

Contrôle continu 1 SUJET A - Durée 15 min

Barème sur 10 : 1 pt pour la comparaison à 0, 2 pour la deuxième comp + jump, 1 pt pour la troisième comparaison, 1 pt pour le add, 2 pts pour l'adresse dans r3, 2 points pour écrire mémoire, 1 pour un print (énoncé flou donc soit r2 dans la boucle, soit r1 à la fin).

L'objet de cet exercice est d'imprimer et compter le nombre de caractères *en minuscule* d'un mot stocké en mémoire, et de stocker ce nombre en mémoire. On utilise r0, a0 pour les adresses d'accès mémoire en lecture, r2 pour le code ascii de la lettre courante, r3, a0 pour l'accès mémoire pour écrire le résultat. On vous fournit le code à remplir et quelques indications.

Solution:

Listing 1 – nbminus.s

```
1      ;;
      lea r0 str          ; the address of label str
strlen:
      leti r1 0
      setctr a0 r0       ; r0 has the parameter @str
6 loop:
      readse a0 8 r2     ; read 8 bits in memory (a char)
      add r2 r2 0
      ;; a0 is incremented by 8 and r2 is the last written register ;
      jumpif z end      ; if the char is 0, end.
11     cmp r2 97         ; if not a regular char, do not inc
      jumpif slt endif
      cmp r2 122        ; same
      jumpif sgt endif
      print char r2
16     add r1 1
endif:
      jump loop
end:
      print signed r1
21     ;; now store in memory
      lea r3 res
      setctr a1 r3
      write a1 16 r1
26 fin:
      jump fin
str:
      .string "He190LLo, 99wOrLd!01"
res:
31     .const 16 #100
```


Contrôle continu 1 SUJET B - Durée 15 min

Barème sur 10 : 1 pt pour la comparaison à 0, 2 pour la deuxième comp + jump, 1 pt pour la troisième comparaison, 1 pt pour le add, 2 pts pour l'adresse dans r3, 2 points pour écrire mémoire, 1 pour un print (énoncé flou donc soit r2 dans la boucle, soit r1 à la fin).

L'objet de cet exercice est d'imprimer et de compter le nombre de caractères *chiffres* d'un mot stocké en mémoire, et de stocker ce nombre en mémoire. On utilise *r0, a0* pour les adresses d'accès mémoire en lecture, *r2* pour le code ascii de la lettre courante, *r3, a0* pour l'accès mémoire pour écrire le résultat. On vous fournit le code à remplir et quelques indications.

Solution:

Listing 1 – nbchiffres.s

```
1      ;;
      lea r0 str          ; the address of label str
strlen:
      leti r1 0
      setctr a0 r0       ; r0 has the parameter @str
6 loop:
      readse a0 8 r2     ; read 8 bits in memory (a char)
      add r2 r2 0
      ;; a0 is incremented by 8 and r2 is the last written register ;
      jumpif z end      ; if the char is 0, end.
11     cmp r2 48         ; if not a digit, do not inc.
      jumpif slt endif
      cmp r2 57         ; same
      jumpif sgt endif
      print char r2
16     add r1 1
endif:
      jump loop
end:
      print signed r1
21     ;; now store in memory
      lea r3 res
      setctr a1 r3
      write a1 16 r1
26 fin:
      jump fin
str:
      .string "He190llo, 99world!01"
res:
31     .const 16 #100
```

