Université d'Orléans **Maîtrise d'Informatique**

Feuille de Travaux Dirigés 10

Exercice 1 Produire le code à 2 adresses implantant les instructions C suivantes. Indiquer le code intermédiaire (à trois adresses) produit, ainsi que les descripteurs de registres et d'adresses.

```
while d > 0 do
        a := b + c;
        d := d - b;
        e := a + f;
        if e # 0 then
           f := a - d;
           b := d + f
        else
           e := a - c
        fi;
        b := a + c
 od;
```

Comment procédér si on dispose de 3 registres uniquement ? De manière générale, comment passer d'un code avec un nombre arbitraire de registres à un code avec un nombre de registres prédéfini ?

Exercice 2 Donnez le graphe de contrôle pour une boucle :
for $I := E_1$ **step** E_2 **until** E_3 **do** S **od** avec les deux sémantiques suivantes :

1. La première exécution de S est faite sans contrôle. Puis, E_2 et E_3 sont réévaluées, puis le test
(valeur(I) - valeur(E_3)) * sign(valeur(E_2)) > 0 est évalué.
2. E_1, E_2 et E_3 sont évalués une seule fois avant d'entrer dans la boucle.

Exercice 3
1. Construire le graphe de flot de contrôle $\mathcal{G}$ pour le code du premier exercice.
2. Calculer l'ensemble des variables actives pour chaque bloc, sachant qu'aucune variable n'est utilisée après ce code.
3. Quelles sont les variables que vous avez envie de ranger dans des registres ?

Exercice 4 Proposer une traduction oprimisée pour le programme suivant :

```
while d > h*h do
        a := b + c;
        d := d - b;
        e := a * d;
        g := f
        if e # 0 then
           f := g;
           b := d * 5
        else
           f := a * d
```

```
        fi;
        b := a + c
    od;
```

Exercice 5 Produire le code à 3 adresses et le code implantant le programme C suivant :

```
x = f(a) + f(a) + f(a)
```

Même question pour

```
x = f(f(a))
```