|  |  |
| --- | --- |
| *!!!* | ***Esta prueba solo pueden realizarla los estudiantes que hayan aprobado la evaluación continua*** |

**Ficha técnica de la prueba de síntesis**

* No es necesario que escribas tu nombre. Una vez resuelta la prueba final, solo se aceptan documentos en formato .doc, .docx (Word) y .pdf.
* Comprueba que el código y el nombre de la asignatura corresponden a la asignatura de la que te has matriculado.
* Tiempo total **30 minutos.** Se dispone de un tiempo añadido suficiente para la descarga y entrega de la prueba.
* El valor numérico de cada pregunta se indica en cada una de ellas.
* ¿Puede consultarse algún material durante la prueba de síntesis?  ¿Qué materiales están permitidos?
* ¿Puede utilizarse calculadora?  ¿De qué tipo?
* Indicaciones específicas para la realización de esta prueba de síntesis:

Enunciados

**Pregunta 1.** Indica que mostrará el siguiente código y explica porqué.

intmain() {

char cadena[50] = "La\ncadena\nes\nmia\n";

printf("CADENA: %s \n", cadena);

for (int i = 0; i < strlen(cadena); i++) {

if (cadena[i] == '\n')cadena[i] = ' ';

}

printf("CADENA: %s \n", cadena);

return 0;

}

Con el primer printf mostrara en pantalla el valor de cadena, y lo hara realizando saltos de línea debido a los \n.

Por lo que aparecerá:

La

cadena

es

mia

y dos espacios mas. Luego en el bucle for se recorre cada caracter de cadena y si es \n lo substituira por un espacio por lo que en el segundo printf aparecerá la frase completa:

La cadena es mia

Finalizando con un salto de línea del printf.

**(2,5 puntos) (Resolver la pregunta aproximadamente en 3 líneas)**

**Pregunta 2.** Si en un código cometemos un error como por ejemplo el escribir una instrucción sin ser finalizada con un punto y coma, ¿podremos usar las herramientas de depuración para encontrar ese error? Justifica la respuesta.

**(2,5 puntos) (Resolver la pregunta aproximadamente en 5 líneas)**

Si, las herramientas de depuración también detectan errores de sintaxis, y al intentar compilar o debugar el programa nos saltaría un mensaje avisando de la falta del punto y coma.

**Pregunta 3.** Muestra un código programado en C que pida al usuario que introduzca una ip y ejecute el comando necesario para comprobar la conectividad a dicha ip mostrando el mensaje “hay conectividad” o “no hay conectividad” en función de la respuesta obtenida

**(2,5 puntos) (Resolver la pregunta aproximadamente en 8 líneas)**

char ip[20]=””;

char comando[100]=”ping “; char velo[10]=””;

printf(“ Introduce una IP:”);

fflush(stdin);

gets(ip);

strcat(comando, ip);

strcat(comando, “>result.txt”);

system(comando);

FILE\* archivo

archivo=fopen(“result.txt”,”r”);

fseek(archivo, -8, SEEK\_END);

fgets(velo. 8, archivo);

if(strstr(velo,”ms”) printf(“hay conectividad”);

else printf(“no hay conectividad”);

P**regunta 4.** Aplicando lo tratado en los productos 2 y 3, muestra la línea de código que permitiría a una aplicación programada en C listar los archivos y directorios contenidos en el directorio “d:/archivos” y que ese listado se escribiese en el archivo c:/listado.txt. Se ha de mostrar tanto la función de c que lo lanzaría, como el comando lanzado por esta.

**(2,5 puntos) (Resolver la pregunta aproximadamente en 1 línea**)

system(“dir d:\\archivos>c:/listado.txt”);