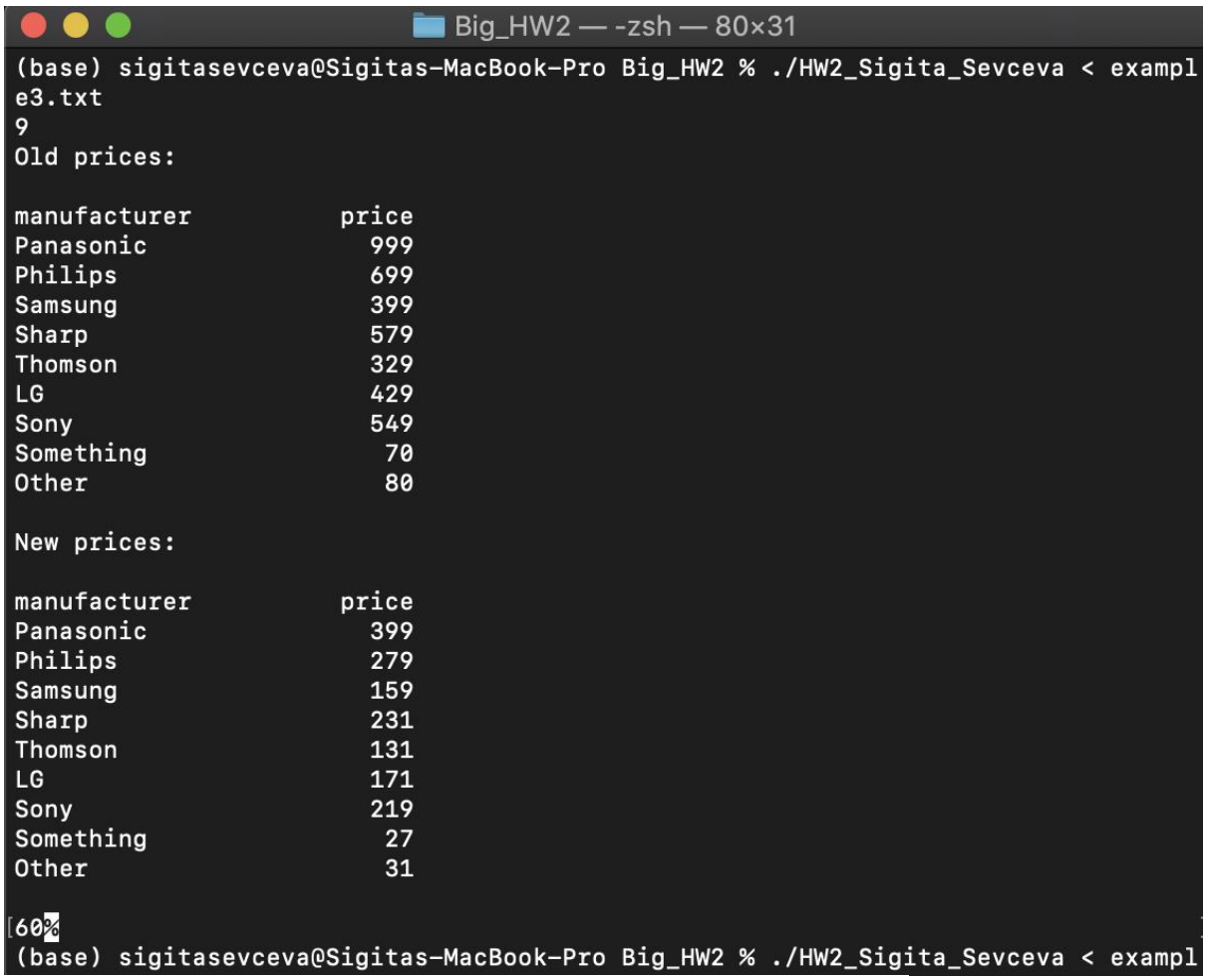

Figure 1: Program functioning properly

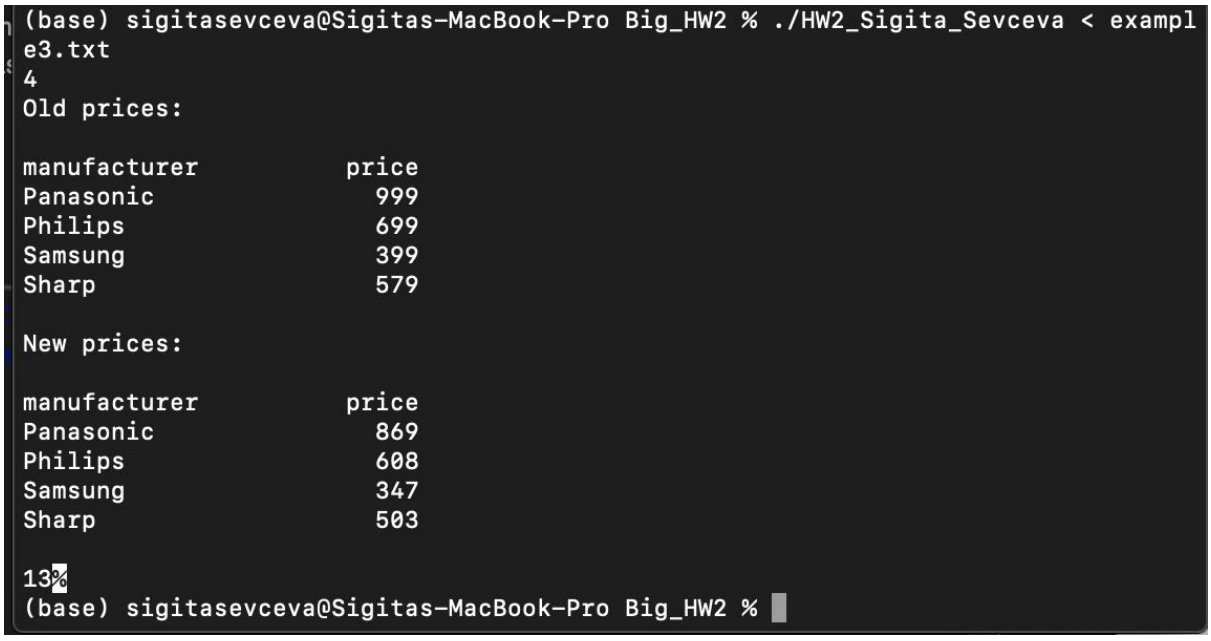

Figure 2: Program functioning properly with different input