

Guía de Ejercicios Prácticos Abacus

A1. - Dada una máquina Abacus con las siguientes características:

RPI = 12 bits

AC = 16 bits

Operaciones disponibles: (Códigos en hexadecimal)

0	Carga inmediata	7	Bifurca si (AC) < 0
1	Carga	F	Fin de Programa
2	Almacena		
3	Suma		
4	NOT (AC)		
5	Bifurca si (AC) = 0		
6	Bifurca si (AC) > 0		

Punto de carga en la celda $500_{(16)}$.

Determinar el tamaño de todos los componentes de la máquina y realizar un programa que resuelva lo siguiente:

En las celdas $150_{(16)}$ y $250_{(16)}$ se encuentran almacenados dos números X e Y. Efectuar $X - Y$ almacenando el resultado en la celda $300_{(16)}$.

A2. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A1 realizar un programa que resuelva lo siguiente:

En las celdas $150_{(16)}$ y $250_{(16)}$ se encuentran almacenados dos números X e Y. Efectuar $X * Y$ almacenando el resultado en la celda $300_{(16)}$

A3. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A1 realizar un programa que resuelva lo siguiente:

En las celdas $150_{(16)}$ y $250_{(16)}$ se encuentran almacenados dos números X e Y. Efectuar X/Y almacenando el cociente en la celda $300_{(16)}$ y el resto en la celda $301_{(16)}$.

A4. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A1 realizar un programa que resuelva lo siguiente:

En la celda $150_{(16)}$ se encuentra almacenado un número N.

$$\sum_{i=1}^N i$$

Calcular $\sum_{i=1}^N i$ almacenando el resultado en la celda $300_{(16)}$

A5. - Dada una máquina Abacus con las siguientes características:

RPI: 12 bits AC: 16 bits

Operaciones disponibles: (Códigos en hexadecimal)

- 1 Bifurca si el (AC) < 0
- 2 Resta
- 3 Suma
- 4 Almacena
- 5 Carga

En memoria, a partir de la celda $230_{(16)}$ se encuentra el siguiente programa:

0001 5233 2233 4200 5200 3201 4200 5202 3230 4202 1234

Los contenidos de las celdas $201_{(16)}$ y $202_{(16)}$ son $0002_{(16)}$ y $FFFE_{(16)}$, respectivamente. El contenido del RPI es $231_{(16)}$.

Se pide efectuar un seguimiento del programa detallando las microinstrucciones e indicar el estado final de todos los componentes (AC, RI, RPI, RM, RDM) y las celdas que se modifican.

A6. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A1 realizar un programa que resuelva lo siguiente:

En las celdas $100_{(16)}$, $200_{(16)}$ y $300_{(16)}$ se encuentran almacenados tres números X,Y,Z. Se pide ordenarlos en forma ascendente y almacenarlos en ese orden a partir de la dirección $700_{(16)}$.

A7. - Dada una máquina Abacus con las siguientes características:

RPI = 12 bits AC = 16 bits

Operaciones disponibles: (Códigos en hexadecimal)

- 0 Carga inmediata
- 1 Suma
- 2 Almacena
- 3 Fin de programa
- 4 Bifurca incondicionalmente

Punto de carga en la celda $000_{(16)}$.

En memoria, a partir de la celda $000_{(16)}$ se encuentra el siguiente programa:

0005 1003 2003 6FFF 0FFF 1003

Se pide efectuar un seguimiento del programa detallando las microinstrucciones e indicar el estado final de todos los componentes (AC, RI, RPI, RM, RDM) y las celdas que se modifican.

Organización del Computador

A8. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A1 realizar un programa que resuelva lo siguiente:

En la celda $150_{(16)}$ se encuentra almacenado un número que representa la máxima longitud de un BPF c/signo en bits. Almacenar en la celda $300_{(16)}$ el menor número almacenable en esa cantidad de bits y el mayor número almacenable en la celda $301_{(16)}$.

A9. - Dada una máquina Abacus con las siguientes características:

RPI = 12 bits

AC = 16 bits

Operaciones disponibles: (Códigos en hexadecimal)

- 1 Carga
- 2 Almacena
- 3 Suma
- 4 Bifurca si (AC) > 0
- 5 Bifurca incondicionalmente
- 6 Fin de programa

En memoria, a partir de la celda $0FC_{(16)}$ se encuentra el siguiente programa:

5100 8700 2101 6000 1071 3084 40FD 50FF

Los contenidos de las celdas $071_{(16)}$ y $084_{(16)}$ son $BF71_{(16)}$ y $925A_{(16)}$ respectivamente.

Se pide efectuar un seguimiento del programa detallando las microinstrucciones e indicar el estado final de todos los componentes (AC, RI, RPI, RM, RDM) y las celdas que se modifican.

