

TDC: AlgDem.

Q 1.1 Post-condition : $a = 0 \wedge b \neq 0$.

Code:
Soit $b \neq 0 \wedge b \neq 0 \wedge a+1+b = 0$
Soit $a+1 \neq 0 \wedge a+1+b \neq 0$
 $a=a+1$
if $a+b==0$:
 $a=b$ # $b=0 \wedge b \neq 0$
else:
 $b=a$ # $a=0 \wedge a \neq 0$
 # $a=0 \wedge b \neq 0$

Q 1.2 Post-condition : $a \% (b+c) = 0$.

Code :
$b+c \neq 0 \wedge b+c \neq 0 \wedge ((c+1)/c) \wedge (b+c) \wedge ((c+1)+a) = a \% (b+c)$
$b+c=0 \wedge c \neq 0 \wedge (a+1/c) \wedge (b*b + ((a+1)/c) * b * b) = 0$
 $a=b*c$
if $a!=0$:
 $b=c*b/a$ # $a \% ((c+1)*b/a) + (a * ((c+1)*b/a)) = 0$
 $c=c+1$ # $a \% (b + (a*b)) = 0$
else:
 # $(a+1/c) \% (b*b + ((a+1)/c) * b * b) = 0 \wedge c \neq 0$
 $a=(a+1)/c$
 $b=b*b$ # $a \% (b*b + (a*b*b)) = 0$
 $c=a*b$ # $a \% (b + (a*b)) = 0$
 # $a \% (b+c) = 0$

Exercice 2 : Division

Le code qui suit implémente la division euclidienne de sorte que en fin de calcul, la variable *res* contient une paire dont :

- le premier élément est le quotient de a par b et
- le second élément est reste de la division de a par b .

```
q=0
r=a
while r >= b:
    q=q+1
    r=r-b
res=(q,r)
```

$r-b=p \Rightarrow r=b+p$. Comme $b>0$, on a $b+p>p$.
$r-b=p$ Comme $r \geq b > 0$, $r > 0$.
$r=p$

Pré-condition $a \geq 0$ et $b > 0$

Post-condition en posant $(q', r') = res$, nous avons $a = q' \times b + r'$ et $0 \leq r' < b$.

Variant de la boucle r

Invariant de la boucle $a = q \times b + r$, $r \geq 0$ et $b > 0$.

Q 2.1 Étude du variant Calculez la plus faible pré-condition de $r = p$ à travers le corps de la boucle. Déduisez-en que le variant décroît strictement. Justifiez également que le variant est positif.

Q 2.2 Étude de l'invariant Calculez la plus faible pré-condition de l'invariant à travers le corps de la boucle. Montrez que cette plus faible condition est toujours vraie en utilisant les hypothèses auxquelles vous avez droit.

$$\# a = q * b + r, r \geq 0 \text{ et } b > 0$$

$$r = r - b$$

$$\# a = q * b + r - b, r - b \geq 0 \text{ et } b > 0$$

$$q = q + 1$$

$$\# a = (q + 1) * b + r - b, r - b \geq 0 \text{ et } b > 0.$$

On sait que $r \geq b$ dans le corps de la boucle, donc ok

On a par précondition que $b > 0$

Et on a : $a = (q + 1) * b + r - b = a = qb + r$. donc ok.

Exercice 3 : Nombre de diviseurs

Le nombre 42 est divisible par 1, 2, 3, 6, 7, 14, 21, 42, il a 8 diviseurs. Le code qui suit permet de compter le nombre de diviseurs d'un nombre.

```
i = 2
res = 2
while i*i < n:
    if n%i == 0:
        res = res + 2
        i = i + 1
if i*i == n:
    res = res + 1
```

L'algorithme implémenté utilise des propriétés particulières des diviseurs :

- Tout d'abord, lorsque l'on connaît un diviseur a d'un nombre n , alors, c'est qu'il existe un nombre b tel que $a \times b = n$ et ainsi, dès lors que $a \neq b$, on connaît en fait deux diviseurs de n . Nous dirons que a et b forment une *paire de diviseurs* de n .
- Enfin, si a et b sont une paire de diviseurs de n (i.e $a \times b = n$), alors l'un d'entre eux est un *petit* diviseur, en ce sens que son carré est inférieur ou égal à n . Plus formellement, nous avons que soit $a \times a \leq n$, soit que $b \times b \leq n$.

Ces deux propriétés sont mises à profit dans l'algorithme par le fait que pour compter les diviseurs de n , nous ne parcourons que les *petits* diviseurs, ceux dont le carré est inférieur à n . Nous comptons les *grands* diviseurs parce qu'ils sont toujours associés à un petit diviseur.

Voici les spécifications du code ci-dessus :

Pré-condition $n > 1$

Post-condition res est le nombre de diviseurs de n

Variant de la boucle $n - i$

Invariant de la boucle res est le nombre de diviseurs de n qui sont compris entre 1 et $i-1$ et entre $n//i + 1$ et n .

Pour simplifier les notations, nous noterons

$$\text{div}(n, i) = |\{k \mid k|n \wedge (1 \leq k < i \vee n//i < k \leq n)\}|$$

ainsi, l'invariant s'écrit :

$$res = \text{div}(n, i)$$

Q 3.1 Étude du variant Calculez la plus faible pré-condition de $i = p$ à travers le corps de la boucle. Déduisez-en que le variant décroît strictement. Justifiez également que le variant est positif.

Q 3.2 Étude de l'invariant Calculez la plus faible pré-condition de l'invariant à travers le corps de la boucle. Montrez que cette plus faible condition est toujours vraie en utilisant les hypothèses auxquelles vous avez droit.

Q 3.3 Étude de la pré-condition Calculez la plus faible pré-condition de l'invariant à travers le code qui précède la boucle. Montrez que cette plus faible pré-condition est toujours vraie lorsque la pré-condition du code est vérifiée.

Q 3.4 Étude de la post-condition Calculez la plus faible pré-condition de la post-condition du code à travers le code qui suit la boucle. Montrez que cette plus faible pré-condition est toujours vraie en utilisant les hypothèses auxquelles vous avez droit.

$$\# i = p \wedge n = a$$

$$i = i + 1$$

$$\# i + 1 = p \wedge n = a \Rightarrow i = p - 1 \wedge n = a$$

Le variant est $a - p$ en fin de boucle ssi

c'est $a - p + 1$ en début de boucle.

$$\text{On a que } a - p + 1 > a - p$$