Travaux Pratiques n°4

Architecture des ordinateurs.

—IR 1—

Avant le **lundi 25 Mai à minuit**, vous enverrez une archive contenant les fichiers sources ainsi qu'un rapport au format pdf à votre chargé de td.

Arnaud Carayol : `arnaud.carayol@univ-mlv.fr`
Laurent Braud : `laurent.braud@univ-mlv.fr`

Vous donnerez à votre mail le sujet suivant:

`[Archi L3][TP4] Nom1--Nom2`

Le rapport doit répondre aux questions posées dans le sujet et peut éventuellement contenir des portions de code pour illustrer votre propos. Votre code doit être commenté sinon il ne sera pas lu. Si un fichier que vous rendez n'a pas le comportement attendu, cela doit être dit dans le rapport.

Les questions bonus sont facultatives.

---

1 Crible d'Erathosthène

L'algorithme du *crible d'Eratosthène* permet de calculer tous les nombres premiers inférieurs à certain nombre N.

Initialement on écrit tous les nombres de 1 à N. On efface 1 et l'on souligne 2 qui est le premier nombre premier.

	<u>2</u>	3	4	5	6	7	8	9	10
10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29

Étape 1. On efface tous les multiples du plus grand nombre souligné (dans notre cas 2) sauf ce nombre.

<u>2</u>	3		5		7		9		
11		13		15		17		19	
21		23		25		27		29	

On souligne le premier nombre (non-effacé) suivant le plus grand nombre déjà souligné. Dans notre cas, c'est 3.

<u>2</u>	<u>3</u>		5		7		9		
11		13		15		17		19	
21		23		25		27		29	

Etape 2. On recommence avec le plus grand nombre souligné dans notre cas 3. On efface tous les multiples de 3 et on souligne 5.

$\underline{2}$	$\underline{3}$		$\underline{5}$		7			
11		13				17		19
		23		25				29

Etape 3. On recommence avec le plus grand nombre souligné dans notre cas 5. On efface tous les multiples de 5 et on souligne 7.

$\underline{2}$	$\underline{3}$		$\underline{5}$	$\underline{7}$		9		
11		13			17			19
		23						29

Étape 4 Le processus s'arrête lorsque le plus grand nombre souligné est plus grand que $\sqrt{N}$. Les nombres non-effacés sont les nombres premiers inférieurs à N.

2 Programme en assembleur

Le but de cette séance est de réaliser un programme qui utilise le crible d'Ératosthène pour afficher tous les nombres premiers inférieurs à un certain nombre. Vous pouvez partir du fichier `era.asm`.

Le programme définit une constante `N` qui vaut `100`. Ceci est réalisé avec l'instruction:

```
%define N 100
```

Le programme réserve `N+1` octets à partir de l'adresse `tab`. C'est la ligne:

```
tab resb N+1
```

Nous allons utiliser les `N+1` octets pour mémoriser quels nombres ont été effacés. L'octet à l'adresse `tab+1+i` vaudra `1` si le nombre `i` n'est pas rayé et `0` sinon.

Question 1 Écrivez une fonction `init` qui met tous les octets de la mémoire entre les adresses `tab` et `tab+N+1` à 1.

Question 2 Écrivez une fonction `efface` qui prend un argument `arg` par la pile et qui met à 0 tous les octets à `tab+2*arg`,`tab+3*arg`,...

Question 3 Écrivez une fonction `nouv_premier` qui prend un argument dans `arg` dans la pile et renvoie dans `eax` la première valeur n supérieure à `arg` telle que l'octet à l'adresse `tab+n` vaut 1. Si une telle valeur n'existe pas la fonction renvoie `N`.

Question 4 Écrivez une fonction `affiche` qui affiche toutes les valeurs `i` comprises entre `0` et `N` telles que l'octet à l'adresse `tab+i` vaut 1.

Question 5 Écrivez un programme utilisant les fonctions ci-dessus et affichant les nombres premiers inférieurs à `N`.

Bonus 1 Modifiez votre programme pour qu'il utilise de manière plus efficace la mémoire. Un octet permet de conserver le statut (effacé ou non-effacé) de 8 nombres et non d'un seul.