Terminale Spécialité / Algorithmes

1. Suites : boucles itératives

E.1 On considère la suite numérique (v_n) définie par :

$$v_0 = 1 \quad ; \quad v_{n+1} = \frac{9}{6 - v_n}$$

(1) On souhaite écrire un algorithme affichant, pour un entier naturel n donné, tous les termes de la suite, du rang 0 au rang n.

Parmi les trois algorithmes suivants, un seul convient. Préciser lequel en justifiant la réponse.

Algorithme 1

Variables :
v est un réel.
i et n sont des entiers naturels

Début algorithme :
Lire n
v prend la valeur 1

Pour i allant de 1 à n faire
v prend la valeur $\frac{9}{6-v}$
Fin pour
Afficher v

Fin algorithme

Algorithme 2

Variables :
v est un réel.
i et n sont des entiers naturels

Début algorithme :
Lire n

Pour i allant de 1 à n faire
v prend la valeur 1
v prend la valeur $\frac{9}{6-v}$
Fin pour
Afficher v

Fin algorithme

Algorithme 3

Variables :
v est un réel.
i et n sont des entiers naturels

Début algorithme :
Lire n
v prend la valeur 1

Pour i allant de 1 à n faire
Afficher v
v prend la valeur $\frac{9}{6-v}$
Fin pour
Afficher v

Fin algorithme

(2) Pour $n=10$, on obtient l'affichage suivant :

1	1,800	2,143	2,333	2,455	2,538	2,600	2,647	2,684	2,714

Pour $n=100$, on obtient l'affichage suivant :

2,967	2,968	2,968	2,968	2,969	2,969	2,969	2,970	2,970	2,970

Quelles conjectures peut-on émettre concernant la suite (v_n) ?

E.2 On considère la suite (u_n) définie par $u_0=1$ et, pour tout entier naturel n : $u_{n+1} = \sqrt{2 \cdot u_n}$

On considère la fonction `f` issue d'un algorithme où l'argument `n` est un entier naturel non-nul :

```
Fonction f(n)
    u ← 1
    Pour i variant de 1 à n
        u ← √(2·u)
    Fin de Pour
    Renvoyer u
```

(1) Donner une valeur approchée à 10^{-4} près de la valeur renvoyée par cette fonction lorsque son appel s'effectue avec pour argument la valeur `n`=3.

(2) Quelle interprétation peut-on donner de la valeur renvoyée par la fonction `u` ?

(3) Par plusieurs appels à la fonction `f`, on a obtenu le tableau ci-dessous :

n	1	5	10	15	20
Valeur renvoyée	1,4142	1,9571	1,9986	1,9999	1,9999

Quelle conjecture peut-on émettre concernant la suite (u_n) ?

E.3 ***L'exercice n'existe pas.***

E.4 On considère l'algorithme suivant :

```
Fonction f(p)
    u ← 5
    Pour k variant de 1 à p
        u ←  0,5u+0,5(k−1)−1,5
    Fin de Pour
    Renvoyer u
```

On effectue un appel à la fonction avec pour valeur du paramètre `p`=2.
Construire un tableau avec les valeurs des variables `k`, `p` et `u` au cours de cet appel à la fonction `f`.
Quel nombre obtient-on en sortie de l'appel à la fonction `g` ?

E.5 On considère la fonction f extrait d'un algorithme où l'appel s'effectue en fournissant un argument `n` entier de valeur supérieur ou égale à 1 :

```
Fonction f(n)
    A ← 1
    B ← 1
    Pour K variant de 1 à n
        A ← (A+√(A²+B²))/3
        B ← B/3
    Fin Pour
    Renvoyer A
```

On appelle cette fonction avec la valeur 2 pour l'argument `n`. Recopier et compléter le tableau ci-dessous contenant l'état des variables au cours de cet appel à la fonction `f` *(on arrondira les valeurs calculées à 10^{-4} près)*

K	A	B
1		
2		

2. Suites : boucles conditionnelles à étudier

E.6 On considère la suite (p_n) définie par :
$p_1 = 0 \quad ; \quad p_{n+1} = 0{,}2 \cdot p_n + 0{,}04$ pour tout $n \in \mathbb{N}^*$

On admet que la suite (p_n) est croissante et converge vers 0,05.

