

1. (1.25 punto) Dado el siguiente código

```

1.  #include <stdio.h>
2.  #define BUCLE 0
3.  #define DEBUG 1
4.  int main () {
5.      int i, x = 0;
6.      #ifdef BUCLE
7.          while (x<5000) {
8.              for (i=0; i<100; i++) {
9.                  x+= 1;
10.                 #if DEBUG
11.                     printf("X= %d\n",x) ;
12.                 #endif
13.             }
14.         }
15.     #endif
16.     return 0;
17. }
```

Ejercicio	Puntos
1	
2	
3	
4	
5	
6	
Total	

- Indique qué imprime.
- Para cada afirmación indique si es verdadera o falsa y **justifique**.
 - Al cambiar la línea 2 por “#define BUCLE 1” se compilarán más instrucciones.
 - Al cambiar la línea 3 por “#define DEBUG 0” se compilarán menos instrucciones.
 - Al cambiar la línea 10 por “#ifndef DEBUG” se compilarán menos instrucciones.

2. (1 punto) Defina una macro que dados tres caracteres retorne SI cuando al menos dos de ellos correspondan a un operador de suma, resta, multiplicación o división (“+”, “-”, “*” o “/”) y NO en caso contrario. Luego escriba un programa que utilice la macro para imprimir lo siguiente:

En el conjunto {0,1,+} NO hay al menos 2 operadores
 En el conjunto {4,+, /} SI hay al menos 2 operadores
 En el conjunto {9,8,7} NO hay al menos 2 operadores
 En el conjunto {*, /, +} SI hay al menos 2 operadores

Nota: En cada línea las palabras SI o NO son el resultado de invocar a la macro. El resto del texto es fijo.

3. (1 punto) Marque con verdadero o falso las siguientes afirmaciones:

a)	Es posible retornar un puntero a un bloque de memoria asignada dentro de una función.
b)	Es posible reservar memoria en tiempo de ejecución para un tipo de dato compuesto.
c)	Suponiendo que <code>sizeof(int)=4</code> , el siguiente código imprime “tamaño = 400 bytes”: <code>int ** matriz = calloc(100, sizeof(int));</code> <code>printf("tamaño = %d bytes", sizeof(matriz));</code>
d)	Si sólo se va a leer un archivo binario es lo mismo abrirlo con modo “ab” que con “a+b”..
e)	Puede utilizarse la función <code>fprintf()</code> tanto para escribir en un archivo como para escribir en pantalla.
f)	Si <code>calloc</code> no tiene éxito al reservar memoria, su comportamiento es indeterminado.
g)	Si se graba la misma información en un archivo binario y en un archivo de texto, el primero siempre tendrá menor tamaño.

4. **(1.25 puntos)** Argumentos.

Dado el siguiente código:

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

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

    if (argc > 5) {
        printf("%s %d \n", argv[1], argc / atoi(argv[4]));
        printf("%s %d \n", argv[3], argv[2][1] - argv[2][0]);
    }else{
        printf("%s %d \n", argv[1], argc);
        printf("%s %d \n", argv[3], argv[1][0] - argv[1][1]);
    }
    return 0;
}
```

Indicar que imprime si luego de compilar el programa es ejecutado de la siguiente forma:

- a) C:\>proceso.exe Modulo 02 Semestre 2 end
- b) C:\>proceso.exe edicion 2019 info UNLP

5. Se dispone de un archivo binario con información de alumnos de la UNLP. Para cada alumno se ha registrado: nombre de la carrera (20 caracteres), legajo (10 caracteres), nombre y apellido (30 caracteres) y promedio (float).

- a) **(2 puntos)** Escriba una función que genere un archivo de texto con la información correspondiente a los alumnos de una carrera dada. La función recibirá 3 parámetros: un puntero a FILE correspondiente al archivo binario, el nombre del archivo de texto a generar y el nombre de la carrera a la que pertenecerán los alumnos seleccionados. En el archivo de texto, los datos de cada estudiante (legajo, nombre y apellido y promedio) deben aparecer en una línea separados por coma.
- b) **(1 punto)** Implemente un programa que abra el archivo binario y genere el archivo de texto usando la función definida en a). Los nombres del archivo binario y del archivo de texto así como el de la carrera se ingresan por teclado. Si al finalizar el archivo de texto queda vacío (es decir que en el archivo binario no había ningún alumno de la carrera ingresada) debe mostrar el texto “SIN ALUMNOS”.

6. **(2.5 puntos)** Memoria dinámica.

Se desea procesar la información climática de cada día del año 2022. Para cada día se tienen 3 valores de tipo float: temperatura mínima, temperatura máxima y humedad.

Realice un programa que cargue la información en una matriz dinámica leyendo los valores de temperatura mínima, temperatura máxima y humedad desde teclado. Una vez inicializada la estructura, realice un módulo que imprima para cada mes: el nombre del mes y el número de día con mayor temperatura. Por último, libere la memoria reservada.

Nota: La reserva de memoria, la carga de información, el procesamiento y la liberación de memoria deben modularizarse.