

Material permitido:
Calculadora NO programable.
Tiempo: **2 horas.**
R

Aviso 1: Todas las respuestas deben estar razonadas.
Aviso 2: Escriba sus respuestas con una letra **lo más clara posible.**
Aviso 3: No use *Tipp-ex* o similares (atasca el escáner).

ESTE EXAMEN CONSTA DE 5 PREGUNTAS

1. (2 p) Explique **razonadamente** si las siguientes afirmaciones son verdaderas o falsas:
 - I) (1 p) En UNIX BSD4.3 la prioridad de planificación de un proceso es un valor comprendido entre 0 y 127. Las prioridades 0 a 63 están reservadas para el núcleo y las restantes para los procesos en modo usuario.
 - II) (1 p) El núcleo del UNIX SVR3 utiliza el algoritmo `attachreg()` para ligar una región al espacio de direcciones virtuales de un proceso.
2. (2 p) Enumere y explique **brevemente** los atributos de un fichero comúnmente soportados por los sistemas de ficheros de UNIX.
3. (2 p) Supóngase que el directorio de trabajo actual de un usuario contiene dos ficheros ordinarios (`carta.txt`, `programa.sh`) y tres subdirectorios (`Escritorio`, `Descargas`, `Basura`).
 - a) (0.5p) ¿Qué orden se debe teclear desde la línea de comandos (\$) para conocer la ruta absoluta del directorio de trabajo actual?
 - b) (0.5p) ¿Qué orden se debe teclear para mover al subdirectorio `Basura` los ficheros `carta.txt` y `programa.sh`?
 - c) (0.5p) ¿Qué orden se debe teclear para conocer los ficheros que se encuentran en la carpeta `Descargas`?
 - d) (0.5p) ¿Qué orden se debe teclear para que el directorio de trabajo actual pase a ser el subdirectorio `fotos`?
4. (1.5 p) Explique **brevemente** los principales servicios realizados por el núcleo de UNIX.
5. (2.5 p) Al compilar el código C del programa que se muestra en la página siguiente se crea el ejecutable `prog`. Supóngase que 1) al invocar `prog` desde la línea de órdenes del terminal (\$) se le asocia el `pid=1120`. 2) que la asignación de los `pid` se realiza secuencialmente a partir del `pid` del padre, es decir, `pid=1121` es el primer hijo creado, `pid=1122` es el segundo hijo creado, etc. 3) El intérprete de comandos desde donde se lanza `prog` tiene asociado el `pid 1000`. Conteste razonadamente a los siguientes apartados:
 - a) (1 p) Explique el significado de las cinco sentencias enumeradas ([1]) del código.
 - b) (1.5 p) Explique **detalladamente** el funcionamiento de este programa si se invoca desde el intérprete de comandos mediante la orden `$./prog 4`

La pregunta 5 continúa en la siguiente página

```
#include <stdio.h>
#include <stdlib.h>

[1] main(int argc, char *argv[])
    {

        int a,b=0,c,d;

        if (argc!=2)
        {
            exit(3);
        }
        else
        {
[2]             a=atoi(argv[1]);
[3]             while (fork()==0 && b!=a)
                {
[4]                 printf("\nMensaje 1[%d]\n", getpid());
                    b=b+1;
                }
            }
[5]             c=wait(&d);
[6]             printf("\nMensaje 2[%d]=%d\n", getpid(),getppid());
        }
    }
```