
Exercice 1

Construire le graphe de flot de contrôle $\mathcal{G}$ pour le code suivant.

```
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;
```

Calculer l'ensemble des variables actives pour chaque bloc, sachant qu'aucune variable n'est utilisée après ce code.

Quelles sont les variables que vous avez envie de ranger dans des registres ?

Exercice 2 Optimisation des boucles

Considérons le programme de multiplication de matrices suivant (`N` est une constante, les matrices `a`, `b` et `c` sont allouées statiquement :

```
pour i := 1 a N faire
   pour j := 1 a N faire
      pour k := 1 a N faire
         c[i,j] := a[i,k]*b[k,j]
```

1. Produire le code à trois adresses pour ce programme.
2. Construire un graphe de flot de contrôle à partir des instructions à trois adresses.
3. Éliminer les sous-expressions communes de chaque bloc de base.
4. Trouver les boucles dans le graphe de flot de contrôle.
5. Déplacer les invariants des boucles à l'extérieur de celles-ci.
6. Déterminer les variables d'induction de chaque boucle et les éliminer quand c'est possible.
7. Produire du code machine cible à partir du résultat des questions 1 et 6 et comparer les codes obtenus.