On considère la fonction `f` ci-dessous issue d'un algorithme où l'argument `k` fourni lors de l'appel est un entier supérieur ou égal à 2 :

```
Fonction f(k)
    P ← 0
    J ← 1
    Tant que  P<0,05−10^-k
        P ←  0,2× P+0,04
        J ←  J+1
    Fin Tant que
    Renvoyer J
```

1. Comment interpréter la valeur renvoyée par la fonction `f` relativement à la valeur de l'argument `k` fourni lors de l'appel à cette fonction?

2. Pourquoi est-on sûr que l'appel à la fonction `f` s'arrête?

E.7 Soit f la fonction définie sur $\left[0\,;+\infty\right[$ par : $f(x)=2x\cdot\mathrm{e}^{-x}$

On admet que la fonction f admet la limite : $\lim\limits_{x\mapsto+\infty} f(x)=0$

On donne l'algorithme suivant :

```
t ← 3,5
p ← 0,25
C ← 0,21
Tant que  C>5×10^-3
    t ←  t+p
    C ← f(t)
Fin Tant que
```

En considérant une exécution pas à pas de l'algorithme, compléter le tableau ci-dessous avec les valeurs prises par les variables `p`, `t` et `C` au cours de son exécution.

Arrondir les valeurs à 10^{-2} près.

	Initialisation	Etape 1	Etape 2
p	0,25		
t	3,5		
C	0,21		

E.8 On définit la suite (d_n) par :

$$d_0=1 \quad ; \quad d_{n+1}=\frac{1}{2}\cdot {d_n}^2 \quad \text{pour tout } n\in\mathbb{N}$$

On considère la fonction `f`, issue d'un algorithme, prenant pour argument `p` un entier strictement positif :

```
Fonction f(p)
    d ← 1
    n ← 0
    Tant que  d>10^-p
        d ← 0,5·d²
        n ←  n+1
    Fin Tant que
    Renvoyer n
```

En appelant la fonction `f` avec la valeur 9 pour l'argument `p`, celle-ci renvoie le nombre 5.

En déduire l'inégalité vérifiée par le nombre d_5?

3. *Suites : boucles conditionnelles à construire*

E.9 Une société produit des bactéries pour l'industrie. En laboratoire, il a été mesuré que, dans un milieu nutritif approprié, la masse de ces bactéries, mesurée en grammes, augmente de 20 % en un jour.
La société met en place le dispositif industriel suivant.
Dans une cuve de milieu nutritif, on introduit initialement 1 kg de bactéries. Ensuite, chaque jour, à heure fixe, on remplace le milieu nutritif contenu dans la cuve. Durant cette opération, 100 g de bactéries sont perdus.
L'entreprise se fixe pour objectif de produire 30 kg de bactéries.

On modélise l'évolution de la population de bactéries dans la cuve par la suite (u_n) définie de la façon suivante :

$$u_0=1\,000 \quad ; \quad u_{n+1}=1{,}2\cdot u_n-100$$

L'entreprise souhaite savoir au bout de combien de jours la masse de bactéries dépassera 30 kg.

On peut utiliser l'algorithme suivant pour répondre au problème posé.

Recopier et compléter cet algorithme.

```
u ← 1000
n ← 0
Tant que ...
    u ← ...
    n ←  n+1
Fin Tant que
```

E.10 On considère la suite (u_n) définie par :

$$u_n=\frac{\ln(n)}{n} \quad \text{pour tout } n\in\mathbb{N}^*$$

On admet que :

- La suite (u_n) est strictement décroissante à partir du terme de rang 2.
- La suite (v_n) est convergente et converge vers 0.

