

Question 1

```
c0 = 1;  
void lock() {  
    s.P();  
}  
void unlock() {  
    s.V();  
}
```

Question 2

Il suffit de mettre le compteur $c0$ à 10.

Question 3

L'idée est de prévoir un sémaphore pour chaque groupe, leur compteur valant le nombre de threads dans ce groupe en attente.

```
sem_A;  
sem_B;  
c0_A = 0;  
c0_B = 0;  
void RDV_A() {  
    sem_A.V();  
    sem_B.P();  
}  
void RDV_B() {  
    sem_B.V();  
    sem_A.P();  
}
```

Question 4

```
sem_AforB;  
sem_AforC;  
sem_BforA;  
sem_BforC;  
sem_CforA;  
sem_CforB;  
// Les compteurs sont tous initialisés à
```

0

```
void RDV_A() {  
    AforB.V();  
    AforC.V();  
    BforA.P();  
    CforA.P();  
}  
void RDV_B() {  
    BforA.V();  
    BforC.V();  
    AforB.P();  
    CforB.P();  
}  
void RDV_C() {  
    CforA.V();  
    CforB.V();  
    AforC.P();  
    BforC.P();  
}
```

Question 5

```
sem_mutex;  
sem;  
c0_sem_mutex = 1;  
c0_sem = 0;  
int c = 0;  
void thread() {  
    sem_mutex.P();  
    c++;  
    if (c == N)
```

```

        sem.V(); //Réveil en cascade
sem_mutex.V();
sem.P();
sem.V();
}

```

Question 6

```

Messages tampon[N];
int i_prod = 0;
int i_cons = 0;
sem_prod;
sem_cons;
sem_protecteur;
c0_prod = N;
c0_cons = 0;
c0_protecteur = 1;
void producteur(Message m) {
    sem_prod.P();
    sem_protecteur.P();
    tampon[i_prod] = m;
    i_prod = (i_prod + 1)%N;
    sem_producteur.V();
    sem_cons.V();
}
void consommateur(Message *m) {
    sem_cons.P();
    sem_protecteur.P();
    *m = tampon[i_cons];
    i_cons = (i_cons + 1)%N;
    sem_protecteur.V();
}

```

```
    sem_prod.V();  
}
```

Question 7

```
sem;  
sem_prot;  
c0_sem = 1;  
c0_prot = 1;  
int nb_lecteurs = 0;  
void debut_lire() {  
    sem_prot.P();  
    if (nb_lecteurs == 0)  
        sem.P();  
    nb_lecteur++;  
    sem_prot.V();  
}  
void fin_lire() {  
    sem_prot.P();  
    nb_lecteur--;  
    if (nb_lecteurs == 0)  
        sem.V();  
    sem_prot.V();  
}  
void debut_redace() {  
    sem.P();  
}  
void fin_redace(){  
    sem.V();  
}
```

Question 8

L'idée est de rajouter un sas