|  |  |
| --- | --- |
| **Apellidos, Nombre:** | RUIZ FERNANDEZ, ANGEL |
| **DNI:** | 23836363Z |
| **Puntuación (rellenar sólo en autoevaluación):** | TEST:: 2 P1:: 1.6 P2:: 2.5 P3:: 2.1 TOTAL: 8.2 |

**Tarea 2 de Fundamentos de Computadores**1° curso de Grado en Ingeniería Informática  
Fecha de entrega: *16 de octubre de 2023*

**Test (2.5 puntos; 0.25 por pregunta)** Rellene la siguiente tabla con la respuesta correcta a las preguntas de test que siguen. Escriba para ello **una X** en la celda correspondiente a cada respuesta correcta. Cada pregunta tiene una y sólo una respuesta correcta (una X para cada una de las columnas T1-T10). Cada 3 respuestas incorrectas anularán 1 correcta.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **T1** | **T2** | **T3** | **T4** | **T5** | **T6** | **T7** | **T8** | **T9** | **T10** |
| **a** |  |  | X |  |  |  |  | X | X |  |
| **b** | X | X | X | X |  | X | X |  |  |  |
| **c** |  |  |  |  | X |  |  |  |  | X |
| **d** |  |  |  |  |  |  |  |  |  | X |

**T1.** Supongamos que un determinado idioma posee un alfabeto de 95 símbolos distintos. El número mínimo de bits necesarios para codificar dicho alfabeto sería:  
 a) Ninguna de las otras tres respuestas es correcta.  
 b) 7  
 c) 11  
 d) 14

**T2.** El número 010001101011)2 expresado en base 2 se corresponde con:   
 a) El número 46C)16 expresado en base 16.  
 b) El número 2153)8 expresado en base 8.  
 c) Ninguna de las otras tres respuestas es correcta.  
 d) El número 1130)10 expresado en decimal.

**T3.** El número 1000 0000 0111 0101 0000 0000 0000 0000 en formato IEEE 754 de simple precisión representa:  
 a) Un número negativo con un valor absoluto muy grande.  
 b) Un número negativo denormalizado.  
 c) NaN (Not a Number).  
 d) Menos infinito.

**T4.** Al ser pasado a binario, el número hexadecimal 3976.A5)16 tendrá:  
 a) 13 bits significativos (e.d., sin contar los posibles bits cero iniciales) en la parte entera, y 8 bits significativos (e.d., sin contar los posibles bits cero finales) en la parte fraccionaria.  
 b) 14 bits significativos (e.d., sin contar los posibles bits cero iniciales) en la parte entera, y 8 bits significativos (e.d., sin contar los posibles bits cero finales) en la parte fraccionaria.  
 c) 14 bits significativos (e.d., sin contar los posibles bits cero iniciales) en la parte entera, y 7 bits significativos (e.d., sin contar los posibles bits cero finales) en la parte fraccionaria  
 d) Ninguna de las otras tres respuestas es correcta.

**T5.** El ASCII es:  
 a) Un sistema de representación numérica para enteros en base dos, con longitud de 7 bits.  
 b) Una codificación para números reales de 16 bits.  
 c) Un código de entrada / salida para representación de caracteres.  
 d) Ninguna de las otras tres respuestas es correcta

**T6.** Sean los dos números expresados en hexadecimal 6291)16 y 9ABB)16 . El resultado de hacer el OR lógico, bit a bit en posiciones correspondientes, de ambos será (expresado también en hexadecimal):  
 a) FD6E  
 b) FABB  
 c) 0291  
 d) 0544

**T7.** El mayor número positivo que puede representarse en 20 bits es (usando representación en complemento a 2):   
 a) 524288  
 b) 524287  
 c) 1048575  
 d) Ninguna de las otras tres respuestas es correcta.

**T8.** Elige la única frase correcta de entre las siguientes:  
 a) El rango correspondiente a una representación entera sesgada que utiliza 9 bits siempre tendrá 512 valores distintos, independientemente del sesgo utilizado.  
 b) El rango correspondiente a una representación entera sesgada que utiliza 9 bits es siempre [-256;255], independientemente del sesgo utilizado.  
 c) El rango correspondiente a una representación entera sesgada que utiliza 9 bits es siempre [-512;511], independientemente del sesgo utilizado.  
 d) Ninguna de las otras tres respuestas es correcta.

