NOLLE

Damien

L3 - informatique

SPP - Partiel 1 :

Note :

/20

Exercice 1)

```
// @predicate p();
// @predicate q();
// @predicate r();
// @lemma l1: !((p || q) && (!q || r) &&
  (p || !r) && (!p || q) && (!p || !q));
```

Exercice 2)

```
// @predicate S(int i);
// @predicate R(int i, int j);

// @lemma l1: (exists x. (forall y. S(x) <->
  S(y))) && (forall z. S(z));

// @lemma l2: forall x. exists y, z. (R(x,x)
  -> R(z,y)) && (! R(x,x) || R(y,z));

// @lemma l3: forall x, y. exists z. (R(x,x) &&
  R(x,y)) -> R(x,z);
```

VOLLE
Pommier
L3 - informatique

Exercice 3 /

Q1 :

Constante : $|t| = 100$ → longueur du tableau.

Données :

t : tableau de n entiers classés en ordre croissant.

x : entier dont on recherche la position d'insertion dans la séquence $t[0..n-1]$.

Résultat : $posx$: entier compris entre 0 et $n-1$, c'est la position d'insertion de x donnée.

Précondition : $n \geq 0 \wedge n < |t| \wedge \forall i, j . ((i \in [0..n-1] \wedge j \in [0..n-1] \wedge i < j) \Rightarrow t[i] \leq t[j])$

Postcondition : $\forall k . (k \in [0..posx-1] \Rightarrow t[k] \leq x) \wedge \forall k . (k \in [posx..n-1] \Rightarrow t[k] > x)$

Q2 :

$min := 0;$ $max := n-1;$

/* $I = \forall j . (j \in [0..min] \Rightarrow t[j] \leq x) \wedge \forall j . (j \in [max+1..n-1] \Rightarrow t[j] > x)$ */

```
while min ≠ max do
    /* min ≠ max ∧ I */
    m := (min + max + 1) div 2;

    if t[m] ≤ x then
        min := m
    else
```

NOLLE
Parmien
L3 - informatique

max := m - 1

fi

/* I */

/* min = max ∧ I, alors posx = min + 1 ∨
posx = max + 1 (indice du premier élément de
t > x */

posx := min + 1

Q3 :

```
//@ predicate I (int min, int max, int t[], int
n, int x) = forall j. (0 <= j <= min -> t[j] <= x)
 && forall j. (max + 1 <= j < n -> t[j] > x);
```

```
//@ predicate postcond1 (int posx, int min, int
max, int t[], int n, int x) = (min == max
&& I(min, max, t, n, x)) -> (posx == min + 1
|| posx == max + 1);
```

Q4 :

```
//@ predicate precond (int n, int [ ]) = (n >= 0
&& n < length(t)) && forall i, j. ((0 <= i < n
&& 0 <= j < n && i < j) -> t[i] <= t[j]);
```

Q5 :

```
//@ predicate postcond2 (int posx, int x, int n,
int t[]) = forall k. (0 <= k < posx -> t[k]
<= x) && forall k. (posx <= k < n -> t[k] > x);
```

Exercice 4)

Q1 :

NOLLE
Pommier
L3-informatique

$\{x \geq 0\}$ $i := 0;$ $p := 0;$ while $i < x$ do
$i := i + 1;$ $p := p + y$ od $\{p = x \times y\}$

Q2.

N°	Triplet ou formule	justification
1.	$(i < x \land p = y \times i \land i \leq x) \Rightarrow (p = y \times i \land i \leq x)$	OK (1)
2.	$\{p = y \times i \land i \leq x\}$ $i := i + 1$ $\{p = y \times (i - 1) \land i \leq x\}$	(assignment)
3.	$(p = y \times (i - 1) \land i \leq x) \Rightarrow (p = y \times (i - 1) \land i \leq x)$	OK (2)
4.	$\{i < x \land p = y \times i \land i \leq x\}$ $i := i + 1$ $\{p = y \times (i - 1) \land i \leq x\}$	(conséquence) 1, 2, 3
5.	$\{p = y \times (i - 1) \land i \leq x\}$ $p := p + y$ $\{p = y \times i \land i \leq x\}$	(assignment)
6.	$\{i < x \land p = y \times i \land i \leq x\}$ $i := i + 1;$ $p := p + y$ $\{p = y \times i \land i \leq x\}$	(séquence) 4, 5
7.	$(x \geq 0 \land i = 0 \land p = 0) \Rightarrow (p = y \times i \land i \leq x)$	OK (3)
8.	$\{p = y \times i \land i \leq x\}$ while $i < x$ do $i := i + 1;$ $p = p + y$ od $\{\neg(i < x) \land (p = y \times i \land i \leq x)\}$	(while) 6
9.	$(\neg(i < x) \land I) \Rightarrow p = x \times y$	OK (4)
10.	$\{x \geq 0 \land i = 0 \land p = 0\}$ while $i < x$ do $i := i + 1;$ $p := p + y$ od $\{p = x \times y\}$	(conséquence) 7, 8, 9
11.	$\{x \geq 0 \land i = 0\}$ $p := 0$ $\{x \geq 0 \land i = 0 \land p = 0\}$	(assignment)

NOLLE
Pommier
L3 - informatique

12. $\{x \geq 0 \wedge i = 0\}$ $p := 0$; while $i < x$ do $i := i + 1$; $p := p + y$ od $\{p = x \times y\}$ (séquence) 11, 10

13. $\{x \geq 0\}$ $i := \{x \geq 0 \wedge i = 0\}$ (assignment)

14. $\{x \geq 0\}$ $i := 0$; $p := 0$; while $i < x$ do $i := i + 1$; $p := p + y$ od $\{p = x \times y\}$ (séquence) 13, 12

OK 1 : pour que le membre gauche soit vrai, il faut que $i < x$. car si $i = x$, $x < i$ sera faux alors que $x \geq i$ sera vrai, donc toute le membre gauche sera fausse. Si $p = y \times i$ est vrai et que $x = i$, alors le second membre sera vrai. Si $x > i$ ou $p \neq y \times i$, les deux membres seront faux. Si $i < x$, alors les deux membres sont vrai. Dans tous les cas, la formule est vrai.

OK 2 : puisque les deux membres sont identiques, soit les deux sont vrais, soit les deux sont faux. Dans tous les cas, la formule est vraie.

OK 3 et 4 :
au début de la boucle :
$x \geq 0$
$i = 0$
$p = 0$

Donc, par Q3, le membre gauche est vrai et

NOLLE
Pormien
L 3 - informatique

$r = y \times i$
$0 = y \times 0$ / ce qui est vrai

$i \leq x$, en sachant que $i = 0$ et x peut être supérieur ou égal à 0, donc I est vrai.

Pendant la boucle :

1ère itération :
$r = 0 + y = y$
$i = 1$
$i < x$

2ème itération
$r = y + y = 2y$
$i = 2$

3ème itération.
$r = 2y + y = 3y$
$i = 3$

etc...

Nous constatons à chaque fois que, $r = y \times i$ et que si nous sommes dans la boucle, alors $i < x$, donc I est vrai.

A la fin de la boucle :
On sait que $i = x$, donc $r = y \times i = y \times x$. I est donc vrai.