

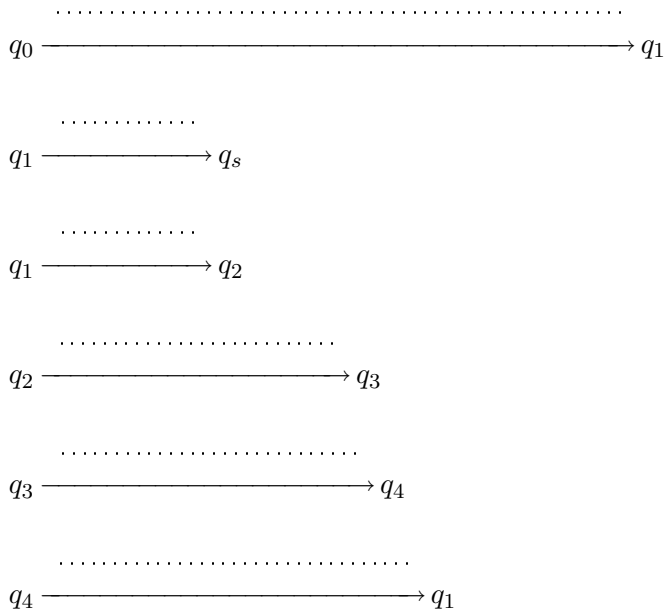
Exercice 1 – Somme des carrés des N premiers entiers pairs

PROGRAMME

```

1  x:=0 ; k:= 0 ; i:=0 ;
2  while(i < N){
3      i:= i + 1 ;
4      k:= k + 2 ;
5      x:= x + k * k ;
6  }
```

Q1. (1 pt) Dessinez l'automate correspondant au programme précédent.



Q2. (1 pt) Faites quelques exécutions pour trouver l'égalité qui relie les variables k et i .

Q3. (8 pt) Montrez par la technique de Floyd-Dijkstra-Hoare que l'état de sortie du programme précédent vérifie la propriété $x = \frac{2N(N+1)(2N+1)}{3}$

SOLUTION

$$\begin{aligned}
 \psi_s &\stackrel{def}{=} x = \frac{2N(N+1)(2N+1)}{3} \\
 \psi_1 &\stackrel{def}{=} i \leq N \wedge x = \frac{2i(i+1)(2i+1)}{3} \wedge k = 2i \\
 \psi_4 &\stackrel{def}{=} \psi_1[x \leftarrow x + k^2] \equiv i \leq N \wedge x + k^2 = \frac{2i(i+1)(2i+1)}{3} \wedge k = 2i \\
 \psi_3 &\stackrel{def}{=} \psi_4[k \leftarrow k + 2] \equiv i \leq N \wedge x + (k + 2)^2 = \frac{2i(i+1)(2i+1)}{3} \wedge k + 2 = 2i \\
 \psi_2 &\stackrel{def}{=} \psi_3[i \leftarrow i + 1] \equiv i + 1 \leq N \wedge x + (k + 2)^2 = \frac{2(i+1)((i+1)+1)(2(i+1)+1)}{3} \wedge k + 2 = 2(i + 1) \\
 \psi_0 &\stackrel{def}{=} \psi_1[x \leftarrow 0; k \leftarrow 0; i \leftarrow 0] \equiv 0 \leq N \wedge 0 = \frac{2(0)((0)+1)(2(0)+1)}{3} \wedge 2 = 2 \times 0 \equiv 0 \leq N
 \end{aligned}$$

Vérification des invariants :

$$\begin{array}{c}
\overbrace{i \leq N \wedge x = \frac{2i(i+1)(2i+1)}{3} \wedge k = 2i \wedge i < N}^{\psi_1} \\
\underbrace{\hspace{10em}}_A \quad \underbrace{\hspace{2em}}_B \quad \underbrace{\hspace{2em}}_{test} \\
\Rightarrow \\
\overbrace{i+1 \leq N \wedge x + (k+2)^2 = \frac{2(i+1)((i+1)+1)(2(i+1)+1)}{3} \wedge k+2 = 2(i+1)}^{\psi_2} \\
\underbrace{\hspace{2em}}_{test} \quad \underbrace{\hspace{10em}}_{d'apres A \wedge B} \quad \underbrace{\hspace{2em}}_B
\end{array}$$

Q4. (1 pt) En déduire les conditions initiales du programme qui garantissent la propriété de correction à l'état de sortie.