A10. - Dada una máquina Abacus con las siguientes características:

CO = 8 bits

OP = 12bits

Operaciones disponibles (Códigos en hexadecimal)

- | | |
|--------------------|-------------------------------|
| A0 Carga inmediata | A5 NOP |
| A1 Carga | A6 Bifurca si (AC) < 0 |
| A2 Almacena | A7 Bifurca incondicionalmente |
| A3 Suma | A8 Fin de Programa |
| A4 Resta | |

En memoria, a partir de la celda $F31_{(16)}$ se encuentra el siguiente programa:

A0FFA A6F34 00001 A4F37 A7F31 A1F3B A3F33 A2F3B
 A7F3B A3F33 A7F36 A6F32 A8F35

Determinar el tamaño de todos los componentes de la máquina y efectuar un seguimiento del programa detallando las microinstrucciones e indicar el estado final de todos los componentes (AC, RI, RPI, RM, RDM) y las celdas que se modifican.

A11. - Dada una máquina Abacus con las siguientes características:

RPI = 12 bits

AC = 16 bits

Operaciones disponibles: (Códigos en hexadecimal)

0	Carga inmediata	5	Bifurca si (AC) = 0
1	Carga	6	Bifurca si (AC) > 0
2	Almacena	7	Bifurca si (AC) < 0
3	Suma	F	Fin de Programa
4	NOT (AC)		

Punto de carga en la celda $105_{(16)}$.

Las celdas $200_{(16)}$ y $201_{(16)}$ contienen dos direcciones, se pide calcular la sumatoria de todas las celdas comprendidas entre las direcciones dadas.

A12. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio anterior realizar un programa que resuelva lo siguiente:

Dado un vector de cinco elementos (cada elemento es una celda de memoria de la máquina Abacus) almacenados a partir de la dirección $200_{(16)}$, intercambiar de posición al primer elemento del vector con el máximo.

Por ejemplo, si el vector fuera: 4 5 8 7 1, debe quedar: 8 5 4 7 1

A13. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A11 realizar un programa que resuelva lo siguiente:

Dado un vector almacenado a partir de la celda $200_{(16)}$ y que termina en $-1_{(10)}$ se pide invertir al vector, es decir, el último elemento queda en el lugar del primero, (en la celda $200_{(16)}$ queda el $-1_{(10)}$), en el anteúltimo queda el segundo, etc.

A14. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A11 realizar un programa que resuelva lo siguiente:

La celda $200_{(16)}$ contiene una dirección de memoria. La celda $201_{(16)}$ contiene otra dirección de memoria mayor o igual que la anterior. Se pide cambiar unos por ceros y ceros por unos en todas las celdas dentro del bloque delimitado por las dos direcciones mencionadas (inclusive).

A15. - Dada una máquina Abacus con las siguientes características:

CO = 4 bits

OP = 12 bits

Operaciones disponibles: (Códigos en hexadecimal)

0: Carga inmediata	4: Suma 1 al AC.
1: Carga.	5: Bifurca si $AC < 0$
2: Almacenamiento.	6: Resta
3: Suma	7: Bifurca si $AC = 0$
	8: Fin de programa.

Sabiendo que en la dirección $1A0_{(16)}$ se encuentra la dirección de inicio de un vector, cuyo fin está indicado con una celda con valor negativo, se pide realizar un programa Abacus que ubique al principio del vector todos los elementos distintos de cero y al final del mismo los iguales a cero (debe mantener en su lugar la marca de fin de vector).

El punto de carga del programa será la celda $200_{(16)}$.

A16. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A11 realizar un programa que resuelva lo siguiente:

Se sabe que en la dirección $3A0_{(16)}$ se encuentra la dirección de inicio del primer nodo de una lista cuyos nodos están formados por dos celdas contiguas. La primera celda de cada nodo contiene un valor en BPF c/signo de 16 bits y la segunda contiene la dirección de inicio del nodo siguiente. En el último nodo la segunda celda contiene un $-1_{(10)}$ para indicar el fin de la lista. Se pide recorrer todos los nodos de la lista y almacenar la sumatoria de todos los BPF c/signo de 16bits. Dicha sumatoria se deberá almacenar en la celda $400_{(16)}$.

A17. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A11 realizar un programa que resuelva lo siguiente:

Se tiene en memoria un vector formado por elementos de 2 celdas de longitud. En cada elemento la primera celda contiene un valor y la segunda un contenido indefinido inicialmente. El último elemento del vector tiene un valor 0 en la primera celda. Todos los demás elementos del vector tienen como valor en la primera celda números positivos o negativos. Se sabe que en la celda $30A_{(16)}$ se encuentra almacenada la dirección del primer elemento del vector.

