[SY5]

“

Wieslas Zielonka
zielonka@liafa.univ-paris-diderot.fr
www.liafa.univ-paris-diderot.fr/~zielonka

Introduction

1. Plan du cours

1. Interaction avec le système de fichiers
2. Processus
3. Communication avec les tubes

2. Language et norme

On utilisera le language C dans la norme POSIX et des commandes UNIX

Rappel de C

1. Pointeur

```
int *p-int; //Pointeur
double *p-d;
int x = 5; // Variable
double d = -3,14;
p-int = &x; //Prise d'adresse.
p-d = &d;
int k;
k = *p-int + 5;
*p-int = k + 3;
```

2. Tableau et arithmétique de pointeurs

```
int tab[] = {-3,6,8,12,-7,2};
double td[] = {-3,14,-0.1,-7,12,-8.0,2.43};
```

Table

-3	6	8	12	-7	2

```
int *a;
a = &tab[2];
printf(*(a+2)); // -7
printf(*(a-1)); // 6
int l = -1
printf(*(a+2*l)); // -3
printf(*(a+2)+1); // -6
*(a+15) = 12;
//Sorti d'espace adressable (Seg Fault ou corruption de donneés)
b = &tab[5]
printf(a-b); // entier postive de 3*size(int)
```

3. Parcours de tableau avec pointeur

```
a = &tab;
b = a+6;
int s = 0;
while(a<b){
    s += *a;
    a++;
}
```

Utilisation de `sizeof(-type/-expression)`

4. Petit programme et fonction

```
int f(int x, int t[]){
    int s;
    t[2] = t[3] + 1
    //*(pointeur + entier) <=> pointeur[entier] pour le compilateur
    //sizeof(t) = 4
    ...
}
int main(void){
    int i;
    int tab[] = {1,2,5,8};
    //sizeof(tab) = 4*4
    int k;
    i=3;
    // On doit passer le nombre d'élément du tableau en parametre
    k = f(i,tab);
    ...
}
```

5. Fichier Compilé

1. **Fichiers ".o"**
   Segment Texte -> Les instruction du programme
   Data Segment -> Les variables initialisés de niveau 0 ou plus static ou non.
   BSS segment -> Les variables de niveau 0 non initialisés
   Pile -> Memoire temporaire desendante
   Tas -> Memoire temporaire montante.