Winter 2011 **CPE 101:** Fundamentals of Computer Science I Alexander Dekhtyar

Lab Test Example # 3

## Instructions

- Time to complete the exam: 50 minutes.
- This exam is individual
- This exam is open book
- This exam is closed everything else, including lecture notes, electronic devices, etc
- The only programs you are allowed to have open on the computer during this exam are:
  - The text editor of your choice
  - Terminal window(s)
  - ssh connection to vogon
- Do not use any code you may have access to from earlier in this course or from other courses
- Before starting your work, open a terminal session, and create a directory for the exam. It will be referred to as `labexam01` below.
- The problem description includes all assumptions necessary to answer the problem. Please raise your hand, or approach the instructor if you have any questions.
- Please, collect candy wrappers after you! Additional candy is available, at the instructor's desk: help yourself!
- GOOD LUCK!

## Assignment

In the past few years real estate prices soared, then stagnated, then went down. Write a program that takes as input information about a list of houses currently on the market and outputs, for each house a prediction of the price it will be sold at. Call the program `predict.c`.

**Input.** The input to your program is information about various houses on the market. For each house, the following information is specified:

| | |
|---|---|
| `bedrooms` | number of bedrooms |
| `size` | size of the house in square feet |
| `DOM` | days on the market |
| `Price` | current price in thousands of dollars |

The input will always end with a line containing four zeros. For example, the following input:

```
4 2500 30 450
3 1750 88 380
4 1950 70 399
0 0 0 0
```

describes three houses. House 1 is a 4-bedroom 2500 sq. ft. house that has been on the market for 30 days and currently sells for $450,000. House 2 is a 3-bedroom, 1750 sq. ft. house sitting on the market for 88 days and currently selling for $380,000. The third house is a 4-bedroom, 1950 sq.ft. house, that's been on the market for 70 days and currently sells for $399,000.

Your program shall read input until it encounters the terminal line (`0 0 0 0`). For each input, your program shall output two numbers: the current price of the house and the projection of the price at which the house will sell.

Your program shall contain a function `int projectPrice()` which takes as input *all* information about a house (number of bedrooms, size, days on the market, price) and outputs back an integer number representing the estimated price for which the house will sell (in thousands).

**int projectPrice().** Your program shall estimate the final price of the house as follows.

Presently, the average number of days on the market for a house is 75. `projectPrice()` predicts the number of days a house will be on the market and based on it computes the predicted final price.

To predict the number of days on the market, first, compute the price-per-sq. foot of the house and square feet per bedroom of the house:

$$\mathsf{pricePerSqFt} = \frac{\mathsf{Price} * 1000}{\mathsf{size}}$$

$$\mathsf{sqFtPerBd} = \frac{\mathsf{size}}{\mathsf{Bedrooms}}$$

Houses with better price per square foot would sell faster. This is especially true for roomier houses. We use sqFtPerBd to measure *roominess.* Assume the average price (AvSqFtPrice) per square foot to be $190 for 3-bedroom houses and $200 for all other houses.

Estimate the remaining days on the market for the house as

$$\mathsf{DaysRemaining} = 75 \cdot \frac{\mathsf{pricePerSqFt}}{\mathsf{AvSqFtPrice}} \cdot \frac{500}{\mathsf{sqFtPerBd}} - \mathsf{DOM}$$

Estimate the *final sale price of the house* as follows:

- If DaysRemaining is 0 or less, estimate the final sale price as 95% of the current price.
- In all other cases, estimate that the final sales price declines 0.2% off the curent price for each remaining day the house spends on the market.

**Please, note: all computations shall be performed using floating point operations.** (use the "multiply by 1.0" trick if you need it). The final house price shall then be converted into an integer by dividing the house price in dollars by 1000 and taking the ceiling (i.e., all prices get rounded up).

**Output.** The output of your program must coincide with the output of the instructor's program. The format of the output is straightforward: for each house on the input list, your program shall output the current price (in thousands of dollars) followed by `" --> "` (notice the spaces), followed by the program's prediction.

Sample output

```
$ cat test-house03
4 2500 30 450
3 1750 88 380
4 1950 70 399
0 0 0 0

$ predict < test-house03
450 --> 429
380 --> 361
399 --> 393
```

**Error checking etc.** All test cases will contain only valid inputs. You do not need to guard your inputs (e.g. to check that entered values are non-negative). Notice that 0 is a valid input for days on the market.

**Testing your solution**

Compile your program using the following command:

```
> gcc -ansi -Wall -Werror -lm -o predict  predict.c
```

You have access to my full set of test cases for this program, my executable of it and scripts to test my executable and your executable. Copy all the files to your `labexam02` directory with the following command:

```
cp ~dekhtyar/www/101-Winter2011/rfgk09/* .
```

Instructor's executable is called `predict-alex`. Do `ls -al` to ensure that it is executable. If not, run `chmod u+x predict-alex`. Perform the same check for predict-test.csh and predict-alex-test.csh files that are also copied. Test file names are test-house01 through test-house10.

To test your program first compile it into an executable named predict. Then run the following command:

```
> predict-test.csh
```

The script will run your program on all test cases and produce output. To check it against the instructor's output, open a second terminal session, change to labexam01 directory and run the command:

```
> predict-alex-test.csh
```

Compare the two outputs.

**Note.** You can run the following commands:

```
> predict-test.csh > my.out
> predict-alex-test.csh > alex.out
> diff my.out alex.out
```

If the last command does not produce any output, your results exactly match the instructor's results.

**Submitting your solution**

Once you are satisfied your program meets all the requirements you may turn it in and leave the exam. Run the following command on vogon to submit your program:

```
handin dekhtyar  labexam03 predict.c
```