

### Corrigé (Réponses attendues)

Architecture matérielle, langages de haut niveau, langage assembleur et langage machine, portes et circuits logiques

**Question 1** – En langage assembleur, il existe une instruction qui permet de charger une valeur située à un emplacement (adresse) de la RAM dans l'accumulateur. Mais où se situe l'accumulateur dans l'architecture d'une machine ?

L'accumulateur se trouve dans l'Unité Arithmétique et Logique (UAL) du Processeur (CPU).

**Question 2** – Quelle différence entre une instruction écrite en langage assembleur et la même instruction écrite en langage machine ?

*En langage assembleur et langage machine, instruction composée des mêmes éléments : code opération et opérands.*

Mais en langage assembleur, **instruction écrite en mode texte et nombre en base 10 ou 16**, donc compréhensible par un humain ,

alors que en langage machine, **instruction écrite en binaire**, donc incompréhensible par un humain.

**Question 3a** – Les instructions de certains langages de haut niveau sont traduites en langage machine par un logiciel appelé 'interpréteur'. Donner un exemple de langage de haut niveau qui est 'interprété'.

Python – Javascript - Php

**Question 3b** – Les instructions de certains langages de haut niveau comme Pascal ne sont pas traduites en langage machine par un logiciel appelé 'interpréteur' mais par un logiciel qui porte un autre nom : lequel ?

Compilateur

**Question 4a** – Compléter le tableau ci-dessus en indiquant dans la colonne de droite le contenu de l'accumulateur quand l'instruction machine est exécutée.

**Question 4b** – Écrire les valeurs contenues aux adresses RAM de 018 à 022 à la fin du programme (remplacer les valeurs déjà écrites par d'autres si besoin).

Adresse RAM	Instruction machine		Accumulateur
001	LOAD	\$12	12
002	SAVE	#020	12
003	LOAD	#021	5
004	CMP	#020	1
005	CAZ		1
006	JMP	#008	1
007	HLT		1
008	LOAD	#021	5
009	QUO	#020	2
010	SAVE	#018	2
011	LOAD	#021	5
012	MOD	#020	2
013	SAVE	#019	2
014	HLT		2
016			
017			
018	2		
019	2		
020	12		
021	5		

**Question 4c** – Si l'on remplace l'instruction située à l'adresse 001 en RAM par celle-ci : LOAD \$4, expliquer ce qui va se passer dans la suite du programme

Les instructions machines situées aux adresses 001, 002 et 003 vont s'exécuter et donc on aura la valeur 4 à l'adresse 020 et 5 à l'adresse 021.

L'instruction située à l'adresse 004 va placer 0 dans l'accumulateur puisque la valeur de l'accumulateur 5 est supérieure à celle de l'adresse mémoire 020 qui est 4.

L'instruction située à l'adresse 005 va donc passer l'instruction située à l'adresse 006 et continuer à l'adresse 007.

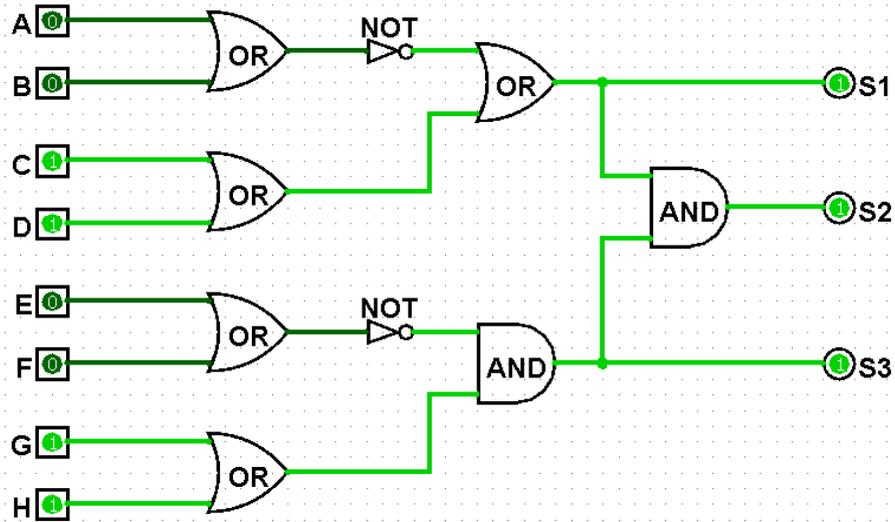
Et comme l'instruction à cette adresse est HLT le programme va s'arrêter.

