Problema 1

```c
#include <stdio.h>
#include <errno.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>

int main ()
{
   pid_t pid = fork ();

   if (pid < 0)
      return errno;
   else if (pid == 0){
      char *argv[] = {"ls", NULL};
      execve ("/bin/ls", argv , NULL);
      perror(NULL);
      }
   else {
      printf("Child PID: %d\n", pid);
      wait(NULL);
      printf("Parent PID: %d\n", getpid());
   }

   return 0;
}
```

Problema 2

```c
#include <stdio.h>
#include <errno.h>
#include <sys/types.h>
#include <unistd.h>
#include <sys/wait.h>
#include <stdlib.h>

int main (int argc, char **argv)
{
   int n = atoi(argv[1]);

   pid_t pid = fork ();

   if (pid < 0)
      return errno;
   else if (pid == 0){
      while (n != 1) {
         printf("%d ", n);
         if (n % 2 == 0)
            n = n/2;
         else n = 3*n + 1;
      }
      printf("1\n");
   }
   else {
      int w = wait(NULL);
      printf("Child PID: %d finished\n", w);
      printf("Parent PID: %d\n", getpid());
   }

   return 0;
}
```

Problema 3

```
#include <stdio.h>
#include <errno.h>
#include <sys/types.h>
#include <unistd.h>
#include <sys/wait.h>
#include <stdlib.h>

int main (int argc, char **argv)
{
   pid_t pidd = fork();

   if(pidd == 0) {
      for (int i=0; i<argc-1; i++) {
         int n = atoi(argv[i+1]);
         pid_t pid = fork ();
         if (pid < 0)
            return errno;
         else if (pid == 0){
            printf("%d: ", n);
            while (n != 1) {
               printf("%d ", n);
               if (n % 2 == 0)
                  n = n/2;
               else n = 3*n + 1;
            }
            printf("1\n");
            exit(0);
         }
      }

      for (int i=0; i<argc-1; i++) {
         int w = wait(NULL);
         printf ("Done Parent %d Me %d\n", getppid(), w);
      }

   }
   else {
      printf ("Starting Parent %d\n", getpid());
      int ww = wait(NULL);
      printf ("Done Parent %d Me %d\n", getppid(), getpid());
   }
   return 0;
}
```