```
    pipesn: pipesn.c pipesn.h
            gcc pipesn.c -DF=0 -o pipesn
    // pipesn.c
    //
    // Author: Michael Kepple
    // Date: 9 Feb 2013
    // Description: pipes to a variable number of process with dup, pipe.
    //
    #include "pipesn.h"

    int main (int argc, char* argv[])
    {
        char buffer[200];
        memset(&buffer[0], 0, sizeof(buffer));
        int numProc, status, i, j, count = 0;
        if (F == 0)
        {
            if ((argc == 3) && (atoi(argv[1])))
            {
                numProc = atoi(argv[1]);
                int pipefds[2 * numProc];
                for (i = 0; i < numProc; i++)
                {
                    snprintf(buffer, sizeof(buffer), "gcc pipesn.c -DF=%d -o p%d", i+1,
     i+1);
                    system(buffer);
                    memset(&buffer[0], 0, sizeof(buffer));
                    pipe(pipefds + i * 2);
                }
                for (i = 0; i <= numProc*2;)
                {
                    if (fork() == 0)
                    {
                        memset(&buffer[0], 0, sizeof(buffer));
                        if (i != (numProc*2))
                            dup2(pipefds[i+1], 1);
                        if (i != 0)
                            dup2(pipefds[i-2], 0);
                        if (i == 0)
                            printf("%s", argv[2]);
                        fflush(stdout);
                        for (j = 0; j <= 2*numProc; j++)
                            close(pipefds[j]);
                        snprintf(buffer, sizeof(buffer), "p%d", count);
                        execl(buffer, "-flag=5", NULL);
                    }
                    count++;
                    i += 2;
                }
                for (j = 0; j <= 2*numProc; j++)
                    close(pipefds[j]);
                for (j = 0; j < numProc; j++)
                    wait(&status);
                printf("\n");
            }
        }
        else
        {
            read(0, buffer, sizeof(buffer));
            printf("%s%d", buffer, F);
        }
    }
    #ifndef PIPESN_H
```

```
    #define PIPESN_H
    #include <stdio.h>
    #include <stdlib.h>
    #include <unistd.h>
    #include <sys/types.h>
    #include <string.h>
    #endif
```