

wxMaxima pour l'Analyse Volume II

Auteur : Zachary Hannan, Solano Community College
Les corrections sont réalisées par Michel Gosse

Exercices du Chapitre 6 et leur corrigé

Présentation

Le manuel wxMaxima for Calculus Volume II, dont l'auteur est Zachary Hannan, propose l'étude de nombreux domaines de l'analyse à l'aide du logiciel Maxima et de son interface graphique wxMaxima. A l'occasion de la traduction française de ce manuel, la correction des exercices de chaque chapitre est proposée. Ce document reprend les exercices du manuel et leur correction à l'aide de wxMaxima.

La dernière version de ce document, ainsi que les fichiers source pour wxMaxima, sont téléchargeables sur le site <https://maxima-french-doc.fr/>, à la rubrique Documentation.

Enoncé des exercices

1. Représentez les premiers termes de la suite de terme général $a_n = f(n)$ où $f(x) = \frac{\sqrt{x^2-4}}{x+3}$ et calculez $\lim_{n \rightarrow \infty} a_n$.
2. Tracez les termes de la suite donnée par $a_n = \frac{3n^4-2n}{n^6}$ ainsi que la suite des sommes partielles $S_n = \sum_{k=1}^n a_k$. Calculez $\lim_{n \rightarrow \infty} S_n$ en utilisant simplement `sum` avec $n = \infty$.
3. On peut définir un « théorème des gendarmes » pour la limite d'une suite : si $a_n \leq b_n \leq c_n$ pour tout $n \geq N$, et $\lim_{n \rightarrow \infty} a_n = \lim_{n \rightarrow \infty} c_n = L$, alors $\lim_{n \rightarrow \infty} b_n = L$. Pour illustrer le théorème des gendarmes, tracez les termes de $\{-\frac{1}{n}\}$ et $\{+\frac{1}{n}\}$ (en noir) comme bornes inférieure et supérieure des termes de $\{\frac{\sin n}{n}\}$ (en rouge). Montrez que la limite de $\{\frac{\sin n}{n}\}$ coïncide avec les limites des bornes inférieure et supérieure.
4. Si le terme de rang n d'une suite ne tend pas vers zéro lorsque n devient très grand, alors la suite des sommes partielles diverge nécessairement. Montrez que la suite $\{1 + \frac{1}{n^2}\}$ converge vers une valeur finie. Tracez 20 termes de $\{1 + \frac{1}{n^2}\}$ ainsi que la suite des sommes partielles afin d'illustrer pourquoi la suite des sommes partielles diverge.
5. Utilisez l'idée de l'exercice précédent pour montrer que $\sum_{n=1}^{\infty} \frac{\sqrt{n}}{\ln n}$ diverge.
6. Le critère de l'intégrale peut montrer la convergence à condition d'utiliser une fonction continue qui est à partir d'une certaine valeur positive et décroissante. Utilisez les graphes de $f(x) = \frac{\ln x}{x^2}$ et de $f'(x)$ avec `find_root` pour établir la valeur de x à partir de laquelle $f(x)$ est décroissante et positive. Trouvez et calculez une intégrale permettant de montrer que $\sum_{n=1}^{\infty} \frac{\ln x}{x^2}$ converge. Déterminez une borne supérieure pour la série à l'aide d'une intégrale, et illustrez votre réponse en traçant la somme de Riemann appropriée.
7. Nous pouvons utiliser l'idée du critère de l'intégrale pour déterminer des bornes d'erreur sur une somme partielle. Supposons que nous additionnions les n premiers termes d'une série pour obtenir une somme partielle S_n , et que les termes de la série soient positifs et décroissants pour $x \geq n$, et engendrés par une fonction continue $f(x)$. La somme partielle est une sous-estimation de la somme totale, et l'on appelle reste la quantité $R_n = S - S_n$, où S est la somme exacte. Si l'on considère les termes restants de la série $a_{n+1}, a_{n+2}, \dots$ comme une somme de Riemann à droite, alors l'intégrale $\int_n^{\infty} f(x) dx$ nous donne une borne supérieure sur le reste. Si l'on considère les termes restants comme une somme de Riemann à gauche, alors l'intégrale $\int_{n+1}^{\infty} f(x) dx$ nous donne une borne inférieure sur le reste.