Écrire un algorithme déterminant le plus petit entier n_0 supérieur ou égal à 2 tel que $\left|u_{n_0}\right|\leqslant 10^{-2}$

4. *Suites et boucles*

E.11 On considère la fonction `f`, extrait d'un algorithme, où la valeur passée en argument est un entier naturel.

```
Fonction f(N)
```



```
    U ← 0
    Pour k allant de 0 à  N−1
        U ←  3·U−2k+3
    Fin pour
    Renvoyer U
```

1. Quelle est la valeur renvoyée par la fonction `f` lorsque la valeur passée en argument est `N=3`?
2. On considère la suite (u_n) définie par :

$$u_0 = 0 \quad ; \quad u_{n+1} = 3 \cdot u_n - 2 \cdot n + 3 \quad \text{pour tout } n \in \mathbb{N}$$

On admet que la suite (u_n) est croissante et admet pour limite :

$$\lim_{n \mapsto +\infty} u_n = +\infty$$

Proposer une fonction `f` d'un algorithme qui, pour une valeur p passée en argument, renvoie la valeur du plus petit entier n_0 tel que :

Pour tout $n \geqslant n_0$, on ait : $u_n \geqslant 10^p$

5. Somme des termes d'une suite

E.12 Soit (u_n) la suite définie pour tout entier strictement positif par :

$$u_n = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} - \ln n$$

1. On considère la fonction `f` ci-dessous, extrait d'un algorithme et prenant pour argument `n` un entier strictement positif. :

```
Fonction f(n)
    u ← 0
    Pour i variant de 1 à n
        u ← u+1/i
    Fin Pour
    Renvoyer u
```

Donner la valeur exacte renvoyée par cette fonction lorsque l'utilisateur appelle la fonction `f` avec la valeur $n=3$.

2. Recopier et compléter l'algorithme précédent afin que la valeur renvoyée soit le terme u_n de rang n lorsque la fonction `f` est appelée avec la valeur n.
3. Voici les résultats fournis par l'algorithme modifié, arrondis à 10^{-3}.

n	4	5	6	7	8	9	10	100	1000	1500	2000
u_n	0,697	0,674	0,658	0,647	0,638	0,632	0,626	0,582	0,578	0,578	0,577

À l'aide de ce tableau, formuler des conjectures sur le sens de variation de la suite (u_n) et son éventuelle convergence.

E.13 Soit (v_n) la suite définie par :

$$v_1 = \ln 2 \quad ; \quad v_{n+1} = \ln\left(2 - \mathrm{e}^{-v_n}\right) \text{ pour tout } n \in \mathbb{N}$$

On admet que cette suite est définie pour tout entier naturel n non nul. On définit ensuite la suite (S_n) pour tout entier naturel n non-nul par :

$$S_n = \sum_{k=1}^{n} v_k = v_1 + v_2 + \cdots + v_n$$

1. Recopier et compléter la fonction `f` qui renvoie la valeur de S_n pour une valeur de n passée en argument :

```
Fonction f(n)
    v ← ...
    S ← ...
    Pour k variant de ... à ... faire
        ... ← ...
        ... ← ...
    Fin Pour
    Renvoyer S
```

2. Par appels successifs de cette fonction, on obtient quelques valeurs de S_n. Les valeurs arrondies au dixième sont données dans le tableau ci-dessous :

n	10	100	1 000	10 000	100 000	1 000 000
S_n	2,4	4,6	6,9	9,2	11,5	13,8

En expliquant votre démarche, émettre une conjecture quant au comportement de la suite (S_n).

E.14 On considère la suite (A_n) dont les termes sont obtenus par l'étude des valeurs successives prises par la variable `A` lors de l'exécution pas à pas de cet algorithme :

```
Fonction f(n)
    A ← 0
    Pour k allant de 0 à  n−1
        A ←  A+1/2×sin(2π/n)×(1+n/k)(1+(k+1)/n)
    Fin Pour
```

On appelle la fonction `f` avec la valeur 10 pour l'argument `n`.