**T9.** Dado un texto íntegramente en inglés, indicar cuál de las siguientes afirmaciones es la única correcta:  
 a) El espacio en bytes que ocupará al ser guardado en un fichero será idéntico si se usa codificación ISO-Latin1 que usando codificación UTF8.  
 b) No es posible almacenar dicho texto en codificación ASCII.  
 c) No es posible almacenar dicho texto en codificación UTF16.  
 d) El espacio en bytes que ocupará al ser guardado en un fichero será mayor al utilizar codificación UTF8 que al utilizar codificación ISO-Latin1.

**T10.** Supongamos que disponemos de un alfabeto de 32 símbolos distintos, utilizado para expresar números en base 32:  
 a) Convertir números entre dicha base y secuencias de bits será muy fácil, ya que se trata de un código intermedio equivalente a 16 bits por cada dígito empleado.  
 b) Convertir números entre dicha base seleccionada secuencias de bits será muy fácil, ya que se trata de un código intermedio equivalente a 4 bits por cada dígito empleado.  
 c) Ninguna de las otras tres respuestas es correcta.  
 d) Convertir números entre la base seleccionada y secuencias de bits no será trivial, ya que en general tendremos que emplear primero una conversión a decimal y después la conversión a bits mediante divisiones sucesivas.

**P1. (1.6 puntos; 0.4 por apartado)** Determina el **mínimo** número n de bits necesarios para poder codificar los números enteros 95 y -68 en C2 y, una vez determinada dicha longitud, codifica ambos números en C2, realiza su suma binaria, y comprueba numéricamente que el resultado se corresponde con el resultado correcto (27) en C2 de n bits. Rellena entonces la siguiente tabla:

|  |  |
| --- | --- |
| **Pregunta** | **Respuesta** |
| **Número mínimo de bits n:** | 8 |
| **Representación en C2 de esa longitud del número 95 (ristra de bits):** | 01011111 |
| **Representación en C2 de esa longitud del número -68 (ristra de bits):** | 10111100 |
| **Representación en C2 del resultado de la suma binaria (ristra de bits):** | (1)00011011 = 27  00011011 |

**P2. (3.5 puntos; 0.25 por apartado)** Se tiene un fichero de datos que sigue el convenio *little-endian*, y cuyo volcado hexadecimal, siguiendo el mismo formato utilizado por okteta, es el siguiente:

0000:0000 81 02 00 00 20 41 4e 47 45 4c 20 52 55 49 5a 20 |.... ANGEL RUIZ |  
0000:0010 46 45 52 4e 41 4e 44 45 5a 00 40 f8 c3 |FERNANDEZ.@øÃ |

*a)* Sabiendo que dicho archivo contiene exactamente, y por este orden: 1) un entero de 32 bits codificado en C2; 2) una cadena de carateres; y 3) un número real codificado en IEEE-754 de simple precisión; contestar adecuadamente las preguntas expresadas en la tabla siguiente:

