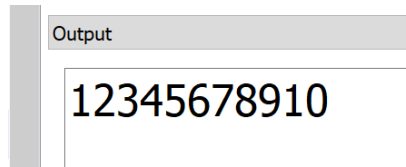


Semaine 6 – Langage machine – Exercices

Exercice n°6.1 – Programme à réaliser en NASM 32 bits

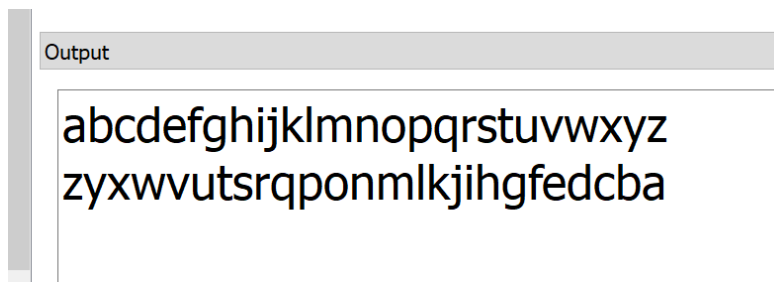
En utilisant l'IDE SASM, écrivez un programme en NASM qui affiche à l'écran les chiffres de 1 à 10 en décimal.



Pour réaliser cet exercice, vous aurez besoin de la macro d'I/O écrite dans la librairie io.inc intitulée PRINT_UDEC. Consultez l'aide (F1) dans SASM.

Exercice n°6.2 – Programme à réaliser en NASM 32 bits

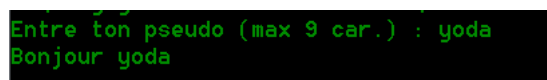
En utilisant l'IDE SASM, écrivez un programme en NASM qui affiche à l'écran les lettres de l'alphabet de a à z et puis de z à a. Allez à la ligne entre les deux affichages.



Pour réaliser cet exercice, vous aurez besoin des macros d'I/O écrites dans la librairie io.inc intitulées PRINT_CHAR et NEWLINE. Consultez l'aide (F1) dans SASM.

Exercice n°6.3 – Programme à réaliser en NASM 32 bits

En utilisant l'IDE SASM, écrivez un programme en NASM qui demande à l'utilisateur d'entrer un pseudo de maximum 9 caractères, l'utilisateur entre son pseudo et ensuite, le programme affiche "Bonjour " suivi du pseudo entré :



Dans le cas où l'utilisateur entre + de 9 caractères, seuls les 9 premiers caractères sont considérés.

Le programme affiche ensuite combien de lettres ont été entrées. Voici quelques cas possibles :

```
I:\Langage machine - 2018-2019\Semaine 6>sol1
Entre ton pseudo (max 9 car.) : jack
Bonjour jack

Nombre de lettres de ton pseudo : 4
I:\Langage machine - 2018-2019\Semaine 6>sol1
Entre ton pseudo (max 9 car.) : jackiechan
Bonjour jackiecha
Nombre de lettres de ton pseudo : 9
I:\Langage machine - 2018-2019\Semaine 6>sol1
Entre ton pseudo (max 9 car.) :
Bonjour
Nombre de lettres de ton pseudo : 0
```

Pour réaliser cet exercice, vous aurez besoin des macros d'I/O écrites dans la librairie io.inc intitulées GET_STRING, PRINT_STRING et PRINT_UDEC. Consultez l'aide (F1) dans SASM.

Exercice n°6.4 – Programme à réaliser en NASM 32 bits

En utilisant l'IDE SASM, écrivez un programme en NASM qui demande à l'utilisateur d'entrer un mot de passe de maximum 10 caractères tant qu'il n'est pas correct :

```
Entre le mot de passe (max. 10 car.) :  
Mot de passe incorrect  
Entre le mot de passe (max. 10 car.) : bonjour  
Mot de passe incorrect  
Entre le mot de passe (max. 10 car.) : vinci2018  
Mot de passe correct :)
```

Votre programme doit vérifier que le mot de passe est bien "vinci2018" pour pouvoir se terminer.