Recopier et compléter, en arrondissant au millième près, le tableau ci-dessous qui illustre le fonctionnement de l'algorithme :

k	0	1	2	3	4	5	6
A	0,323	0,711	1,170	1,705	2,322	3,027	3,826

k	7	8	9
A	4,726		

6. Suites définies conjointement

E.15 On considère les deux suites (x_n) et (y_n) définies par :

$$\begin{cases} x_0 = -1 \\ x_n = \dfrac{5}{4} \cdot x + \dfrac{3}{4} \cdot y \end{cases} \quad ; \quad \begin{cases} y_0 = 5 \\ y_n = \dfrac{3}{4} \cdot x + \dfrac{5}{4} \cdot y \end{cases}$$

On considère la fonction `f` d'un algorithme présentée ci-dessous. Appeler avec un argument `n` entier supérieur ou égal à 1, son exécution permet de renvoier le couple $(x_n\,;y_n)$ dont les coordonnées sont les valeurs des termes des suites (x_n) et (y_n) de rang `n`.

La fonction ne renvoye pas les valeurs attendues. Modifier le code de cette fonction en conséquence :

```
Fonction f(n)
    x ← −1
    y ← 5
    Pour i allant de 1 à n
        x ← 5/4 x + 3/4 y
        y ← 3/4 x + 5/4 y
    Fin Pour
    Renvoyer (x; y)
```

E.16 On considère deux suites de nombres réels (d_n) et (a_n) définies par $d_0=300$, $a_0=450$ et, pour tout entier naturel $n\geqslant 0$:

$$\begin{cases} d_{n+1} = \dfrac{1}{2}\cdot d_n + 100 \\ a_{n+1} = \dfrac{1}{2}\cdot d_n + \dfrac{1}{2}\cdot a_n + 70 \end{cases}$$

1. Calculer d_1 et a_1.
2. On souhaite écrire une fonction dans un algorithme qui prendra pour argument un entier naturel n et qui renverra le couple de valeurs $(\,d_n\,;a_n\,)$ associé au rang n.

   On propose la fonction suivante est proposée :

```
Fonction f(n)
    D ← 300
    A ← 450
    Pour k variant de 1 à n
        D ← D/2 +100
        A ← A/2 + D/2 +70
    Fin pour
    Renvoyer (D ; A)
```

a) Quel couple de nombres est renvoyé par l'appel à la fonction f avec pour argument `n=1`?
Ces résultats sont-ils cohérents avec ceux obtenus à la question 1?

b) Corriger cette fonction pour qu'elle renvoie les résultats souhaités.

E.17 On considère l'algorithme suivant :

Entrée	Saisir un réel strictement positif non nul a. Saisir un réel strictement positif non nul b $(b>a)$ Saisir un entier naturel non nul N
Initialisation	Affecter à u la valeur a Affecter à v la valeur b Affecter à n la valeur 0
Traitement	TANT QUE : $n<N$ Affecter à n la valeur $n+1$ Affecter à u la valeur $\dfrac{a+b}{2}$ Affecter à v la valeur $\sqrt{\dfrac{a^2+b^2}{2}}$ Affecter à a la valeur u Affecter à b la valeur v.
Sortie	Afficher u, afficher v

Reproduire et compléter le tableau suivant, en faisant fonctionner cet algorithme pour $a=4$, $b=9$ et $N=2$. Les valeurs successives de u et v seront arrondies au millième.

n	a	b	u	v
0	4	9		
1				
2				

7. *Vers les probabilités*

E.18 On considère l'algorithme :

```
C ← 0
Pour i allant de 1 à 9
    A ← valeur aléatoire entière entre 1 et 7
    Si  A>5 Alors
        C ← de  C+1
    Fin Si
Fin Pour
```

Dans l'expérience aléatoire simulée par l'algorithme précédent, on appelle $\mathcal{X}$ la variable aléatoire prenant la valeur de la variable `C` en fin d'exécution de l'algorithme.

Quelle loi suit la variable $\mathcal{X}$? Préciser ses paramètres.