|  |  |
| --- | --- |
| **Pregunta** | **Respuesta** |
| **Desplazamientos comienzo-final del valor entero:** | 0000:0000-0000:0003 |
| **Valor entero representado (entero con signo, en decimal):** | 641 |
| **Desplazamientos comienzo-final de la cadena de caracteres:** | 0000:0004-0000:0018 |
| **Código ASCII del primer carácter de la cadena (en hexadecimal):** | 20  0x20 |
| **Carácter representado en esa posición, entre comillas (ejemplo, 'A'):** | ‘A’  ‘ ‘ Creía que se refería a posición en el archivo, el principio de la cadena, no carácter ASCII correspondiente |
| **¿En qué código (o códigos) de representación de caracteres crees que podría estar representada la cadena? Justifica brevemente tu respuesta:** | El minimo estandar que representa correctamente esos valores como esos simbolos es ASCII de 7-bits.  Como no se usa el octavo bit, sería correcto decir que también es ANSI o ISO-Latin1 o Windows-1252 o UTF-8 etc valido.  ASCII, ISO-Latin1 ó UTF-8 (cualquiera de los tres), si no hay Ñ en el nombre / sólo ISO-Latin1, si sí que hay alguna Ñ en el nombre |
| **Código UTF8 de la primera vocal del nombre (A), si ésta fuese acentuada (Á) (dos bytes, ejemplo 0xC3 0xB1):** | 0xC3 0x81 |
| **Desplazamientos comienzo-final del número real:** | 0000:0019-0000:0022  0000:0019-0000:001c Despieste mio |
| **Bit de signo del número real:** | 1 |
| **Campo exponente del número real (ristra de bits):** | 10000111 |
| **Exponente E del número real (entero con signo, en decimal):** | 8 |
| **Mantisa M del número real (ristra de bits, formato 1.m, ejemplo 1.01):** | 1.11110000100000000000000  1.111100001 |
| **Valor real representado (en decimal, ejemplo -17.25):** | 496.5  -496.5 Se me olvidó el signo |
| **Indicar los 8 bytes que habría que añadir al fichero para añadir ese mismo número real, pero codificado en IEEE754 de doble precisión (seguir con convenio little endian, ejemplo: 0x00 0x00 0x00 0x00 0x00 0x00 0xBF 0xB0)** | 0x00 0x00 0x00 0x60 0x18 0x14 0x08 0x70  0x00 0x00 0x00 0x00 0x00 0x08 0x7f 0xc0 (0xc07f080000000000) |

**Notas importantes:** Expresar los desplazamientos comienzo (primer byte) - final (último byte) solicitados en formato hexadecimal como el utilizado por okteta (ejemplo: 0000:0004-0000:0007). Expresar el resto de números solicitados (enteros y/o reales) en decimal, binario o hexadecimal, según se solicite en cada apartado, y atendiendo siempre al ejemplo mostrado en cada caso.

**P3. (2.4 puntos; 0.3 por apartado)** Se tiene una imagen almacenada en un archivo .ppm cuyo volcado hexadecimal, siguiendo el mismo formato utilizado por okteta, es el siguiente (no se muestra el contenido completo, sólo los primeros bytes):

0000:0000 50 36 0a 23 20 49 6d 61 67 65 6e 20 70 61 72 61 |P6.# Imagen para|  
0000:0010 20 20 41 4e 47 45 4c 20 52 55 49 5a 20 46 45 52 | ANGEL RUIZ FER|  
0000:0020 4e 41 4e 44 45 5a 0a 37 37 20 33 36 0a 32 35 35 |NANDEZ.77 36.255|  
0000:0030 0a ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff |.ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ|  
0000:0040 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff |ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ|  
0000:0050 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff |ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ|  
0000:0060 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff |ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ|  
0000:0070 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff |ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ|  
... ... ...

Se pide contestar a las siguientes cuestiones:

|  |  |
| --- | --- |
| **Pregunta** | **Respuesta** |
| **¿Cuál es la anchura, en píxeles, de la imagen? (en decimal)** | 77 |
| **¿Cuál es la altura, en píxeles, de la imagen? (en decimal)** | 36 |
| **¿Cuál es el tamaño exacto de la cabecera, en número de bytes ? (en decimal)** | 49 |
| **¿Cuál es el (único) color presente en la imagen?** | Blanco |
| **¿Cuál es el desplazamiento donde empieza el primer píxel (arriba a la izquierda de la imagen)?** | 0000:0031 |
| **¿Cuál es el desplazamiento donde empieza el pixel de coordenadas (fila,columna) = (24,57)?** | 0000:5764  0000:1684 Despiste: no lo convertí a hexadecimal |
| **¿Cuáles son las coordenadas (fila,columna) del pixel que empieza en el desplazamiento 0000:1a59?** | (28, 76) |
| **¿Cuál es el tamaño total del archivo, en número de bytes ? (en decimal)** | 8365 |

**Notas importantes:** Expresar todos los desplazamientos solicitados en formato hexadecimal como el utilizado por okteta (ejemplo: 0000:002f). Suponer que las coordenadas (fila,columna) empiezan a numerarse desde cero; es decir, que las coordenadas del primer pixel (arriba a la izquierda) serían exactamente (0,0).