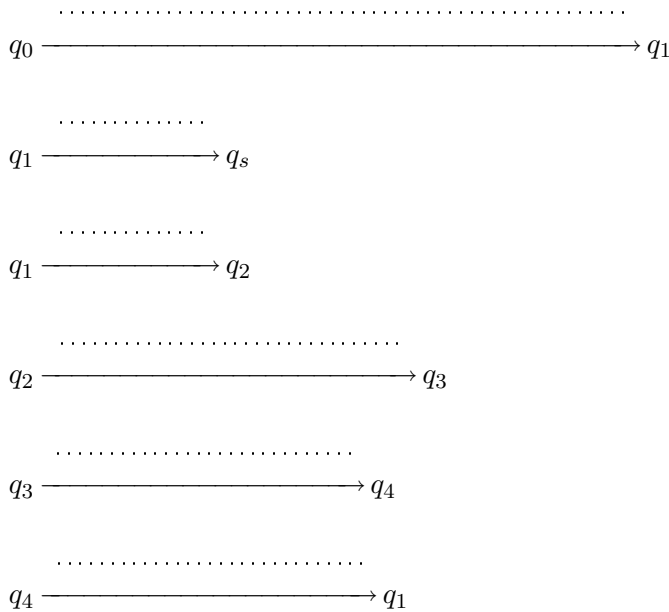
Exercice 2 – Somme des carrés des K premiers entiers impairs

PROGRAMME

```

1  r:=0 ; u:= 1 ; j:=0 ;
2  while(j < K){
3      r:= r + u * u ;
4      j:= j + 1 ;
5      u:= u + 2 ;
6  }
```

Q5. (1 pt) Dessinez l'automate correspondant au programme précédent.



Q6. (1 pt) Faites quelques exécutions pour trouver l'égalité qui relie les variables u et j .

Q7. (8 pt) Montrez par la technique de Floyd-Dijkstra-Hoare que l'état de sortie du programme précédent vérifie la propriété $r = \frac{K(2K-1)(2K+1)}{3}$

SOLUTION

$$\begin{aligned}
 \psi_s &\stackrel{\text{def}}{=} r = \frac{K(2K-1)(2K+1)}{3} \\
 \psi_1 &\stackrel{\text{def}}{=} j \leq K \wedge r = \frac{j(2j-1)(2j+1)}{3} \wedge u = 2j + 1 \\
 \psi_4 &\stackrel{\text{def}}{=} \psi_1[u \leftarrow u + 2] \equiv j \leq K \wedge r = \frac{j(2j-1)(2j+1)}{3} \wedge u + 2 = 2j + 1 \\
 \psi_3 &\stackrel{\text{def}}{=} \psi_4[j \leftarrow j + 1] \equiv j + 1 \leq K \wedge r = \frac{j+1(2j+1-1)(2j+1+1)}{3} \wedge u + 2 = 2(j + 1) + 1 \\
 \psi_2 &\stackrel{\text{def}}{=} \psi_3[r \leftarrow r + u^2] \equiv j + 1 \leq K \wedge r + u^2 = \frac{j+1(2j+1-1)(2j+1+1)}{3} \wedge u = 2j + 1 \\
 \psi_0 &\stackrel{\text{def}}{=} \psi_1[r \leftarrow 0; u \leftarrow 1; j \leftarrow 0] \equiv 0 \leq K \wedge 0 = \frac{(0)(2(0)-1)(2(0)+1)}{3} \wedge 1 = 2 \times 0 + 1 \equiv 0 \leq K
 \end{aligned}$$

Vérification des invariants :

$$\begin{array}{c}
\overbrace{j \leq K \wedge r = \frac{j(2j-1)(2j+1)}{3} \wedge u = 2j+1 \wedge j < K}^{\psi_1} \\
\underbrace{\hspace{10em}}_A \quad \underbrace{\hspace{5em}}_B \quad \underbrace{\hspace{2em}}_{test} \\
\Rightarrow \\
\overbrace{j+1 \leq K \wedge r+u^2 = \frac{j+1(2j+1-1)(2j+1+1)}{3} \wedge u = 2j+1}^{\psi_2} \\
\underbrace{j+1 \leq K}_{test} \quad \underbrace{r+u^2 = \frac{j+1(2j+1-1)(2j+1+1)}{3}}_{d'apres A \wedge B} \quad \underbrace{u = 2j+1}_B
\end{array}$$

Q8. (1 pt) En déduire les conditions initiales du programme qui garantissent la propriété de correction à l'état de sortie.