Université 20 Août 1955. Skikda	Faculté des Sciences	Département d'Informatique
Méthodes Formelles	Examen de rattrapage	Durée 1h00

Exercice 1 (12 points) : Un distributeur automatique de boissons offre les opérations suivantes :

Opérations	Paramètres	Significations
Init	-> Distrib	Création
Pay	Distrib -> Distrib	Exprime le paiement d'une boisson quelconque
Select	Distrib Saveur-> Distrib	Exprime la sélection d'une saveur. Saveur peut être un entier. Par exemple 1 : orange, 2 : citron, etc.
Deliver	Distrib Cup -> Distrib	Exprime la distribution d'une boisson. Cup peut être un entier. Par exemple 1 : petit, 2 : moyen, 3 : grand.
NbPieces	Distrib -> Integer	Renvoie le nombre de pièces de monnaie insérée dans le distributeur
Current	Distrib -> Saveur	Renvoie la saveur de la sélection courante
Full	Distrib Cup > Boolean	Renvoie vrai si le verre en cours de remplissage est plein
CupContent	Distrib Cup -> Saveur	Renvoie la saveur du contenu du verre en cours de distribution

La spécification **incomplète** d'une telle spécification (le type abstrait de donnée) est donnée dans ce qui suit :

```
type DISTRIBUTEUR is imports BOOLEAN, INTEGER
      sort Distrib
      opns
            Init : → Distrib;
            Pay : Distrib → Distrib;
            Select : Distrib, Saveur → Distrib;
            Deliver : Distrib, Cup → Distrib;
            NbPieces : Distrib → Integer;
            Current : Distrib → Saveur;
            Full : Distrib, Cup → Boolean;
            CupContent : Distrib, Cup → Saveur;
      eqns
      forall dist: Distrib, v,vv: Cup, s:Saveur;
      1. NbPieces(Init) = 0;
      2. NbPieces(Pay(dist)) = NbPieces (dist) + 1;
      3. NbPieces(Select(dist,s)) = NbPieces (dist);
         NbPieces (Deliver(dist,v)) =
            4. if NbPieces (dist) >= 20 and not Full(dist,v) then NbPieces (dist) - 20
            5. else NbPieces (dist)
               fi;
      6. Full(Init,v) = False;
      7. Full (Pay(dist),v) = Full (dist,v);
      8. Full (Select(dist,s),v) = Full (dist,v);
         Full (Deliver(dist,v),vv) =
            9. if v=vv and NbPieces (dist) >= 20 and  not Full (dist,v) then True
            10.else Full(dist,vv)
               fi
endtype
```

1. Complétez les espaces pointillés de cette spécification (utilisez les chiffres)

Exercice 2 (8 points) : On considère le programme suivant :

```
c := 0 ;
s := 1 ;
while (s < n) do
      c := c + 1 ;
      s := s + 2 * c + 1
end
```

Le résultat recherché est la valeur de la variable c en fin d'exécution.

1. Pouvez-vous deviner ce qu'il calcule ?

   Le programme retourne la partie entière de la racine carrée de n.

2. On spécifie le programme de la façon suivante :

- La post condition est $c >= 0 \wedge c * c <= n < (c + 1) * (c + 1)$
- L'invariant de la boucle est $c >= 0 \wedge n >= c * c \wedge s = (c + 1) * (c + 1)$

Prouver ce programme utilisant la logique de Hoare.

{pré} c := 0 ; s := 1 ; while (s <= n) do c := c + 1 ; s := s + 2 * c + 1 {post}

{pré} c := 0 ; s := 1 ; while (s <= n) do c := c + 1 ; s := s + 2 * c + 1 {post}

{pré} c := 0 ; s := 1 ; while (s <= n) do c := c + 1 ; s := s + 2 * c + 1 $\{c >= 0 \wedge c^2 <= n < (c + 1)^2\}$

{pré} c := 0 ; s := 1 {Inv} et {Inv} while (s <= n) do c := c + 1 ; s := s + 2 * c + 1 $\{Inv \wedge$ not cond$\}$

{pré} c := 0 ; s := 1 $\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$ et $\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$ La Boucle $\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2 \wedge s>n\}$

{pré} c := 0 ; s := 1 $\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$

s := 1

$\{c >= 0 \wedge c^2 <= n \wedge 1 = (c + 1)^2\}$

c := 0

$\{0 >= 0 \wedge 0^2 <= n \wedge 1 = (0 + 1)^2\}$

n>=0

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$ La Boucle $\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2 \wedge s>n\}$

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2 \wedge s <= n\}$ c := c + 1 ; s := s + 2 * c + 1 $\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2 \wedge s <= n\}$ {c:=c+1 } $\{c >= 0 \wedge c^2 <= n \wedge s = c^2\}$

Et

$\{c >= 0 \wedge c^2 <= n \wedge s = c^2\}$ {s := s + 2 * c + 1 } $\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2 \wedge s <= n\}$ => Q et Q {c:=c+1 } $\{c >= 0 \wedge c^2 <= n \wedge s = c^2\}$

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2 \wedge s <= n\}$ => $\{c+1 >= 0 \wedge (c+1)^2 <= n \wedge s= (c+1)^2\}$ et $\{c+1 >= 0 \wedge (c+1)^2 <= n \wedge s= (c+1)^2\}$ {c:=c+1 } $\{c >= 0 \wedge c^2 <= n \wedge s = c^2\}$

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2\}$

s := s + 2 * c + 1

$\{c >= 0 \wedge c^2 <= n \wedge s + 2 * c + 1 = (c + 1)^2\}$

$\{c >= 0 \wedge c^2 <= n \wedge s + 2 * c + 1 = c^2 +2*c+ 1\}$

$\{c >= 0 \wedge c^2 <= n \wedge s= c^2\}$

c := c + 1

$\{c+1 >= 0 \wedge (c+1)^2 <= n \wedge s= (c+1)^2\}$

$\{c >= 0 \wedge c^2 <= n \wedge s = (c + 1)^2 \wedge s <= n\}$=>$\{c+1 >= 0 \wedge (c+1)^2 <= n \wedge s= (c+1)^2\}$

On remplace s par $(c+1)^2$ On obtient $\{\mathbf{c >= 0 \wedge c^2 <= n} \wedge (c + 1)^2 <= n\}$=>$\{\mathbf{c+1 >= 0} \wedge (c+1)^2 <= n\}$

Bon Courage !