- Calculez la somme partielle $\sum_{n=1}^{10000} \frac{n+3}{n^3+n}$, calculez les bornes supérieure et inférieure sur le reste, et écrivez un intervalle pour les valeurs possibles de S .
8. Utilisez le critère de l'intégrale pour montrer que $\sum_2^\infty \frac{1}{\sqrt{x} \ln x}$ diverge.
 9. Utilisez le critère de comparaison avec calcul de la limite pour déterminer si la série $\sum_{n=1}^\infty \frac{\sqrt{3n^2-1}}{n^3+5n}$ converge ou non.
 10. Utilisez le critère de d'Alembert pour montrer que $\sum_{n=1}^\infty \frac{n}{2^n}$ converge. Utilisez ensuite le critère de comparaison avec passage à la limite pour montrer que $\sum_{n=1}^\infty \frac{n}{2^{n-1}}$ converge.
 11. Tentez d'utiliser le critère de l'intégrale pour montrer que $\sum_{n=1}^\infty \frac{\ln n}{2^{n-1}}$ converge. Que se passe-t-il ? Utilisez une boucle `do` pour afficher les sommes partielles pour $n = 100, 200, \dots, 2000$. La série semble-t-elle converger ? Utilisez les résultats de l'exercice précédent pour montrer que la série converge (vous devrez établir que $\ln n < n$).
 12. Montrer que la série $\sum_{n=1}^\infty \frac{(-1)^n n^{10}}{e^n}$ converge absolument.
 13. Montrer que la série $\sum_{n=1}^\infty (-1)^n \left(\frac{\pi}{2} - \tan^{-1} n \right)$ est semi-convergente.
 14. Utilisez `taylor` pour calculer les dix premiers termes de la série de Taylor de $\ln x$ centrée en $x = 1$. Trouvez une formule explicite pour le $n^{\text{ième}}$ terme du développement, et calculez l'intervalle de convergence. Montrez que la série converge en $x = 2$, puis exprimez $\ln 2$ comme une série infinie. Quel est le nom usuel de cette série ?
 15. Calculez les dix premiers termes de la série de Maclaurin pour $f(x) = (1+x)^n$ en utilisant `taylor`. Quelle est l'erreur commise par l'approximation linéaire lorsque $x = 0.001$?
 16. Utilisez `taylor` pour écrire au moins les dix premiers termes de la série de Maclaurin pour e^{ix} (le symbole de l'unité imaginaire est `%i` dans wxMaxima). Séparez (mentalement) les termes réels et imaginaires pour obtenir deux séries infinies distinctes. Reconnaissez-vous ces deux séries ? Enfin, écrivez e^{ix} comme une combinaison linéaire de $\cos x$ et $\sin x$.
 17. Utilisez `taylor` pour écrire la série de Maclaurin pour $\cosh x$. Pouvez-vous trouver un argument de la fonction cosinus produisant cette même série ? Vérifiez votre réponse avec wxMaxima.
 18. Exprimez $f(x) = \frac{1}{1+x^2}$ sous forme d'une série de Maclaurin (en gardant au moins cinq termes non nuls) à l'aide de la commande `taylor`. Trouvez ensuite une équation avec $f(x)$ à gauche et sa série de puissances à droite. Appliquez `integrate` à cette équation, puis évaluez le résultat en $x = 1$ pour obtenir une série infinie égale à π . Écrivez la formule explicite du $n^{\text{ième}}$ terme de cette série. Enfin, utilisez `sum` pour trouver une approximation de π avec 100 termes. Quelle est l'erreur commise par cette approximation ?
 19. Pour un pendule simple, une analyse des moments de force conduit à l'équation différentielle du second ordre $\frac{d^2\theta}{dt^2} = -\frac{g}{L} \sin \theta$, où g est l'accélération de la gravité, L est la longueur du pendule et θ est l'angle du pendule mesuré par rapport à la « verticale ». Cette équation différentielle est non linéaire et n'admet pas de solution sous forme algébrique. Cependant, on peut faire une approximation pour les petits angles en utilisant la série de Taylor centrée en $c = 0$: si θ est « petit », on peut tronquer la série de $\sin \theta$ après le terme de degré 1 pour linéariser l'équation différentielle.
 - (a) Quelle est l'approximation linéaire de $\sin \theta$ lorsque θ est proche de zéro ?
 - (b) Tracez $\sin \theta$ et l'approximation linéaire dans la même fenêtre sur l'intervalle $[-.5, .5]$. Rappelez-vous que l'angle est mesuré ici en radians, ce qui correspond à un déplacement d'environ 30° par rapport à la position d'équilibre.
 - (c) Calculez l'intervalle des valeurs de θ pour lesquelles l'erreur commise par l'approximation linéaire est inférieure à 5%.
 - (d) Utilisez `ode2` pour calculer la solution générale de l'équation différentielle linéarisée. Quelle est la période de la solution ?
 20. *Une *onde stationnaire* résulte d'une vibration sur une corde pour laquelle un nombre entier de demi-longueurs d'onde s'inscrit précisément sur la longueur de la corde. Pour une corde donnée, il n'existe

