1. **[1,25 ptos.]**Se tiene la siguiente secuencia de caracteres EBCDIC. Su codificación hexadecimal representa una cadena de 2 BPF c/signo de 16 bits. (sin superposiciones)

@ C 1 h

Se pide:

* 1. **[0,5 ptos.]** Indicar cuáles son los números almacenados en base 10 en cada BPF c/signo.
  2. **[0,75 ptos.]** Restar los números obtenidos en a) en formato BPF c/signo de 16 bits indicando la validez del resultado.

1. **[0,75 ptos.]**Se dispone de un formato de representación de números flotantes basado en IEEE 754 en el cual se utilizan 2 bytes para almacenar: el signo (1 bit), el exponente en exceso como BPF s/signo (6 bits) y los dígitos de la mantisa (9 bits) normalizados en base 2.

Signo (1 ))bit)



Mantisa (9 bits)



Exp. en exceso (6 bits)



Dado A = 9613 que representa la configuración decimal de un número almacenado en formato empaquetado de 2 bytes, almacenar dicho número en el formato flotante propuesto indicando su configuración octal.

1. **[2,5 ptos.]** Se tiene una lista (L) cuya dirección de inicio se encuentra almacenada en la celda 20016 en donde cada nodo representa el balance de las cuentas bancarias. Cada nodo de la lista (L) está formado por 3 celdas contiguas en memoria:

* La primera celda contiene el número de la cuenta del cliente.
* La segunda celda contiene el saldo de la cuenta del cliente.
* La última celda contiene la dirección del siguiente nodo de la lista. El final de la lista (L) se indica con un valor -1 en la última celda del último nodo.

Se pide realizar un programa ABACUS con punto de carga en la celda 30016 que recorra la lista (L) y almacene en un vector (V) que comienza en la celda 10016, el número de las cuentas cuyo saldo sea superior al valor almacenado en la celda 20116, marcando el final del vector con el valor -1.

Datos de la máquina: L(RPI) = 12 bits; L(CO) = 4 bits

Código de operaciones:

0 - Carga inmediata 3 – Suma 8 – Bifurcación si (AC) < 0

1 – Carga 4 – Not (AC) 9 – Bifurcación si (AC) > 0

2 – Almacenar 7 – Bifurcación si (AC) = 0 F – Fin de programa

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

1. **[2 ptos.]** Dado A que representa la configuración octal de un número almacenado en formato empaquetado de 3 bytes y B que representa la configuración en base 4 de otro número almacenado en formato BPF C/signo de 16 bits:

A = 44332158 B = 332231204

Se pide:

1. **[0,5 ptos.]** Indicar cuáles son los números almacenados en base 10.
2. **[0,75 ptos.]** Restar ambos números (A-B) en formato BPF c/signo de 16 bits indicando la validez del resultado.
3. **[0,75 ptos.]** Si es posible, expresar el resultado de la resta anterior en formato flotante IEEE 754 de precisión simple dando su configuración hexadecimal.
4. **[2.5 ptos]** Se tiene una lista (L) cuya dirección de inicio se encuentra almacenada en la celda 20016 en donde cada nodo contiene información de una cuenta bancaria. Cada nodo de la lista (L) está formado por 3 celdas contiguas en memoria:

* La primera celda contiene la dirección inicial de un vector de movimientos que finaliza con valor 0.
* La segunda celda contiene el monto inicial de la cuenta.
* La última celda contiene la dirección del siguiente nodo de la lista. El final de la lista (L) se indica con un valor -1 en la última celda del último nodo.

Se pide realizar un programa ABACUS con punto de carga en la celda 30016 que recorra la lista (L) y actualice el monto inicial con los movimientos del vector.