Licence Info
Algorithmique Avancée
TD 8 : novembre 2006
Analyse du temps de calcul

Déterminer (en fonction de n à $O(\ )$ près) le temps de calcul de chaque boucle :
(`P()` est une procédure dont le temps de calcul est constant ; et `deux(i)` calcule la `i`-ème puissance de 2 en temps constant.)

1. ```
   pour i allant de 1 a n par pas de 1 faire P() finpour;
   ```

2. ```
   pour i allant de 1 a n par pas de 1 faire
      pour j allant de 1 a i par pas de 1 faire P() finpour
   finpour;
   ```

3. ```
   i:=1;
   tant que i<n faire i:=i*2
   fintantque;
   ```

4. ```
   pour i allant de 1 a n par pas de 1 faire
      j:=1;
      tant que j<n faire j:=j*2
      fintantque
   finpour;
   ```

5. ```
   i:=1;
   tant que i<n faire
      i:=i*2;
      pour j allant de 1 a i par pas de 1 faire P() finpour
   fintantque;
   ```

6. ```
   i:=1;
   pour j allant de 1 a n par pas de 1 faire i:=i*2 finpour;
   pour j allant de 1 a i par pas de 1 faire P() finpour;
   ```

7. ```
   i:=1;
   pour k allant de 1 a n par pas de 1 faire
      i:=i*2;
      pour l allant de 1 a i par pas de 1 faire
         pour m allant de 1 a k faire P() finpour
      finpour
   finpour;
   ```
   ```

8.
```
pour k allant de 1 a n par pas de 1 faire
   i:=1;
   pour j allant de 1 a k par pas de 1 faire i:=i*2 finpour;
   pour j allant de 1 a i par pas de 1 faire P() finpour
finpour;
```

9.
```
i:=2;
tant que i<n faire i:=i*i
fintantque;
```

10.
```
i:=1;
tant que i<n faire i:=deux(i)
fintantque;
```

11.
```
i:=0;j:=0;
tant que j<n faire
   i:=i+1;
   j:=j+i;
fintantque;
```

12.
```
pour i de 1 a n faire T[i]:=0 finpour;
j:=0;
tant que j<n faire
   j:=1;
   tantque T[j]=9 et j<n faire
      T[j]=0;
      j:=j+1;
   fintantque;
   T[j]:=T[j]+1
fintantque;
```