8. *Prévoir le fonctionnement d'un algorithme*

E.19 Soit m et m' deux entiers relatifs.
On considère l'équation (E) définie par :

$$\left(\frac{m\cdot m'}{4}\right)^2 + (m-1)\cdot(m'-1) + \frac{m\cdot m'}{4} = 0$$

On considère l'algorithme suivant :

```
Pour m allant de −10 à 10
    Pour m' allant de −10 à 10
        Si (m·m')²+16·(m−1)·(m'−1)+4·m·m'=0
            Alors (a ; b) ← (m ; m')
        Fin Si
```

```
    Fin du Pour
Fin du Pour
```

Lors de l'exécution pas à pas, on s'intéresse aux valeurs prises successivement par les variables `a` et `b`.

1. Quel est le rôle de cet algorithme?
2. Lors de l'exécution de cet algorithme, le couple $(\mathtt{a}\,;\mathtt{b})$ se verra affecter de six couples d'entiers dont :
   $(-4;1)$; $(0;1)$; $(5;-4)$.
   Écrire les six couples dans l'ordre de leur affectation successive au cours de l'exécution de l'algorithme.

E.20 Un groupe de 50 coureurs, portant des dossards numérotés de 1 à 50, participe à une course cycliste qui comprend 10 étapes, et au cours de laquelle aucun abandon n'est constaté.
À la fin de chaque étape, un groupe de 5 coureurs est choisi au hasard pour subir un contrôle antidopage. Ces désignations de 5 coureurs à l'issue de chacune des étapes sont indépendantes. Un même coureur peut donc être contrôlé à l'issue de plusieurs étapes.

1. À l'issue de chaque étape, combien peut-on former de groupes différents de 5 coureurs?
2. On considère l'algorithme ci-dessous dans lequel :
   - "rand(1,50)" permet d'obtenir un nombre entier aléatoire appartenant à l'intervalle $[1\,;50]$;
   - l'écriture "x:=y" désigne l'affectation d'une valeur y à une variable x.

```
a ← 0
b ← 0
c ← 0
d ← 0
e ← 0
Tant que  (a=b) ou (a=c) ou (a=d) ou
(a=e)
       ou (b=c) ou (b=d) ou (b=e) ou
(c=d)
       ou (c=e) ou (d=e)
    a ← rand(1,50)
    b ← rand(1,50)
    c ← rand(1,50)
    d ← rand(1,50)
    e ← rand(1,50)
Fin Tant que
```

On s'intéresse à l'ensemble composé de 5 entiers naturels formés par les valeurs des variables `a`, `b`, `c`, `d`, `e` obtenues à l'issue de l'exécution de l'algorithme.

a. Parmi les ensembles de nombres suivants, lesquels ont pu être obtenus à l'aide de cet algorithme :
$L_1=\{2\,;11\,;44\,;2\,;15\}$; $L_2=\{8\,;17\,;41\,;34\,;6\}$
$L_3=\{12\,;17\,;23\,;17\,;50\}$; $L_4=\{45\,;19\,;43\,;21\,;18\}$

b. Que permet de réaliser cet algorithme concernant la course cycliste?

E.21 Voici un algorithme applicable à des entiers de trois chiffres dont le chiffre des centaines n'est pas égal à celui des unités :

Etape 1 : Inverser l'ordre des chiffres *(par exemple 275 devient 572)*
Etape 2 : Calculer la différence du plus grand et du plus petit de ces deux nombres.
Etape 3 : Ré-itérer l'étape 1 sur le nombre obtenu.
Etape 4 : Additionner ces deux derniers nombres

1. a. Appliquer l'algorithme aux nombres 123, 448 et 946.
   b. Que peut-on conjecturer?
2. Pour implémenter cet algorithme, l'étape 2, implicite lorsqu'on effectue les calculs *"à la main"*, nécessite de dissocier l'entier saisi d'en isoler le chiffre des unités, celui des dizaines puis celui des centaines.