qu'un ensemble discret de longueurs d'onde (et de fréquences) produisant des ondes stationnaires. Pour une longueur L , une amplitude A et une vitesse d'onde v , l'équation d'une onde stationnaire est $y(x, t) = A \sin\left(\frac{n\pi x}{L}\right) \cdot \cos\left(\frac{n\pi v}{L}t\right)$ pour $n = 1, 2, 3, \dots$.

Le code suivant utilise $L = 1$, $v = 1$, $n = 3$ et $A = 1$ pour créer une animation montrant une onde stationnaire avec des scènes calculées toutes les 0.02 s pendant 10 s :

```
TIMEDEP(x,t,n):=sin(n*pi*x)*cos(n*pi*t)$
TIMEDEP3:TIMEDEP(x,t,3)$
load(draw)$
wxanimate_autoplay:true;
wxanimate_framerate:20;
with_slider_draw(
  i, makelist(0.05*i, i, 0, 200),
  title=concat("t=", 0.05*i),
  grid=true,
  xaxis=true,
  yaxis=true,
  nticks=120,
  xrange=[0,1],
  yrange=[-1,1],
  color=black,
  line_width=2,
  explicit(subst(0.05*i,t,TIMEDEP3),x,0,1)
)$
```

- (a) Exécutez l'animation pour $n = 3$.
 - (b) Créez une deuxième animation pour le cas $n = 5$.
 - (c) Créez une animation combinant les ondes stationnaires $n = 3$ et $n = 5$, en donnant au cas $n = 5$ un tiers de l'amplitude du cas $n = 3$.
21. *Pour modéliser une corde de guitare pincée, on utilise une série de Fourier en sinus pour exprimer l'état initial comme une superposition de nombreuses ondes sinusoïdales qui s'annulent aux extrémités. L'avantage de cette approche est que l'évolution temporelle de chaque onde sinusoïdale est simple : chaque fonction sinus est simplement une onde stationnaire, et elle évolue selon la formule $y(x, t) = A \sin\left(\frac{n\pi x}{L}\right) \cdot \cos\left(\frac{n\pi v}{L}t\right)$.

Supposons qu'une corde de guitare soit contrainte à l'intervalle $[0, 2]$ et qu'elle soit pincée selon une forme initiale donnée par la fonction définie par morceaux : $f(x) = \begin{cases} x & 0 \leq x \leq 0.5 \\ 1 - x & 0.5 \leq x \leq 1 \\ 0 & 1 \leq x \leq 2 \end{cases}$

- (a) Calculez une approximation par une série de Fourier en sinus à 50 termes de cette fonction sur $[0, 2]$, puis tracez l'approximation avec $f(x)$ pour vérifier qu'elle fonctionne.

Remarque : `foursin` ne fonctionnera pas dans ce cas car $f(x)$ est définie par morceaux. Vous devrez calculer les coefficients en utilisant `integrate` et en divisant l'intervalle d'intégration.

