

Assignment 3

Due: Sunday, 13 May 2018, 11:59 PM on TEACH as a .tar

Inheritance through Real Estate Tycoon

Summary Problem Statement

In this program you will create a real estate tycoon game which is turn based (a turn simulating one month). The game will be won if the player reaches a bank account value of \$1 million. The player will lose the game if they go bankrupt (have \$0). The player starts with a bank account value of \$500,000 and no properties.

Each turn:

- Payments will be paid to outstanding debt and taxes.
- Tenants may leave properties if the rent exceeds their ability to pay.
- Rent will be collected from rent producing properties which have tenants.
- A random event will occur. Random events include:
 - Weather: hurricane, tornado, earthquake, wildfire.
 - Stock market crash.
 - Gentrification: big business moves to town, natural resource discovered, positive press about the city.
- A listing of properties with potential to purchase a property.
- The ability to sell a property.
- The ability to raise or lower rents on a property.

Each property has the following attributes:

- Property value
- Location
- Mortgage
- Property tax: fixed at 1.5%
- Tenant(s)

The types of properties are:

- House: can be rented by one tenant
- Apartment Complex: can collect rent on each apartment in the complex at the same rate
- Business Complex: has spaces of various sizes and rent values, can have multiple tenants

Implementation Details

Set Up

The game will start with three of each property type. Each one will be randomly assigned a location (United States: SE, NE, Midwest, SW, and NW), randomly assigned a property value (House \geq 100,000; Apartment Complex \geq 300,000; Business Complex \geq 400,000; all starting

property values are capped at 600,000), a mortgage which includes a randomly assigned monthly (turn) price (not to exceed 5000) and a calculated duration of mortgage based on the monthly price (exclude interest). The property tax for all properties will be fixed at 1.5% of the property value. Property taxes will be assessed every twelve months (turns). Apartment complexes will have a random number of rooms assigned to them (max 10). Business complexes will have a random number of spaces (max 5) assigned with a random size attached to each space (small, medium, or large).

Properties will have tenants in each building. The tenants may be citizens or businesses. Both will have a max, randomly assigned monthly budget which is fixed through out the game (500 to 5000 for citizens, 2000 to 10000 for businesses). Both will also have a randomly assigned agreeability score ranging from 1 to 5 (1 being not agreeable and 5 being incredibly agreeable) which will be fixed throughout the game. Citizens may occupy houses and apartments, businesses may only occupy business complexes. Each business that pays rent in a complex increases the property value by 1% each turn.

The player will begin with a bank account of \$500,000 and no properties.

Playing the Game

The goal of the game is to make money, but to make money, some money must be spent. The turn will begin by first collecting the monthly rent from the appropriate properties, adding this amount to the bank. If rent was set higher on the last turn than what a tenant's monthly budget is, then this turn they will leave the property and not pay rent. Tenants cannot be replaced in this game once they leave. The room they vacate will be left empty. If a tenant's agreeability score is less than 2 and the rent is higher than their monthly budget, they will refuse to pay rent but will stay in the property. In this game, the player can not evict tenants, the only way to collect rent on disagreeable tenants is to lower the rent.

The monthly mortgage payments will then be made for the appropriate properties, subtracting this amount from the bank. Every twelve turns, property tax will be paid, subtracting this amount from the bank. The property tax is 1.5% of the current value of the property.

A random event will then occur. The following effects will occur for all properties that meet the requirements:

- A hurricane occurs, impacting properties in the SE by decreasing their property value by 50%.
- A tornado occurs, impacting properties in the Midwest by decreasing their property value by 30%.
- The earthquake occurs, impacting properties in the Northwest by decreasing their property value to 10%.
- A wildfire occurs, impacting properties in the Southwest by decreasing their property value by 25%.
- Stock market crash: all properties decrease in value by 30%.

- Gentrification (big business moves to town, natural resource discovered, positive press about the city): a random location is selected, all properties in that location see and increase in property value of 20%.

After the random event, the player will be given the option to buy a property, sell a property, or adjust rents on a property.

If the player chooses to buy a property, three random properties should be pulled from the arrays of properties set up at the beginning of the game. If all of those properties have been bought, new ones should be generated into the array. The property will be added to the player's list of properties of that type and removed from the general arrays. The mortgage on the property will begin being collected the next turn and will continue until the property is paid off.

If the player chooses to sell a property, the player will select which property they wish to sell and at what price. Properties can only be sold if there are no tenants on the property. A random number from 1 to 3 will be selected to determine if the player will get their asking price (3), get the property value (2) or get 10% less than the property value (1). The amount received will be added to the bank account.

If the player chooses to adjust rents, they must select which property and specify what the new rent will be. For business complexes, the player can select a specific space in the property to adjust the rent on.

Ending the Game

Play will continue until the bank account is 0 or 1,000,000.

Requirements

- The program must run to the above specified requirements in their entirety.
- A readable and understandable user interface must be present.
- Classes must be used.
- Inheritance must be used, demonstrating the use of base and child classes.
- Appropriate use of the Big 3.
- Files must be separated.
- A functioning Makefile must be provided.
- No memory leaks.
- No functions greater than 20 lines.
- No use of libraries that have not been permitted in the course (namely <algorithms> and <vector>)
- Lack of correct coding style will incur an automatic 10 point deduction.