**Question 5** - Écrire ce programme dans un langage comparable à de l'assembleur en utilisant les opérations définies dans le tableau donné en page 1.

Adresse RAM	Instruction machine		Commentaire
001	LOAD	\$2	Correspond à $x \leftarrow 2$ , $x$ représentant l'adresse mémoire 020
002	SAVE	#020	
003	LOAD	\$3	Correspond à $y \leftarrow 3$ , $y$ représentant l'adresse mémoire 021
004	SAVE	#021	
005	LOAD	\$4	Correspond à $z \leftarrow 4$ , $z$ représentant l'adresse mémoire 022
006	SAVE	#022	
007	CPZ	#022	Compare la valeur de $z$ avec zéro
008	JMP	#010	Cas où $z \neq 0$ : on continue à l'instruction située à l'adresse 010
009	HLT		Cas où $z == 0$ : on arrête le programme
010	LOAD	#020	Correspond à $x \leftarrow x * y$
011	MUL	#021	
012	SAVE	#020	
013	LOAD	#020	Correspond à $y \leftarrow x + y$
014	ADD	#021	
016	SAVE	#021	
017	DEC	#022	Correspond à $z -= 1$ (décrémentaion)
018	JMP	#007	Continuer le programme à l'adresse 007
019			
020		2	
021		3	
022		4	

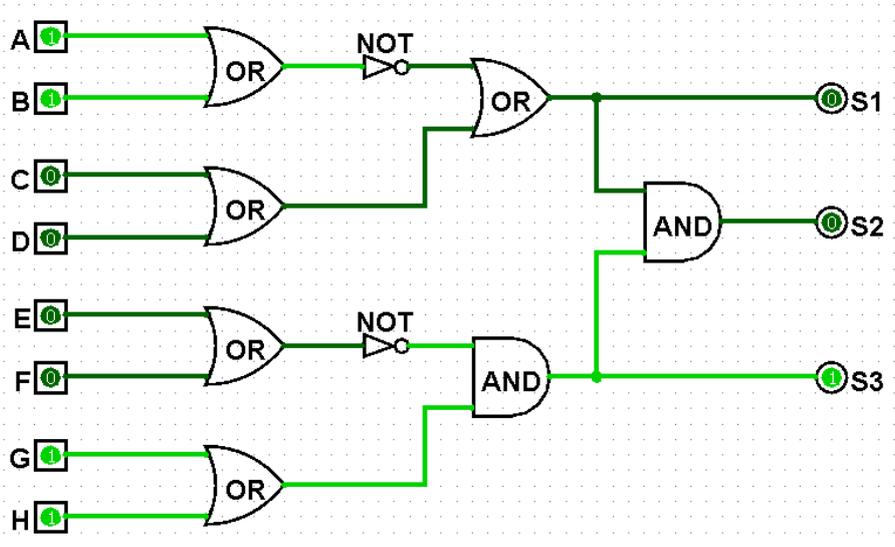
**Question 6** - Indiquer quel sera l'état de chacune des sorties pour les deux états suivants des entrées :

État n°1



A	B	C	D	E	F	G	H		S1	S2	S3
0	0	1	1	0	0	1	1		1	1	1

État n°2



A	B	C	D	E	F	G	H		S1	S2	S3
1	1	0	0	0	0	1	1		0	0	1

NOM Prénom Classe : \_\_\_\_\_

<b>QUESTIONS</b>	<b>Note obtenue</b>	<b>Barème</b>	<b>Observation</b>
1		<b>1</b>	
2		<b>1</b>	
3a		<b>1</b>	
3b		<b>1</b>	
4a		<b>1</b>	
4b		<b>1</b>	
4c		<b>1</b>	
5		<b>1</b>	
6		<b>2</b>	Réponse fausse si une des trois sorties est incorrecte
Note finale		<b>10</b>	