- (b) En supposant $v = 5$, ajoutez la dépendance temporelle correcte à chaque terme du développement de Fourier et créez une animation montrant comment la forme de la corde évolue pendant vingt secondes après le pincement initial. Utilisez des pas de temps de 0.02 seconde pour calculer les scènes de l'animation.

- (c) Dans des ondes réalistes, les fréquences plus élevées s'atténuent plus rapidement que les fréquences plus basses. Utilisez un facteur d'amortissement $e^{-(0.1n)t}$ attaché à chaque $n^{\text{ième}}$ terme et relancez l'animation.

Si votre animation fonctionne correctement, les vibrations se stabiliseront après un certain temps vers le mode « fondamental » (dans le mode fondamental, le centre de la corde monte et descend simplement).

Corrigé des exercices

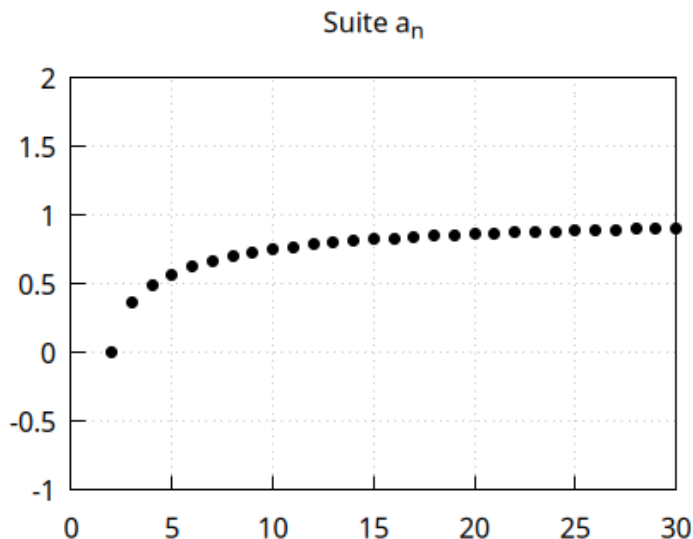
1 Exercice1

(% i1) `f(x):=sqrt(x^2-4)/(x+3);`

(%o1) $f(x) := \frac{\sqrt{x^2 - 4}}{x + 3}$

(% i2) `POINTS :makelist([n,f(n)],n,1,30)$`

(% i3) `wxdraw2d(
grid=true,
xrange=[0,30],
yrange=[-1,2],
title= Suite a_n ,
color=black,
point_type=7,
points(POINTS)
)$`



(%t3)

(% i4) `limit(f(x),x,inf);`

(%o4) 1

2 Exercice 2

(% i5) kill(all)\$

(% i1) $a(n) := (3 \cdot n^4 - 2 \cdot n) / n^6$;

(%o1) $a(n) := \frac{3n^4 - 2n}{n^6}$

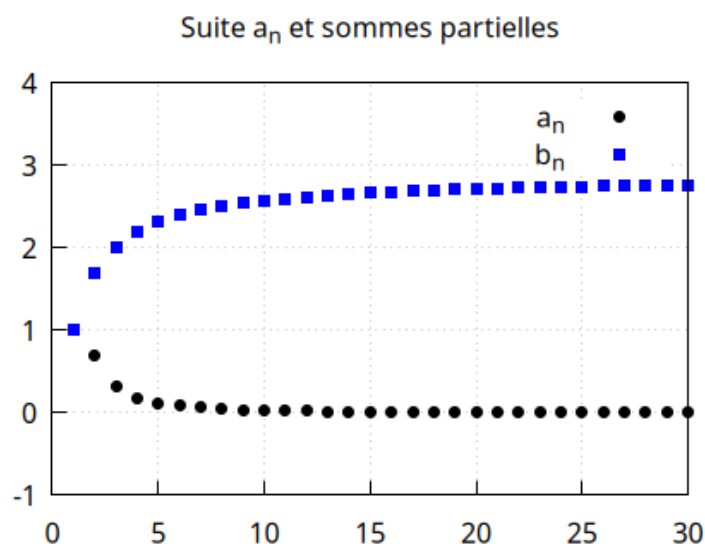
(% i2) $b(n) := \text{sum}(a(k), k, 1, n)$;