Se pide recorrer el vector y "encadenar" los elementos que tengan **números positivos** en sus celdas valor. Este encadenamiento se llevará a cabo poniendo en la segunda celda de cada elemento la dirección del próximo elemento de la cadena. La dirección del primer elemento de la cadena se deberá almacenar en la celda $300_{(16)}$ y en el último elemento de la cadena se deberá dejar un $-1_{(10)}$ en la segunda celda.

Por ejemplo si el vector fuera (suponiendo que comienza en la celda $500_{(16)}$):

(300) XXXX . . . (500) 0FE3 (501) XXXX (502) F33A (503) XXXX (504) 15AB
(505) XXXX (506) 00FA (507) XXXX (508) 0000 (509) XXXX

El encadenamiento resultante sería:

(300) **0500** . . . (500) 0FE3 (501) **0504** (502) F33A (503) XXXX (504) 15AB
(505) **0506** (506) 00FA (507) **FFFF** (508) 0000 (509) XXXX

A18. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A11 realizar un programa que resuelva lo siguiente:

Se tiene en memoria una lista formada por elementos de 2 celdas de longitud. En cada elemento la primera celda contiene un valor y la segunda, la dirección del siguiente elemento. El último elemento de la lista tiene $-1_{(10)}$ como dirección del siguiente elemento. Se sabe que en la celda $330_{(16)}$ se encuentra almacenada la dirección del primer elemento de la lista.

Se pide pasar todos los valores de la lista (es decir las primeras celdas de cada elemento) a un vector que comience en la dirección $400_{(16)}$. Se sabe que los elementos de la lista se encuentran todos entre las celdas $200_{(16)}$ y $500_{(16)}$ y que la cantidad de elementos de la lista es menor a $60_{(10)}$.

Por ejemplo si la lista fuera (suponiendo que la lista comienza en la $220_{(16)}$):

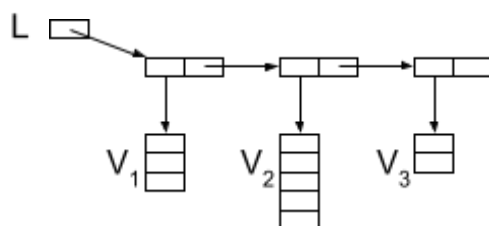
(220) 0001 (221) 0226
(222) 0003 (223) 0224
(224) 0004 (225) FFFF
(226) 0002 (227) 0222

A partir de la dirección $330_{(16)}$ debe quedar:

(330) 0001
(331) 0002
(333) 0003
(334) 0004

A19. - Dada una máquina Abacus con las mismas características y operaciones disponibles que el ejercicio A11 realizar un programa que resuelva lo siguiente:

Se tiene una lista (L) en donde cada nodo contiene una celda que apunta al inicio de un vector de celdas (V) y otra celda que contiene la dirección del siguiente nodo de la lista. El vector de celdas contiene en cada celda un BPF sin signo de 12 bits y termina con un valor 0. A su vez el final de la lista está determinado por un valor $-1_{(10)}$ en la segunda celda del último nodo. Se pide realizar un programa Abacus que recorra todas las celdas de los vectores (V) de todos los nodos de la lista (L) y obtenga el máximo BPF sin signo almacenado. Este valor se deberá dejar almacenado en la celda $200_{(16)}$. La celda $300_{(16)}$ contiene la dirección de inicio de L.



A20. – Dada una máquina Abacus con las siguientes características:

RPI = 12 bits AC = 16 bits

Operaciones disponibles: (Códigos en hexadecimal)

0	Carga inmediata	7	Bifurcación si (AC) = 0
1	Cargar	8	Bifurcación si (AC) > 0
2	Almacenar	9	Bifurcación si (AC) < 0
3	Sumar	F	Fin de Programa
6	Not (AC)		

Se tiene una lista (L) cuya dirección de inicio se encuentra almacenada en la celda 200₍₁₆₎. Cada nodo de la lista (L) está formado por tres celdas contiguas en memoria. La primera celda contiene un número n que representa la dimensión de una matriz cuadrada de n*n. La segunda celda contiene la dirección de inicio de una matriz (M) cuadrada. La tercera celda del nodo contiene la dirección del siguiente nodo de la lista. El final de la lista (L) se indica con un valor -1 en la tercera celda del último nodo.

Se pide realizar un programa ABACUS con punto de carga en la celda 300₍₁₆₎ que recorra todos los nodos de la Lista (L) y calcule la sumatoria de los elementos de la diagonal principal de cada matriz (M) asociada almacenando en la celda 400₍₁₆₎ la mínima sumatoria calculada.

