


### Question 1

Complete

Mark 0.00 out of

1.00

 Flag question

Usando circuitos **builtin** del Simulador de Hardware, construya una ALU que recibe dos entradas de un bit (A y B) , una entrada de control (C) de un bit y dos bits de salida (Y, Z).

De acuerdo al valor de C debe realizar las siguientes operaciones

$C=0$ ,  $Z = A+B$ . Si hay carry  $Y = 1$ , de lo contrario  $Y = 0$

$C=1$ ,  $Z = A \rightarrow B$ ,  $Y=A>B$ , donde  $\rightarrow$  es la implicación lógica


Como respuesta copie el texto del HDL



Question 2

Partially correct

Mark 0.67 out of  
1.00

 Flag question

Dado el fragmento de código

Org 100

Clear

Add B

Store A

InS Rutina

Halt

N, Dec 5

C1, Dec 10

C2, Dec 15

C3, Dec 20

C4, Dec 25

C5, Dec 30

A, Dec 35

B, Hex 106

Rutina, Clear

Store S

Load A

Ciclo, Sub N

Skipcond 800

Jump Fin

Load S

AddI A

Store S

Load A

Add D

Store A



AddI A  
Store S  
Load A  
Add D  
Store A  
Jump Ciclo  
Fin, Jmpl Rutina  
S, Dec 40  
D, Dec 1

Select one or more:

- ☒ a. Esta escrito en lenguaje ensamblador de MARIE ✓
- ☒ b. Se queda en ciclo infinito ✓
- ☐ c. Tiene errores de sintaxis y de logica
- ☐ d. Calcula la suma de los elementos del vector
- ☐ e. Esta escrito en lenguaje de maquina MARIE


The correct answers are: Esta escrito en lenguaje ensamblador de MARIE, Tiene errores de sintaxis y de logica, Se queda en ciclo infinito

**Question 3**

Complete

Mark 1.00 out of

1.00

 Flag question

Escriba en lenguaje ensamblador de Marie el código equivalente al fragmento de código

```
if X = 1 then
  if X < 10 then
    Y := X + X
  endif
endif
```

```
ORG 100
IF,   LOAD X
      SUBT ONE
      SKIPCOND 400
      JUMP ENDIF
IF2,  LOAD X
      SUBT TEN
      SKIPCOND 000
      JUMP END IF
THEN, LOAD X
      ADD X
      STORE X
      STORE Y
ENDIF, HALT
X, DEC 0
Y, DEC 0
ONE, DEC 1
TEN, DEC 10
```



Question 4

Complete

Mark 1.00 out of

1.00

Flag question

Escriba en lenguaje ensamblador de **Marie** el código equivalente al fragmento de código

```
X := 1;  
while X <= 10 do  
    X := X + 1  
endwhile
```

```
ORG 100  
LOAD ONE  
STORE X  
WHILE,    LOAD X  
          SUBT TEN  
          SKIPCOND 000  
          JUMP ENDWHILE  
          LOAD X  
          SUBT TEN  
          SKIPCOND 400  
          JUMP END WHILE  
          LOAD X  
          ADD ONE  
          STORE X  
          JUMP WHILE  
ENDWHILE, HALT  
X, DEC 0  
ONE, DEC 1  
TEN, DEC 10
```



Question 5

Complete

Mark 0.50 out of

1.00

Flag question

Escriba en lenguaje ensamblador de **MARIE** un programa que evalúe la expresión  $Z = (A/2) + (B/2) + (C/2)$ . Implemente y utilice la función `div(X,Y)` que retorna en el acumulador la división entera de dos números positivos X e Y

```
ORG 100
DIV, CLEAR
STORE X
STORE Y
LOAD ZERO
STORE K
IF, LOAD X
SKIPCOND 000
JUMP ENDIF
SKIPCOND 400
JUMP END IF
THEN, LOAD X
SUBT Y
STORE X
JUMP IF
```

```
ENDIF, LOAD K
ADD ONE
STORE K
HALT
P5,    LOAD A
      JNS DIV(A,TWO)
      STORE A
      LOAD B
      JNS DIV(B,TWO)
      STORE B
      LOAD C
      JNS DIV(C,TWO)
      STORE C
      LOAD Z
      ADD A
```

```
      ADD B
      ADD C
      STORE Z
      JMPL DIV
ENDP5, CLEAR
      HALT
Z, DEC 0
A, DEC 20
B, DEC 10
C, DEC 5
K, DEC 0
X, DEC 0
Y, DEC 0
TWO, DEC 2
ONE, DEC 1
```

ZERO, DEC 0