Apuntes de PSP

* Los includes que hemos visto son:
  + #include <stdio.h>
  + #include <strings.h>
  + #include <unistd.h>
  + #include <stdlib.h>
  + #include <signal.h>
  + #include <sys/types.h>
  + #include <sys/wait.h>
  + #include <math.h>
* Para comparar chars lo haces de esta manera: if(variableQueContieneChar==’+’){

Printf(“Soy un mas”);

}

* Para declarar finales es:
  + #define nombreVariable 0-9/strings/double/….
    - Ejemplo: #define CERO 0
    - Ejemplo #define APUNTES “APUNTES”
* Para crear un random es:
  + Variable=rand%el numero que quieras;
  + Para hacer que sea random, se usa srand(time(NULL))
* Para crear un pipe es:
  + Creas un array llamado acciones[2];
  + Pipe(acciones)
    - Si quiero cerrar un extremo es: close(acciones[0]) si quieres cerrar lectura y close(acciones[1]) si quieres cerrar la escritura
    - Si quiero hacer lectura o escritura: read(acciones[0](lectura),variable a pasar, sizeof(variable a pasar)). Write(acciones[1],variable para enviar, sizeof(Variable para enviar));
    - TEN EN CUENTA QUE TIENEN QUE TENER EL MISMO TAMAÑO SI ES UN ARRAY Y EL MISMO TIPO
    - IMPORTANTE, PARA PASAR CADENAS DE TEXTO Y/O CARACTERES, CUANDO DEFINAS UN PIPE(SIEMPRE TIENE QUE IR CON & POR EJEMPLO pipe(&variableQueContieneUnChar);
    - CUANDO LO ESCRIBAS CON WRITE VA DE ESTA MANERA: write(envioSimbolo[1],&variableQueContieneUnChar,sizeof(variableQueContieneUnchar)) Mas detalles en: REPASOEXAMEN2/ejercicio6.c
* Escritura en archivos se hace de esta manera:
  + Primero definimos el archivo : FILE\* nombreArchivo;
  + Luego lo abrimos: archivo=fopen(“nombreDelArchivo.txt”,\*);
  + \*hay varios modos para abrirlo:
    - R: lectura
    - W: escritura
    - A: append y no sobrescribe
    - R+/W+: lectura y escritura
    - A+: escritura y no sobrescribe
  + Si quiero escribir en el documento se puede hacer de dos maneras:
    - Fprintf(archivo,”lo que haya que escribir”)
    - Fwrite(datos, sizeof(datos),archivo)
    - IMPORTANTE NUNCA HACER: ~~archivo~~=fprintf/fwrite (archivo,””);
  + IMPORTANTE, CUANDO SE HAYA TERMINADO HACER fclose(archivo)
* Para crear y manejar las señales como dioses del olimpo necesitas:
  + Para cualquier señal que necesites para manejar determianda señal haces:
    - Signal(Tipo de señal que quieres manejar,función que va a manejar esa señal).
    - Si estas usando SIGNKILL necesitas hacer:
      * Kill(lo que quieras matar, SIGKILL)
    - IMPORTANTE!!!! Kill es un comando para mandar señales, es decir, si quieres “Automatizar señales” tienes que lanzar un kill a la variable del hijo o a lo que tenga pid\_t y luego el tipo de señal que quieras mandar, si esto lo combinas con Signal, puedes lanzar señales y encima personalizar esas Señales, funciona como los try/catch en java
* Fflush(stdout): es una función que lo que hace es forzar todos los printf posteriores a este para que se escriban en la pantalla de ejecución del proceso
* En los printf tenemos:
  + %d para los integer
  + %s para los strings/char
  + %c para chars