

Project 1: Design

Austin Chase Minor

March 28, 2016

1 Algorithms

```
function ADD_QUESTION(string quesiton, string answer, user_prop modifica-
tions, quiz_list list)
    quiz_node *elem = quiz_node(null, question, answer, modifications)
    if list.size > 0 then
        list.tail.next = elem
        list.tail = list.tail.next
    else
        list.head = elem
        list.tail = elem
    end if
end function

function REMOVE_QUESTION(quiz_list list)
    if list.size > 0 then
        quiz_node* elem = list.head
        if list.size == 1 then
            list.head = null
            list.tail = null
        else
            list.head = list.head.next
            list.size = list.size - 1
        end if
        return elem
    else
        return null
    end if
end function

function PARSE_QUESTION(file_stream file)
    string question, answer, user_mods
    question = first line
    answer = second line
    user_mods = third line
    return quiz_node(null, question, answer, parse_user_prop(user_mods))
```

```

end function
function PARSE_USER_PROP(string props)
    string time, money, intelligence
    time = first value before comma in props
    answer = second value before comma in props
    user_mods = third value before comma in props
    return user_prop(time, money, intelligence)
end function
function GRUNT_WORK(user& user_in)
    time_loss = ((random() % 2) + 1) * -1
    intelligence_loss = ((random() % 2) + 1) * -1
    mod_user_prop(user_in.attr, user_prop(time_loss, 0, intelligence_loss))
end function
function DO_NOTHING(user& user_in)
    mod_user_prop(user_in.attr, user_prop(1, 0, 0))
end function
function PROFESSOR(user& user_in)
    time_loss = ((random() % 2) + 1) * -1
    intelligence_gain = (random() % 3) * -1
    mod_user_prop(user_in.attr, user_prop(time_loss, 0, intelligence_loss))
end function
function GRAD_STUDENT(user& user_in)
    time_loss = ((random() % 2) + 1) * -1
    mod_user_prop(user_in.attr, user_prop(time_loss, 0, 0))
end function
function GRADE_PAPERS(user& user_in)
    time_loss = ((random() % 2) + 1) * -1
    money_gain = ((random() % 2) + 1)
    mod_user_prop(user_in.attr, user_prop(time_loss, 0, money_gain))
end function
function PUZZLE(user& user_in)
    quiz_node* question = remove_question(quiz_list)
    string user_response
    output question.question
    output "User Response: "
    input user_response
    if user_response == question.answer then
        output "You are correct"
        mod_user_prop(user_in.attr, question.mods)
    else
        output "You are wrong. Correct answer: "
        output question.answer
        user_prop neg_mods = user_prop(question.mods.time, question.mods.money
        * -1, question.mods.intelligence * -1)

```

```

        mod_user_prop(user_in.attr, neg_mods)
    end if
end function

```

2 System-Tests

Methods	start menu	move select	display high scores	write high scores	display character
Test	Tester should verify that the proper menu is displayed.	Tester should verify that the proper menu is displayed.	Tester should verify that the high scores are displayed.	Tester should verify that the proper high scores are displayed. (With default scores file) Chase: Score:3 ... Minor: Score: 5 ... Austin: Score: 7	Tester should verify that the user is properly displayed.
Test	Tester should verify that each command does appropriate action.	Tester should verify that each command does appropriate action.	Nothing	Nothing	Tester should verify that after some game-play user statistics have changed.
Test	Tester should verify that a non-menu command brings up message and repeats menu.	Tester should verify that a non-menu command brings up message and repeats menu.	Nothing	Nothing	Nothing