Tutorial 4: Arcade Menu

Thomas Chapman

# Specification

Develop a simple command line menu for an arcade game.

# Technical design

This is the pseudocode for the arcade game menu, it can handle user inputs, balance, and game fees:

controlGameApplication

**local option, balance**

call payInitialFee**(balance)**

call enterOption**(option)**

while (not(option = 'Q') and (balance>0))

call processOption**(option, balance)**

call enterOption**(option)**

endwhile

output("Thanks for playing")

call showBalance(balance)

proc payInitialFee**(OUT: balance)**

set balance to 100

endproc

proc enterOption**(OUT: option)**

output("Enter option (P:play or B:balance or Q: quit> ")

input(option)

call putInUppercase(**option)**

endproc

proc putInUppercase(**IN/OUT: option)**

set option to uppercase(option)

endproc

proc processOption**(IN: option, IN/OUT: balance)**

if(option = 'P') then

call playGame**(balance)**

else

if(option = 'B') then

call showBalance**(balance)**

else

output(“ERROR: Invalid Command!”)

endif

endif

endproc

proc playGame**(IN/OUT: balance)**

output(“Playing...”)

set balance to balance – 20

//...

endproc

proc showBalance**(IN: balance)**

output(“The current balance is £”, balance)

endproc

Here is a flowchart of the main loop:

Start

call payInitialFee

call enterOption(option)

option is not Q and balance > 0

True False

call processOption(option, balance)

output("Thanks for playing")

call enterOption(option)

call showBalance(balance)

End

# Test plan

|  |  |  |
| --- | --- | --- |
| Input | Expected Output | Actual Output |
| P | Playing… \*another prompt\* | Playing… \*another prompt\* |
| B | Your current balance is £\*balance\* | Your current balance is £\*balance\* |
| Q | Exit program | Exit program |
| Any other input | ERROR: Invalid Command! \*another prompt\* | ERROR: Invalid Command! \*another prompt\* |

# Schedule

Estimated and actual amount of hours spent on each part of the program:

* Spec: Estimated 0.25, Actual 0.25
* Design: Estimated 0.25, Actual 0.5
* Implement: Estimated 0.5, Actual 0.25
* Debug and test: Estimated 0.25, Actual 0.25
* Slack: Estimated 0.25, Actual 0
* Total: Estimated 1.5, Actual 1.25

This was done within my estimated time limit, overall I’m very happy with the outcome.