Travaux pratiques

Objectif

- Utiliser des boucles bornées et non bornées

Suites numériques — Approximation de ln(2)

L'objectif de ce TP est de donner une approximation de ln(2) par deux méthodes différentes.

PARTIE 1 — Utilisation d'une suite

On considère la fonction suite ci-contre écrite dans l'éditeur Python où n est un entier naturel non nul.

```python
def suite(n):
    u=0
    for k in range(1,n+1):
        u=u+1/(n+k)
    return u
```

1 Quelle est parmi les propositions suivantes, la somme que cet algorithme permet de calculer ?

a. $\sum_{k=1}^{n}\frac{1}{k}$ b. $\sum_{k=1}^{n}\frac{1}{n}$ c. $\sum_{k=1}^{n}\frac{1}{n+k}$ d. $\sum_{k=1}^{n+1}\frac{1}{n+k}$

Dans la suite de cette partie, on notera, pour n entier naturel non nul, u_n la somme précédente.

2 Donner les valeurs exactes de u_1, u_2 et u_3. ....

3 Recopier la fonction suite dans l'éditeur Python. Quel est le résultat affiché lorsqu'on écrit dans la console Python ?

a. suite(10) ? .... b. suite(100) ? ....

c. suite(1000) ? ....

4 Comparer les résultats précédents à ln(2).

....

5 Dans cette question, on admet que pour tout réel x strictement positif, $1-\frac{1}{x}\leqslant \ln(x)\leqslant x-1$.

a. En appliquant cet encadrement à $x=\frac{k+1}{k}$, avec k entier naturel non nul, montrer que $\frac{1}{k+1}\leqslant \ln(k+1)-\ln(k)\leqslant \frac{1}{k}$.

....

b. Additionner les inégalités de la question **4. a.** pour k allant de n à $2n-1$ et en déduire un encadrement de ln(2).

....

c. En déduire que $0\leqslant \ln(2)-u_n\leqslant \frac{1}{2n}$ et justifier alors $\lim\limits_{n\to+\infty} u_n$.

....

....

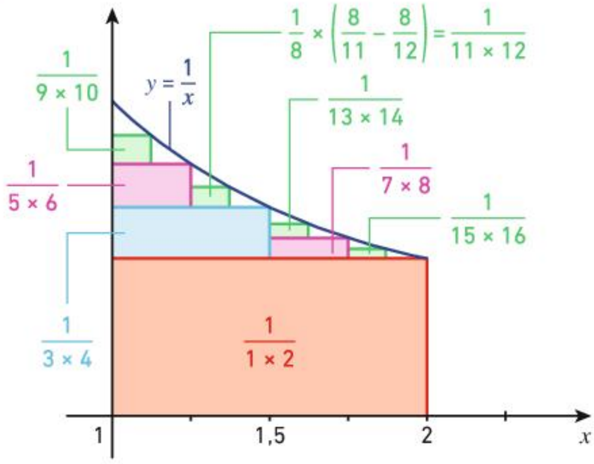
6 a. Écrire ci-contre et dans l'éditeur Python, une fonction valeurapprochée qui prend pour paramètre un entier naturel non nul p et qui renvoie le rang n ainsi que la valeur de u_n telle que $\ln(2)-u_n\leqslant 10^{-p}$.

....

b. Quelle est la valeur affichée lorsqu'on écrit dans la console Python valeurapprochée(3) ? valeurapprochée(5) ?

PARTIE 2 Utilisation de l'algorithme de Brouncker

William Brouncker est un linguiste et mathématicien anglais du XVII^e siècle. Il est l'un des premiers mathématiciens à travailler sur l'aire du domaine $\mathcal{D}$ délimité par l'hyperbole, l'axe des abscisses et les droites d'équations $x=1$ et $x=2$.



1 Donner la valeur exacte de l'aire de ce domaine $\mathcal{D}$.

Il approche l'aire du domaine $\mathcal{D}$ par la somme des aires des rectangles comme dessinés ci-contre.

On définit pour tout entier naturel n non nul, la suite (v_n) par :

$$v_n = \frac{1}{1\times 2} + \ldots + \frac{1}{(2n-1)2n} = \sum_{k=1}^{n} \frac{1}{(2k-1)2k}.$$

2 a. Calculer v_1, v_2 et v_3.

b. Quelle remarque peut-on faire ?

3 a. Compléter et recopier dans l'éditeur Python la fonction suite_Brouncker ci-contre de paramètre n entier naturel non nul afin qu'elle renvoie une valeur approchée de v_n.

```
def suite_Brouncker(n):
    v=0
    for k in range(........):
        v=..................
    return v
```

b. Quels sont les résultats affichés lorsqu'on écrit dans la console Python suite_Brouncker(10) ? suite_Brouncker(100) ? suite_Brouncker(1000) ?

4 a. Écrire ci-contre et dans l'éditeur Python une fonction valeurapprochée_Brouncker qui prend pour paramètre un entier naturel non nul p et qui renvoie tous les termes de la suite (v_n) et le plus petit rang n tel que $\ln(2) - v_n \leqslant 10^{-p}$.

Cette fonction utilisera une liste et la fonction suite_Brouncker précédente.

b. Quel est le résultat affiché lorsqu'on saisit dans la console Python valeurapprochée_Brouncker(1) ?