(%o2) $b(n) := \sum_{k=1}^n a(k)$

(% i4) POINTS1 :makelist([n,a(n)],n,1,30)\$

POINTS2 :makelist([n,b(n)],n,1,30)\$

(% i5) wxdraw2d(
 grid=true,
 xrange=[0,30],
 yrange=[-1,4],
 title= Suite a_n et sommes partielles ,
 key= a_n ,
 color=black,
 point_type=7,
 points(POINTS1),
 color=blue,
 point_type=5,
 key= b_n ,
 points(POINTS2)
)\$



(%t5)

(% i6) $\text{sum}(a(k), k, 1, \text{inf}), \text{simplsum}$;

(%o6) $\frac{\pi^2}{2} - 2 \text{zetaeta}(5)$

```
(% i7) float(%);
(%o7) 2.860946690257939
```

La limite de la série est donc d'environ 2,86.

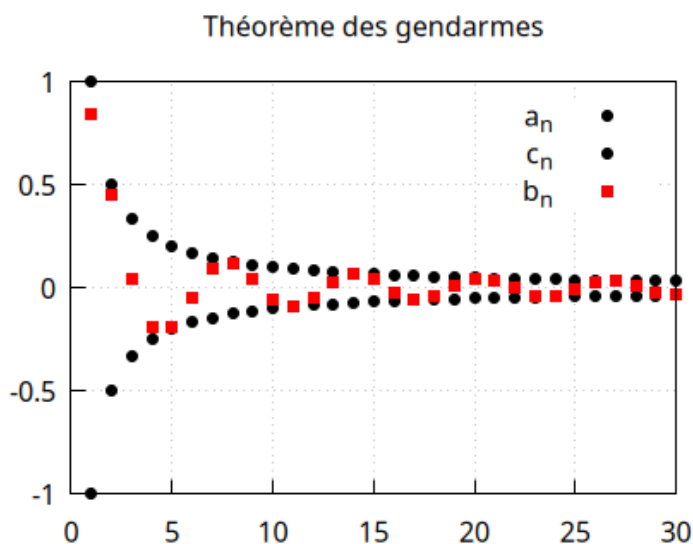
3 Exercice 3

```
(% i8) kill(all)$
```

```
(% i3) a(n) := -1/n $ b(n) := sin(n)/n $ c(n) := 1/n $
```

```
(% i6) POINTS1 :makelist([n,a(n)],n,1,30)$
POINTS2 :makelist([n,b(n)],n,1,30)$
POINTS3 :makelist([n,c(n)],n,1,30)$
```

```
(% i7) wxdraw2d(
  grid=true,
  xrange=[0,30],
  yrange=[-1,1],
  title= Théorème des gendarmes ,
  key= a_n ,
  color=black,
  point_type=7,
  points(POINTS1),
  key= c_n ,points(POINTS3),
  color=red,
  point_type=5,
  key= b_n ,
  points(POINTS2)
)$
```



```
(%t7)
```

```
(% i10) limit(a(n),n,inf);limit(b(n),n,inf);limit(c(n),n,inf);
```

```
(%o8) 0
```

```
(%o9) 0
```

```
(%o10) 0
```

Les limites des suites $(1/n)$ et $(-1/n)$ sont égales à 0, tout comme celle de $(\sin(n)/n)$

