

1 Problème I - Connaissance du processeur ARM (.../1.5pt)

Q.1.1) Au cycle i , nous exécutons l'instruction située à l'adresse 0x00001000. Quelle est la valeur du PC (Program counter) à la fin de ce cycle.

Q.1.2) Quelle opération devons nous réaliser pour effectuer le retour d'une routine.

Q.1.3) Quel est le contenu de R1 et de R0 après l'exécution de l'instruction LDR R1, [R0, #4]. Nous avons les valeurs suivantes en mémoire et dans les registres :

- Registres
 - R0 : 0x00001004
 - R1 : 0x0000100C
- Mémoire (adresse : contenu) :
 - 0x00001000 : 1
 - 0x00001004 : 2
 - 0x00001008 : 3
 - 0x0000100C : 4
 - 0x00001010 : 5

2 Problème II - Développement de code assembleur pour le processeur ARM (.../5pt)

Soit A une matrice carrée de taille $N_A \times N_A$. L'objectif de cet exercice est de calculer la trace de cette matrice ($Tr()$) et de construire le vecteur représentant la diagonale de la matrice A . La trace d'une matrice correspond à la somme des éléments de la diagonale de la matrice.

Considérons l'exemple suivant, la matrice A est définie ci-dessous :

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 10 & 11 & 12 & 13 \\ 20 & 21 & 22 & 23 \\ 30 & 31 & 32 & 33 \end{pmatrix} \quad (1)$$

La trace de cette matrice A est égale à :

$$Tr(A) = \sum_{i=0}^{N_A-1} A_{ii} = 1 + 11 + 22 + 33 = 67 \quad (2)$$

Le vecteur représentant la diagonale de cette matrice A est le suivant :

$$v = (A_{00} \ A_{11} \ \dots \ A_{N_A-1 \ N_A-1}) = (1 \ 11 \ 22 \ 33) \quad (3)$$

Les conventions suivantes sont utilisées pour l'interface avec cette routine :

- Entrée :
 - r0 : adresse de la matrice A
 - r1 : adresse du vecteur v
 - r2 : nombre de lignes de la matrice A
- Sortie :

— r0 : trace de la matrice A.

Nous disposons du code suivant pour tester cette routine (AnalyseSimilitude) permettant de déterminer la similitude entre deux vecteurs :

```
#define Na      4      /* Nombre de lignes et de colonnes de A */

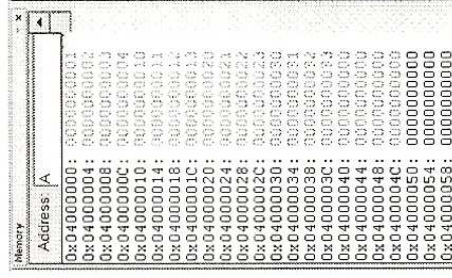
int v[Na]; int A[Na][Na] = {{1,2,3,4},{10,11,12,13},{20,21,22,23},{30,31,32,33}};

extern ??? Trace(???);

int main(void)
{
    Trace(???);

    while(1);
}
```

La zone mémoire contenant la matrice A est présentée à la figure 1.



Address	Value
0x04000000	00000001
0x04000004	00000002
0x04000008	00000003
0x0400000c	00000004
0x04000010	0000000a
0x04000014	0000000b
0x04000018	0000000c
0x0400001c	0000000d
0x04000020	00000014
0x04000024	00000015
0x04000028	00000016
0x0400002c	00000017
0x04000030	0000001e
0x04000034	0000001f
0x04000038	00000020
0x0400003c	00000021
0x04000040	00000022
0x04000044	00000023
0x04000048	00000024
0x0400004c	00000025
0x04000050	00000026
0x04000054	00000027
0x04000058	00000028

FIG. 1 – Zone mémoire contenant la matrice A

Q.2.4) Donner le prototype C de la routine Trace et l’appel en C de cette routine. Vous utiliserez les conventions de passage de paramètres associées au processeur ARM.

Q.2.5) Écrire le code C de la routine Trace.

Q.2.6) Écrire le code assembleur (avec commentaires) de cette routine en utilisant les conventions définies ci-dessus.

3 Problème III - Analyse de code (.../3.5pt)

Nous considérons une fonction pour laquelle nous allons analyser le code et étudier le contenu de certains registres. Le code obtenu est présenté en annexe.

La zone mémoire associée au code de cette fonction est fournie. La première colonne représente l’adresse des éléments de la mémoire. La seconde colonne représente le contenu de la mémoire. La troisième colonne

représente l'instruction associée à chaque case de la mémoire.

La zone mémoire associée à la pile est fournie sur la partie droite du document. La première colonne représente l'adresse des éléments de la mémoire. La seconde colonne représente le contenu de la mémoire.

Le contenu des registres présents au sein de l'unité de traitement avant l'appel de la fonction est le suivant :

- R0 : 0x00000100
- R1 : 0x00000101
- R2 : 0x00000102
- R3 : 0x00000103
- R4 : 0x00000104
- R5 : 0x00000105
- R10 : 0x00000110
- R11 : 0x00001F6C
- R12 : 0x00001F70
- R13 : 0x00001F34
- R14 : 0x010002D8
- R15 : 0x010001B4

Q.3.7) Compléter le tableau (a) pour connaître le contenu des registres FP(R11), IP(R12), SP(R13), R0, R2, R3.

Q.3.8) Compléter le tableau (b) pour représenter l'évolution du contenu de la pile.

Q.3.9) Dans le cas présent quelle est la valeur (hexa) renvoyée par cette fonction.