

## Suites numériques, algorithmique et programmation en Python

```
def expl(n):  
    u=(n+1)/n  
    return u  
  
def recur(n):  
    i=0  
    u=1  
    while i<n:  
        u=(u+1)/u  
        i=i+1  
    return(u)
```

- 1) Que renvoient ces deux fonctions lorsque l'argument  $n=4$  ?
- 2) Programmer ces deux fonctions dans un programme que vous complétez au fur et à mesure de l'année nommée **premiere**. Appeler chacune de ces deux fonctions dans la console d'exécution pour vérifier les résultats précédents
- 3) Ecrire une fonction nommée **qrest** permettant de calculer quantité restante d'une cuve pleine de 100 litres de liquide volatile qui s'évapore de 20% par mois et à qui l'on rajoute 10 litres chaque mois.
- 4) La cuve va-t-elle déborder ou se vider au bout d'un très grand nombre de mois ?
- 5) Modifier la fonction **qrest** pour déterminer au bout combien de mois la cuve aura perdu 25%, 40% puis 50% de son volume.

## Suites numériques, algorithmique et programmation en Python

```
def expl(n):  
    u=(n+1)/n  
    return u  
  
def recur(n):  
    i=0  
    u=1  
    while i<n:  
        u=(u+1)/u  
        i=i+1  
    return(u)
```

- 1) Que renvoient ces deux fonctions lorsque l'argument  $n=4$  ?
- 2) Programmer ces deux fonctions dans un programme que vous complétez au fur et à mesure de l'année nommée **premiere**. Appeler chacune de ces deux fonctions dans la console d'exécution pour vérifier les résultats précédents
- 3) Ecrire une fonction nommée **qrest** permettant de calculer quantité restante d'une cuve pleine de 100 litres de liquide volatile qui s'évapore de 20% par mois et à qui l'on rajoute 10 litres chaque mois.
- 4) La cuve va-t-elle déborder ou se vider au bout d'un très grand nombre de mois ?
- 5) Modifier la fonction **qrest** pour déterminer au bout combien de mois la cuve aura perdu 25%, 40% puis 50% de son volume.

## Suites numériques, algorithmique et programmation en Python

```
def expl(n):  
    u=(n+1)/n  
    return u  
  
def recur(n):  
    i=0  
    u=1  
    while i<n:  
        u=(u+1)/u  
        i=i+1  
    return(u)
```

- 1) Que renvoient ces deux fonctions lorsque l'argument  $n=4$  ?
- 2) Programmer ces deux fonctions dans un programme que vous complétez au fur et à mesure de l'année nommée **premiere**. Appeler chacune de ces deux fonctions dans la console d'exécution pour vérifier les résultats précédents
- 3) Ecrire une fonction nommée **qrest** permettant de calculer quantité restante d'une cuve pleine de 100 litres de liquide volatile qui s'évapore de 20% par mois et à qui l'on rajoute 10 litres chaque mois.
- 4) La cuve va-t-elle déborder ou se vider au bout d'un très grand nombre de mois ?
- 5) Modifier la fonction **qrest** pour déterminer au bout combien de mois la cuve aura perdu 25%, 40% puis 50% de son volume.

## Suites numériques, algorithmique et programmation en Python

```
def expl(n):  
    u=(n+1)/n  
    return u  
  
def recur(n):  
    i=0  
    u=1  
    while i<n:  
        u=(u+1)/u  
        i=i+1  
    return(u)
```

- 1) Que renvoient ces deux fonctions lorsque l'argument  $n=4$  ?
- 2) Programmer ces deux fonctions dans un programme que vous complétez au fur et à mesure de l'année nommée **premiere**. Appeler chacune de ces deux fonctions dans la console d'exécution pour vérifier les résultats précédents
- 3) Ecrire une fonction nommée **qrest** permettant de calculer quantité restante d'une cuve pleine de 100 litres de liquide volatile qui s'évapore de 20% par mois et à qui l'on rajoute 10 litres chaque mois.
- 4) La cuve va-t-elle déborder ou se vider au bout d'un très grand nombre de mois ?
- 5) Modifier la fonction **qrest** pour déterminer au bout combien de mois la cuve aura perdu 25%, 40% puis 50% de son volume.