

Compilation

TD 4 : Organisation des données

C. ALIAS

Le but de ce TD est de déterminer l'emplacement et la structure des données. Ces informations nous permettront ensuite de construire notre compilateur.

Exercice 1. *Enregistrements d'activation*

On considère le programme suivant :

```
int f(int x)
{
    return x+1;
}

void main()
{
    int result;
    result = f(1);
}
```

Q1) Dérouler l'exécution en dessinant l'état de la pile.

Q2) Donner `[[f]]` et `[[main]]`.

Exercice 2. *Avec des tableaux*

On considère le programme suivant :

```
void copy(int[3] tab_dest, int[2] tab_source, int n)
{
    tab1[n] = tab2[n];
}

void main()
{
    int[2] tab1;
    int[2] tab2;
    tab1[1] = 2;
    copy(tab2, tab1, 1); //tab2[1] <-- tab1[1]
}
```

Q1) Dérouler l'exécution en dessinant l'état de la pile.

Exercice 3. *Avec des pointeurs*

On considère le programme suivant :

```
typedef struct { int re; int im; } complex_t;
typedef struct { complex_t item; list_t* next; } list_t;

list_t* build(int n)
{
    list_t list;

    if(n==0)
        return NULL;
    else
    {
        list.next = build(n-1);
        list.item.re = n;
        list.item.im = n;
        return &list;
    }
}
```

Q1) Donner `[[list_t]]`^{tmp}.

Q2) Dessiner l'état de la pile après l'exécution de `build(2)`. Conclusion ?