

Name: Thomas Zalonis
Date: 02/09/2007
Class: CSCI 360
Description: Individual homework assignment number 4,
chapter 9 – exercises 2, 7 and 14

Exercise #2:

Data Structure: a linked list

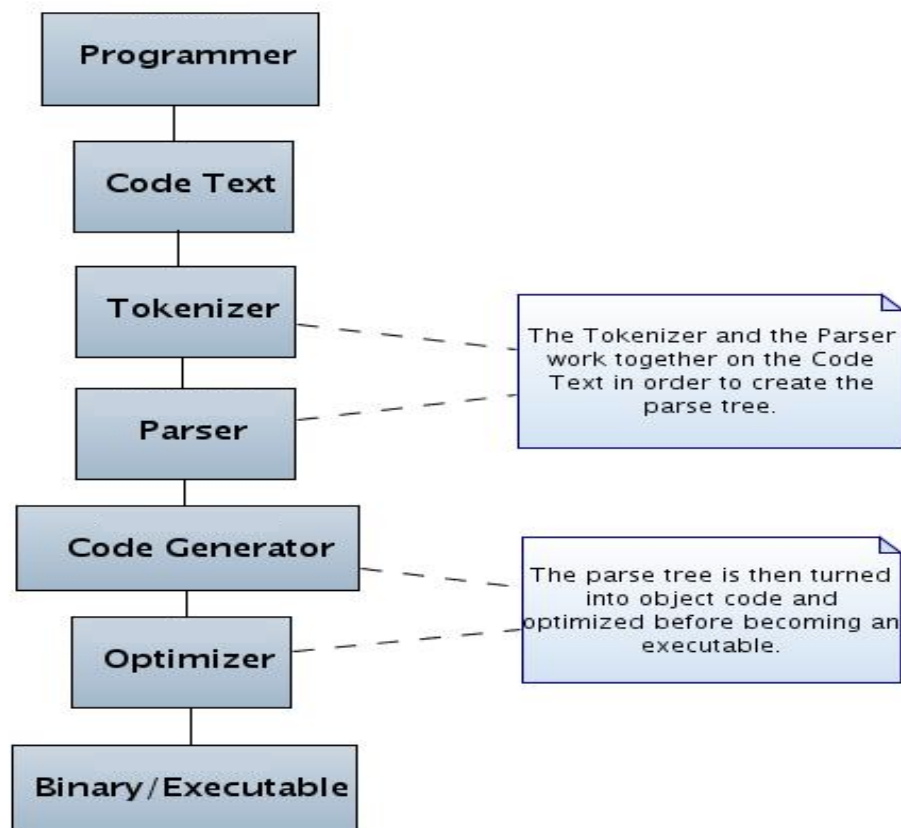
Alternative: an array

Rationale:

I chose the linked list to store the playlists in because the linked list allows for a dynamic capacity. Since the station requires the ability to modify the list after it has been generated, an array could cause problems if the station wishes to add more songs to the list than the array has room for. Efficiency of such modifications is also an issue. An array would allow for more efficient removal/addition of songs since the linked list would have to first traverse its way through the list to get to the correct location. However, I think the ability to allow for a variable length playlist is more important than efficiency for this problem because a slightly slower program will still work. A capacity limit, on the other hand, will limit what the user is able to do.

Exercise #7:

Below is a box-line-diagram representation for the compiler.



Exercise #14

Corrections Made:

1. The CreditCardValidator component was stereotyped as `<<component>>`
2. Added the stereotype `<<delegate>>` to the delegation connector arrow connecting the CardValidation interface to the Validator.
3. The interface that is realized by BankProxy was given the name BankInterface.
4. Added a delegation arrow and stereotype for the delegation connector between BankProxy and InternetPaymentGateway.

(Component diagram for Exercise14 using the corrections above)

