John Hibbert / CSC402 / 11/13/2015

Final Proposal

For my CSC402 final project, I propose that I write a small domain specific language that will allow a user to easily create animations in Windows batch scripts. It will compile the input into a batch file with the correct syntax.

The original idea was to do this for DOS instead, and specifically DOSBox, which is an open-source DOS emulator. The problem is that the normal animation techniques don't want to run in it and I don't know why. I suppose that if I do manage to work it out at some point, even if the techniques are radically different, I can expand the language to include the output.

For this, in order to create animations, we are just going to redraw the images repeatedly with small modifications in a brute-force approach, similar to how a cartoon was drawn in the early days. The user will give a set of strings and some parameters that describe how the image will be animated, and the compiler will take that image and encode it into the batch file.

An example piece of code could be:

```
// Draw the image TEST with a downward wipe

mode = downwipe     // what kind of animation
x = 1               // where we are drawing the given image
y = 1               // on the x and y plane
                    // if we implement sliding animations
                    // we might need two sets of X and Y
method = ping       // how we are delaying to simulate animation

startimage          // a set of strings that are the picture
"  ________                          "
"|__    __|                         "
"   | |      ______   __ __   ______ "
"   | |    | _____| |  ___| |_    _|"
"   | |    | _____| |___  |   | |  "
"   |_|    |______| |_____|   |_|  "
endimage
```

The return on this would be a fleshed out batch file that draws the top line first, waits a moment then clears the screen, draws the top two lines, waits a moment then clears the screen, draws the top three lines, and so on until the whole object was drawn and the batch file had completed.

The output file would look something like:

```
echo  ________
echo.
echo.
echo.
echo.
echo.
ping 1.1.1.1 -n 1 -w 500 >NUL
cls
echo  ________
```

John Hibbert / CSC402 / 11/13/2015

```
echo ^|__    __^|
echo.
echo.
echo.
echo.
ping 1.1.1.1 -n 1 -w 500 >NUL
cls
echo  ________
echo ^|__    __^|
echo    ^| ^|    ______   __ __   ______
echo.
echo.
echo.
ping 1.1.1.1 -n 1 -w 500 >NUL
cls
echo  ________
echo ^|__    __^|
echo    ^| ^|    ______   __ __   ______
echo    ^| ^|   ^| ____^| ^|  ___^| ^|_   _^|
echo.
echo.
ping 1.1.1.1 -n 1 -w 500 >NUL
cls
echo  ________
echo ^|__    __^|
echo    ^| ^|    ______   __ __   ______
echo    ^| ^|   ^| ____^| ^|  ___^| ^|_   _^|
echo    ^| ^|   ^| ____^| ^|___  ^|   ^| ^|
echo.
ping 1.1.1.1 -n 1 -w 500 >NUL
cls
echo  ________
echo ^|__    __^|
echo    ^| ^|    ______   __ __   ______
echo    ^| ^|   ^| ____^| ^|  ___^| ^|_   _^|
echo    ^| ^|   ^| ____^| ^|___  ^|   ^| ^|
echo    ^|_^|   ^|______^| ^|_____^|   ^|_^|
ping 1.1.1.1 -n 1 -w 500 >NUL
```

What these commands mean:
ECHO: Print the following line to the terminal. (To print "|" it must be escaped, which is why the ^ mark is there before each of them.)
PING: This is one of a few options we have as far as how we make the computer wait.
Here, we ping a non-existant IP address once and wait 1/2 a second.
"-n 1" means it tries once instead of the default 4 time
"-w 500" means we wait for 500 milliseconds instead of the default.
">NUL" means to redirect the standard output to the NUL device, which just discards it.
This technique doesn't work if your computer is not connected to the internet, so I will probably include other options such as a loop that just goes in circles doing nothing.
CLS: Clear the screen.

John Hibbert / CSC402 / 11/13/2015

(Much of my knowledge about this comes from these links)
http://www.robvanderwoude.com/battech_redirection.php
http://www.robvanderwoude.com/wait.php
http://ss64.com/nt/echo.html
http://hubpages.com/technology/How-to-Make-A-Windows-Batch-File-Loop-Sleep-or-Delay-For-Specific-Interval-of-Time

The main challenge of this will be to take the user's input and convert it into a datatype that we can manipulate with java code. My current idea is a queue of arrays of strings. Each array contains 24 elements, each of which is a string of 79 characters (not including those escape characters), which is what is to be drawn on that line for that frame. The benefit of doing it this way is that once I have worked out a way to go from language to data type to output, adding additional animation modes would be just a matter of adding additional methods that describe how to manipulate the data.

Milestones:

Consider the scope of the problem and compose a grammar that would allow the user to express the basics, even if the starting animations are somewhat simple.

Compose semantic actions that will convert the given information into Java code.

Create classes that will accept the information and output an approrpiate string which can be made the output for the batch file.

Add additional forms of animations if time allows.

Add support for DOS Scripting, or maybe even Linux Terminal scripting, if time allows.

Resources required: Nothing much.

To create it: My computer, Antlr, Gedit, VirtualBox, Ubuntu.

To test it: Microsoft Windows and Notepad (or in this case Notepad++).

As far as I can tell, this won't work on a Mac. I don't know what that means for the assignment.