4 Exercice 4

```
(% i1) kill(all)$
```

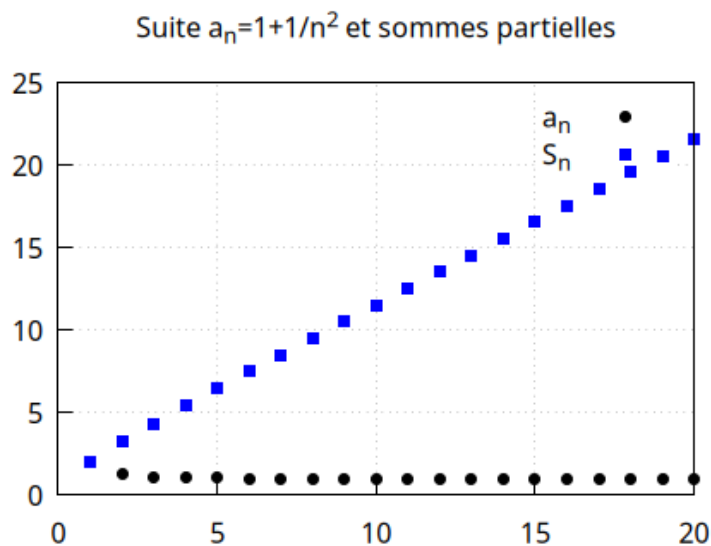
```
(% i2) a(n) := 1 + 1/n^2; S(n) := sum(a(k), k, 1, n);
```

```
(%o1)  $a(n) := 1 + \frac{1}{n^2}$ 
```

```
(%o2)  $S(n) := \sum_{k=1}^n a(k)$ 
```

```
(% i4) POINTS1 :makelist([n,a(n)],n,1,20)$  
POINTS2 :makelist([n,S(n)],n,1,20)$
```

```
(% i5) wxdraw2d(  
  grid=true,  
  xrange=[0,20],  
  yrange=[0,25],  
  title= Suite a_n=1+1/n^2 et sommes partielles ,  
  key= a_n ,  
  color=black,  
  point_type=7,  
  points(POINTS1),  
  color=blue,  
  point_type=5,  
  key= S_n ,  
  points(POINTS2)  
)$
```



```
(%t5)
```

Plus la valeur de n augmente, et plus la somme partielle augmente d'une unité car $a(n)$ se rapproche de 1, et donc la suite des sommes partielles diverge.

5 Exercice 5

```
(% i6) kill(all)$
```

```
(% i2) a(n) :=sqrt(n)/log(n);S(n) :=sum(a(k),k,2,n);
```

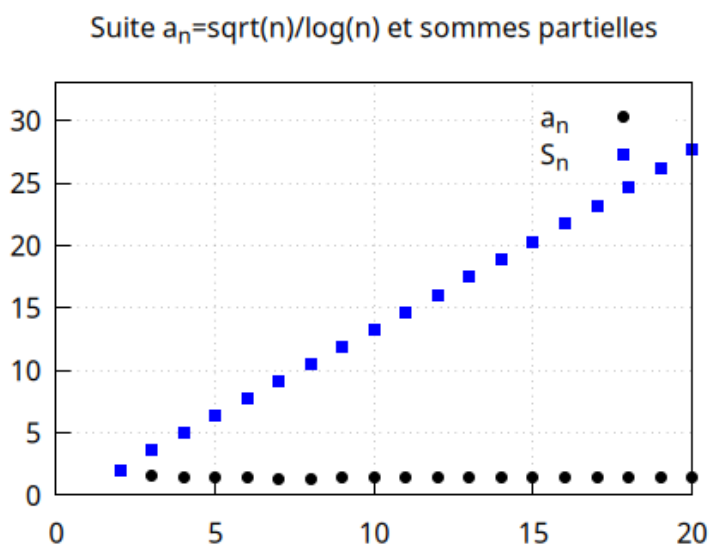
```
(%o1)  $a(n) := \frac{\sqrt{n}}{\log(n)}$ 
```

```
(%o2)  $S(n) := \sum_{k=2}^n a(k)$ 
```

```
(% i4) POINTS1 :makelist([n,a(n)],n,2,20)$
```

```
POINTS2 :makelist([n,S(n)],n,2,20)$
```

```
(% i5) wxdraw2d(
  grid=true,
  xrange=[0,20],
  yrange=[0,33],
  title= Suite a_n=sqrt(n)/log(n) et sommes partielles ,
  key= a_n ,
  color=black,
  point_type=7,
  points(POINTS1),
  color=blue,
  point_type=5,
  key= S_n ,
  points(POINTS2)
)$
```



```
(%t5)
```

```
(% i6) limit(a(n),n,inf);
```

```
(%o6)  $\infty$ 
```

La divergence de la suite $a(n)$ entraîne de manière immédiate la divergence de la série.

