TD Vérification statique et vérification dynamique — FEUILLE 3

Exercice 1.

▷ **Question 1** *On considère dans un premier temps les commandes* $x := e$, C_1 ; C_2 *et* IF e THEN C_1 ELSE C_2 END, *vues en cours. Montrer que les égalités suivantes sont vraies pour ces commandes :*

1. $wp(C, A \wedge B) = wp(C, A) \wedge wp(C, B)$
2. $wp(C, A \vee B) = wp(C, A) \vee wp(C, B)$

▷ **Question 2** *On ajoute la commande* WHILE e DO C INVARIANT J VARIANT V END. *Que peut-on dire des égalités ci-dessus ?*

Exercice 2. Soit le programme suivant :

```
x,v,y,z,t : INT ;
tab:array(5..10) of INT;
begin
1.   read(x) ;
2.   v := 2*x ;
3.   if x>0
4.     then y:=x+1
5.     else z:=2*x
6.   end ;
7.   if v>0
8.     then t:=y
9.     else t:=z
10.  end ;
11.  tab(t):=3;
end
```

▷ **Question 1** *Utiliser le calcul de plus faible précondition pour caractériser les valeurs possibles pour* x *garantissant l'absence de buffer overflow.*

▷ **Question 2** *Même question pour les débordements arithmétiques.*

Exercice 3. Soit le programme suivant :

```
begin
   z:=0;
   while x>0 do
    z:=z+y ;
    x:=x-1
   end
end
```

▷ **Question 1** *Que calcule ce programme (contenu de la variable* `z`*) pour* x *initialement positif ou nul ? On prouvera la réponse on donnant un invariant. On notera* $x0$ *et* $y0$ *les valeurs initiales respectives de* x *et* y.

▷ **Question 2** *Prouver l'arrêt de ce programme.*

Exercice 4. Soit le programme suivant :

```
begin
   u:=0;
   while x>1 do
    if pair(x) then x:=x/2 ; y:=y*2
               else x:=x-1;u:=u+y
    end
   end  ;
   y:=y+u
end
```

Prouver que ce programme est tel que $y = x0 * y0$ si $x0 > 0$, $x0$ et $y0$ désignant les valeurs initiales de `x` et `y`.

Exercice 5. On s'intéresse à la définition du calcul de plus faible précondition pour une instruction for de la forme :

```
for i :=e1 to e2 do C invariant I end
```

avec C ne modifiant pas i. **Dans un premier temps on supposera que l'expression** $e2$ **n'est pas modifiée par** C**.**

▷ **Question 1** *Proposez une définition de wp(*`for i :=e1 to e2 do C invariant I end`*, R).*

▷ **Question 2** *Proposer une définition de l'instruction for à l'aide d'un while qui produit la même plus faible précondition. On prouvera l'arrêt de cette traduction.*

▷ **Question 3** *Reprendre les questions précédentes en levant l'hypothèse* **l'expression** $e2$ **n'est pas modifiée par** C**.**