   Compléter la fonction suivante, issue d'un algorithme, dont le rôle est de prendre en argument un entier `n` de trois chiffres et d'effectuer cette dissociation. Dans cet algorithme a est le chiffre des centaines, b celui des dizaines et c celui des unités du nombre n que l'on souhaite décomposer.

```
Fonction f(n)
    a ← 0
    b ← 0
    c ← 0
    Tant que n⩾100
        a ← a+1
        n ← n−100
    Fin Tant que
    Tant que n......
        b ← ......
        ...... ← ......
    Fin Tant que
    c la valeur ......
    Renvoyer (a;b;c)
```

9. *Utilisation de la calculatrice*

E.22 Soit f la fonction définie sur $\mathbb{R}$ par : $f(x)=x-\ln(x^2+1)$

On admet que la fonction f admet le tableau de variations suivant :

x	$-\infty$ $\qquad\qquad$ $+\infty$
Variation de f	$-\infty$ $\nearrow$ $+\infty$

On considère l'algorithme suivant :

```
Fonction g(A)
```



```
    N ← 0
    Tant que  N−ln(N²+1)<A
        N ← N+1
    Fin Tant que
    Renvoyer N
```

où la fonction `g` est appelée avec un argument `A` qui est un nombre réel.

1. Quel sens donne-t-on à la valeur renvoyée par la fonction `g`?
2. Déterminer la valeur `N` renvoyée par l'appel à la fonction `g` est effectuée avec la valeur `100` de son paramètre `A`.

E.23 Dire si l'affirmation ci-dessous est vraie ou fausse en justifiant la réponse.

Soit f la fonction définie sur $\mathbb{R}$ par : $f(x)=\dfrac{3}{4+6\cdot e^{-2x}}$

En fin d'exécution, cet algorithme affecte à la variable `X` la valeur 0,54.

```
X ← 0
Y ← 3/10
Tant que  Y<0,5
    X ← X+0,01
    Y ← 3/(4+6·e^(−2X))
Fin Tant que
```

10. *Autour de la dichotomie*

E.24 On considère l'algorithme suivant :

```
Fonction f(x)
    Renvoyer ....

Fonction g(a,b)
    Tant que  b−a>0,3
        x ← (a+b)/2
        Si  f(x)·f(a)>0
            alors a ← x
        sinon b ← x
        Fin Si
    Fin Tant que
    Renvoyer (a+b)/2
```

Indiquer si l'affirmation ci-dessous est vraie ou fausse et justifier la réponse.

> On complète l'algorithme pour que la fonction `f` puisse renvoyer les images du paramètre x pour la fonction :
> $f(x)=x^3-3$.
> On effectue un appel à la fonction `g` avec les valeurs des paramètres `a=1` et `b=2`.
> La valeur renvoyée par cet appel à la fonction `g` est le nombre 1,6875

E.25 On considère la fonction f définie sur $[0\,;5]$ par :
$f(x)=e^x-1$

On admet que la fonction f est strictement croissante et on note m la valeur e^5-1.

On considère l'algorithme ci-dessous :

```
a ← 2
b ← 2e
Tant que  b−a>10^(−3)
    c ← (a+b)=2
    Si  f(c)<3,5
        Alors a ← c
        Sinon b ← c
    Fin Si
Fin Tant que
d ← f(c)
```

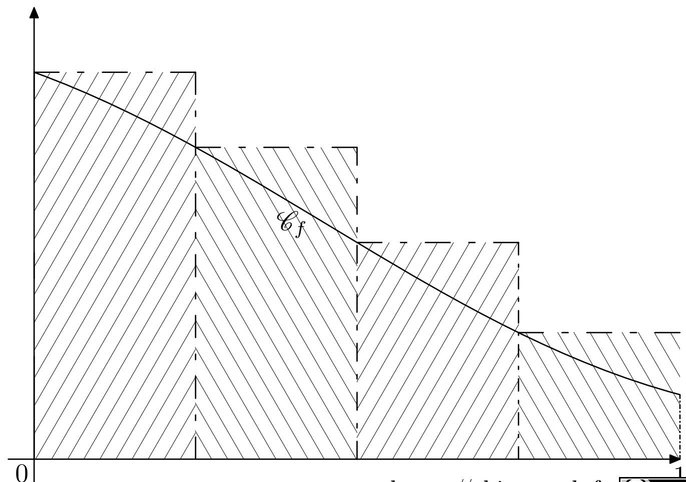
Interpréter la valeur de la variable `d` en fin d'exécution de l'algorithme.

