Lab 07: doit

doit is an sh-compatible command language interpreter that executes commands read from the standard input.

Your program should read a command from the standard input and then fork a child to execute the command the user asked for. The only build-in command your program should support is **q**, use to **quit** your program.

Sample Execution:

```
root@luna:CST8234# ./doit
doit# ls
MakeArg.c  01_exec.c  01_fork.c  02_fork.c  03_fork.c  04_fork.c
doit# pwd
/home/Material/CST8234
doit# date
Sat Nov 21 14:10:31 EST 2015
doit# q
```

You can use the function `MakeArg( )`, posted with this lab, to convert a string into a vector of strings and then use the `execvp()` function.

Your program should loop forever, present a prompt to the user and read a string, if the string is the letter q, your program should terminate. If not, it should fork a new process and wait until it finishes executing. The child, should convert the string into an array of strings, call the execvp( ).   When the child finishes, the parent presents the prompt again to the user.