Raphaël Charrondière
raphael.charrondiere@inria.fr
Semaine 1 - échéance S2

Feuille de TD 0: Le début

Où l'on se prépare

Introduction : Ce premier TD se déroule avant le premier cours, il s'agit donc de préparer le premier cours. Tout d'abord un exercice de logique, parce que c'est la base pour bien raisonner, ensuite deux exercices pour aborder les problématiques du cours.

Règles et fonctionnement des TD : Il vous est demandé de travailler en groupe de trois ou quatre personnes maximum. Chaque groupe devra rendre une correction à son responsable de TD. Elle devra inclure la solution de **tous** les exercices (même ceux non corrigés en TD), avec les justifications nécessaires. Vous devez envoyer votre travail scanné ou sous format PDF par e-mail **avant la séance** indiquée sur la fiche de TD, ou le remettre en main propre **avant le début de TD**. Chaque rendu sera noté, un non rendu recevra automatiquement un zéro. Si le correcteur se rend compte de plagiat entre les groupes, il mettra 0 à l'ensemble du rendu.

Exercice 1 / Un peu de logique

Dans cet exercice, on part d'un ensemble d'opérateurs logiques et relationnels restreint pour retrouver l'ensemble de nos opérateurs usuels. De manière générale il est plus facile, efficace de prouver quelque chose s'il y a moins de cas. Ici, on montre qu'il suffit de très peu d'opérateurs.

1) Soit P et Q deux propositions logiques. À l'aide des relations $\neg$ (la négation) et $\wedge$ (l'opérateur «et»), traduire les propositions suivantes :

— $P \vee Q$ («ou»)
— $P \veebar Q$ («ou exclusif», la proposition est vraie ssi P est vrai quand Q est faux et inversement)
— $P \implies Q$ (P implique Q)
— $\mathbb{0}$ («faux»)

2) (*Bonus*) On est parti de deux opérateurs logiques, ne pourrait-on pas utiliser un seul opérateur ? Indice : la réponse est oui, à vous de trouver l'opérateur en question.

3) Pour $a, b \in \mathbb{N}$, en utilisant la relation binaire $<$, et les opérateurs logiques usuels, traduire les relations suivantes :

— $a > b$
— $a = b$
— $a \geqslant b$
— $a \leqslant b$
— $a \neq b$

4) Soit une proposition logique $A(i)$. En utilisant seulement le quantificateur existentiel $\exists$, et les opérateurs booléens classiques, traduire $\forall i\ A(i)$.

5) Soit une proposition logique $A(i)$. En utilisant les quantificateurs $\exists$, $\forall$, et les opérateurs booléens classiques, traduire $\exists! i\ A(i)$ («Il existe un unique i tel que $A(i)$»).

Exercice 2 / Ça termine ?

?) Ci-dessous plusieurs programmes sont présentés. Pour chacun d'eux, justifiez s'il termine ou non. On supposera que la fonction `rand(i,j)` renvoie un entier entre i et j, et `rand()` renvoie un entier positif. On supposera aussi qu'un entier peut être arbitrairement grand, et un double arbitrairement précis.

P1 :
```
int N=1000;
for(int i=0;i<N;i++)
    a=a+i;
```

P2 :
```
int N=1000;
for(int i=N;i>0;i--)
    a=a*i;
```

P3 :
```
int i=10000;
while(int i>0){
    a=rand(1,1000);
    i-=a;
}
```

Raphaël Charrondière
raphael.charrondiere@inria.fr

P4 :
```
double d=1.0;
while(d>0){
   d=d/2;
   a+=d;
}
```

P5 :
```
int i=1000,j=1000;
while (i>0||j>0){
   if(j==0){
      i--;j=1000;
   }
   j--;
}
```

P6 :
```
int i=rand(),j=rand();
while (i>0||j>0){
   if(j==0){
      i--;j=rand();
   }
   j--;
}
```

Exercice 3 Je ne suis pas comme les autres

On propose les programmes IMP ci-dessous. Même si vous ne connaissez pas ce langage vous pouvez le comprendre (**skip** est l'instruction qui ne fait rien) :

P1 : **while** $x > 0$ **do** $x := x + 1$

P2 : **while** $x \neq 0$ **do** $x := x \times y + 1$

P3 : **while** $\mathbb{1}$ **do skip**

P4 : **skip**

P5 : $y := 0$

P6 : **while** $x \neq x$ **do** $x := x + 1$

P7 : **if** x=0 **then** $(x := x + 1$**;** $y := 1)$ **else** $y := 0$**;**
$x := x - y$**;** $y := 0$

P8 : $y := x + 1$**; while** $x \neq y$ **do** $y := x + 1$

P9 : **while** $x \neq y$ **do** $x := y + 1$

? Grouper les programmes qui ont un comportement équivalent. Important : les variables sont considérées comme des entiers naturels.
