Quiz 3 – Programación en C Parte 1

1 – ¿Cuántos valores tiene el siguiente arreglo?

unsigned char temp[10];

1. 9
2. 10
3. 11
4. 8
5. 1

2 - ¿Cuánto espacio en memoria en bytes ocupa el arreglo siguiente?

signed long a[] = {‘A’, 95, 0};

1. 3
2. 24
3. 12
4. 8

3 - ¿Cuál es el resultado de evaluar la siguiente expresión en C?

z = 5;

y = 2;

x = 124/y % z \* 4;

1. 5
2. 0
3. 8
4. 4

4 – Al transformar 456 a binario, que bits están encendidos (en 1)

1. 1, 6, 7, 8
2. 2, 7, 8, 9
3. 0, 5, 6, 7
4. 1, 2, 4, 8

5 - ¿Porqué el siguiente código no compila?

#include <stdio.h>

int main(void)

{

int a, b, c;

a = 1;

b = 2;

c = suma(a,b);

printf("%i", c);

}

a) Hace falta la declaración y la definición de la función suma

b) Debería ser void main(void) en vez de int main(void)

c) Hacemos el llamado a una función suma que no existe en ese archivo

d) No tenemos un argumento de retorno en la función main

6 - ¿Porqué nos da error este código?

#include <stdio.h>

#include <math.h>

int main(void)

{

printf("%f", sin(0.707));

}

1. Porque al función sin() no está definida en el módulo
2. Porque opera en grados y no en radianes
3. No da error, esta correcta
4. No tenemos la definición de la función

7 - ¿Cuántos es el TOTAL DE BYTES EN MEMORIA de esta cadena?

unsigned char a = “MCU”

1. 4
2. 3
3. 5
4. 3.5

8 - ¿Cuál es el valor final de x?

#include <stdio.h>

#include <math.h>

int main(void) {

int i = 0;

int j, k;

char x;

for (j = 0, k = 1; i < 10; i++) {

switch(i) {

case 5:

case 6:

case 7:

x = i + j;

case 1:

x <<= 1;

break;

default:

x--;

break;

}

}

printf("%i", x);

}

1. 12
2. 13
3. 11
4. 15

9 – ¿Cuál de los siguientes enunciados es falso para macros en C?

1. Un macro puede definir constantes
2. Se pueden realizar funciones cortas
3. Siempre debe finalizarce por ;
4. Usa directivas del preprocesador

10 - ¿Cúantos son los caracteres ASCII?

1. 255
2. 127
3. 128
4. 256