CC1 Exercice 1 2019 (Corrigé)

pakpake

1 Mars 2020

Exercice 1

1)

a)

```python
def fact_rec(n):
    if n == 0:
        return 1
    else:
        return n*fact_rec(n-1)
```

b)

Formule récursive de la complexité $T(n)$ de cet algorithme :

si

$$n > 0 \quad T(n) = 5 + T(n-1)$$

car on a +1 test d'égalité (if), +1 return (celui du else), +1 multiplication, +1 appel de la fonction, +1 soustraction

si

$$n = 0 \quad T(0) = 2$$

car si on a +1 test d'égalité et +1 pour le return 1.

c)

Preuve par récurrence que :

$$T(n) = 5n + 2$$

- *Hypothèse :*

$$T(k) = 5k + 2$$

- *Initialisation :*

$$T(0) = 2, oui$$

- *Hérédité :* On a

$$\begin{aligned} T(k+1) &= 5 + T(k) \\ &= 5 + 5k + 2 \\ &= 5(k+1) + 2 \end{aligned}$$

  Donc l'hypothèse est vérifiée

d)

Complexité de la fonction factorielle :

$$\mathcal{O}(n) \rightarrow linraire$$

2)

a)

```python
def fact_it(n):
    r = 1
    i = 1
    while i <= n:
        r *= i
        i += 1
    return r
```

```python
1 def euler_it(n):
2     i = 1
3     somme = 0
4     while i <= n:
5         somme += 1/fact_it(i)
6         i += 1
7     return somme
```

b)

- Preuve de terminaison (ou d'arrêt)
  Il y a un nombre fini d'opération, donc ça termine.
- Preuve de validité (ou de correction)
  Montrons par récurrence que :

$$e_i = \sum_{k=0}^{i} \frac{1}{k!}$$

  – Initialisation :

$$Pour \quad i = 0 \qquad on \quad a : \quad \frac{1}{0!} = \frac{1}{1} = 1$$

– Hérédité :

Soit :

$$P_n = \sum_{k=0}^{n} \frac{1}{k}$$

Vraie Montrons que :

$$P_{(n+1)} = \sum_{k=0}^{n+1} \frac{1}{(k+1)!}$$

Vraie aussi.

$$\begin{aligned} P_{(n+1)} &= 1 + \frac{1}{1!} + \frac{1}{2!} + ... + \frac{1}{n!} + \frac{1}{(n+1)!} \\ &= P_n + \frac{1}{(n+1)!} \\ &= P_{(n+1)} \end{aligned}$$

c)

Complexité de la fonction euler_it :

ligne 2 : +1 affectation

ligne 3 : +1 affectation

ligne 4 :

$$while \equiv \sum_{i=1}^{n} (5i + 7)$$

$5i + 7 \quad car \quad :$

ligne 5 : $+= \equiv$ affection et somme, $+2$
fraction : $+1$
fact_it(i) $\equiv 5i + 2$ (démontré avant)
ligne 6 : $+= \equiv$ affectation et somme, $+2$

$$\equiv 2 + 1 + (5i + 2) + 2 \equiv (5i + 7)$$

Donc :

$$\sum_{i=1}^{n}(5i + 7) = \sum_{i=1}^{n} 5i + \sum_{i=1}^{n} 7 = 5\sum_{i=1}^{n} i + 7n = 5\left(\frac{n(n+1)}{2}\right) + 7n = \frac{5n^2}{2} + \frac{19n}{2}$$

d)

Ordre de grandeur asymptotique de l'algorithme itératif :

$$\mathcal{O}(n^2)$$