11. *Autour des intégrales*

E.26 On considère une fonction f décroissante sur l'intervalle $[0\,;1]$.
On note $\mathscr{C}$ la courbe représentative de la fonction f dans un repère orthogonal.

On note $\mathcal{D}$ le domaine compris entre l'axe des abscisses, la courbe $\mathscr{C}$ et les droites d'équations $x=0$ et $x=1$.

1. On représente ci-dessous une approximation de l'aire du domaine $\mathcal{D}$ à l'aide des quatre rectangles ci-dessous :



Compléter l'algorithme ci-dessous afin que la valeur de la variable `S`, en fin d'exécution de l'algorithme, soit l'aire formée par les quatre rectangles :

```
S ← 0
Pour k variant de 0 à
...
    S ← ...
Fin Pour
```

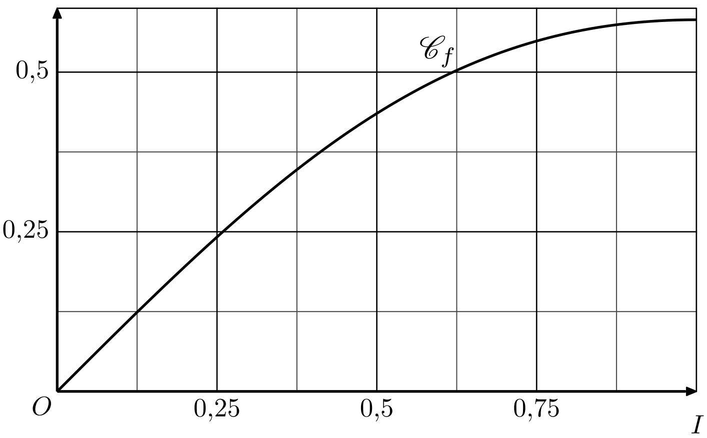
(2) Dans cette question, N est un nombre entier strictement supérieur à 1. On découpe l'intervalle $[0\,;1]$ en N intervalles de même longueur. Sur chacun des intervalles, on construit un rectangle en procédant de la même manière qu'à la question précédente.

Modifier l'algorithme précédent afin que la valeur de la variable S en fin d'exécution de l'algorithme soit la somme des aires des N rectangles ainsi construits.

E.27 Soit f une fonction définit l'intervalle $[0\,;1]$ par :

$$f(x) = \frac{x}{e^x - x}$$

et dont la courbe représentative $\mathscr{C}_f$ est donnée ci-dessous :



On admet que la fonction f est positive sur l'intervalle $[0\,;1]$.

On définit dans un algorithme la fonction `g` dans lequel les variables sont :

- K et i des entiers naturels, K étant non nul ;
- A, x et h des réels.

```
Fonction g(K)
    A ← 0
    x ← 0
    h ← 1/K
    Pour i variant de 1 à K
        A ←  A+h×f(x)
        x ←    x+h
    Fin pour
    Renvoyer A
```

(1) Reproduire et compléter le tableau ci-dessous en y indiquant les valeurs des variables `A` et `x` lorsque la fonction `g` s'exécute pas à pas.
On arrondira les valeurs successives de `A` au millième près.

i	A	x
1		
2		
3		
4		

(2) En l'illustrant sur la représentation graphique ci-dessus, donner une interprétation graphique de la valeur renvoyée par la fonction `g` lorsque l'argument passé à pour valeur `K=8`.

(3) Que peut-on dire de la valeur renvoyée par la fonction `g` lorsque `K` devient grand?