6 Exercice 6

```
(% i7) kill(all)$
```

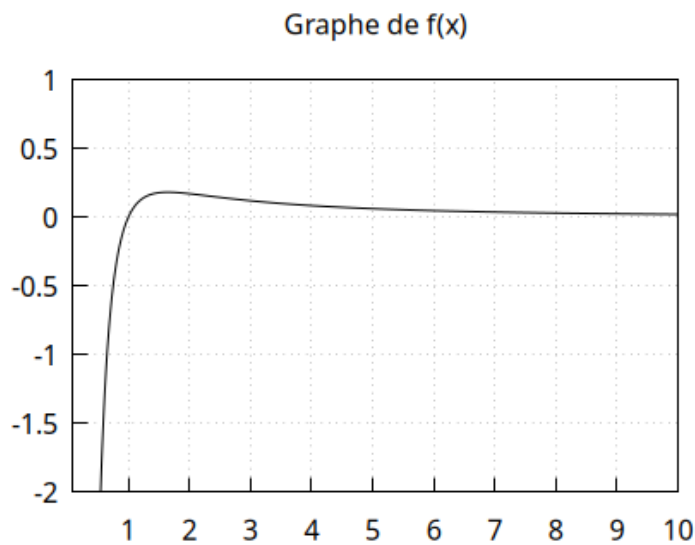
```
(% i1) f(x) :=log(x)/x^ 2;
```

```
(%o1)  $f(x) := \frac{\log(x)}{x^2}$ 
```

```
(% i2) define(fprime(x),diff(f(x),x));
```

```
(%o2)  $fprime(x) := \frac{1}{x^3} - \frac{2 \log(x)}{x^3}$ 
```

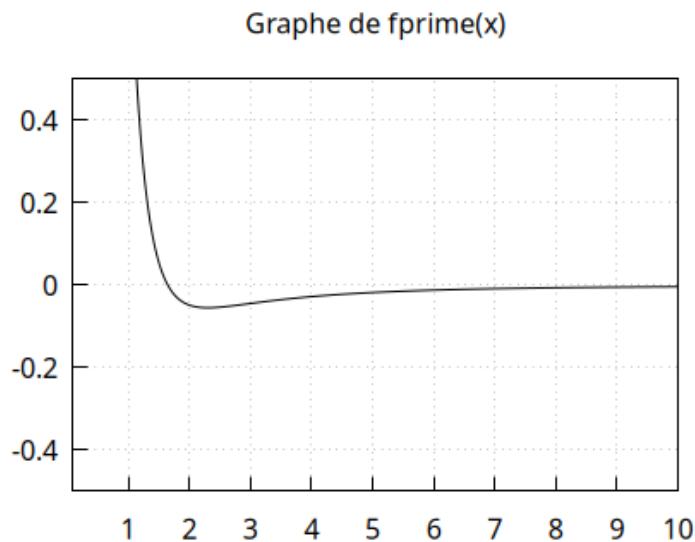
```
(% i3) wxdraw2d(  
  grid=true,  
  xrange=[0.1,10],  
  yrange=[-2,1],  
  title= Graphe de f(x) ,  
  color=black,  
  explicit(f(x),x,0.1,10))$
```



```
(%t3)
```

Il semble que la fonction f soit croissante jusqu'à une valeur de x d'environ 1.5, puis décroissante et positive. Vérifions le avec l'étude de sa dérivée :

```
(% i4) wxdraw2d(  
  grid=true,  
  xrange=[0.1,10],  
  yrange=[-0.5,0.5],  
  title= Graphe de fprime(x) ,  
  color=black,  
  explicit(fprime(x),x,0.1,10))$
```



(%t4)

fprime(x) est bien positive puis négative. Vérifions le par le calcul :

(% i5) `solve(fprime(x)=0,x);`

(%o5) $\left[x = \sqrt{\%e} \right]$

(% i6) `float(%);`

(%o6) $[x = 1.6487212707001282]$

fprime(x) est donc positive sur $]0, \sqrt{e}[$ et négative sur $]\sqrt{e}, +\infty[$.

(% i7) `f(sqrt(%e));`

(%o7) $\frac{\%e^{-1}}{2}$

(% i8) `float(%);`

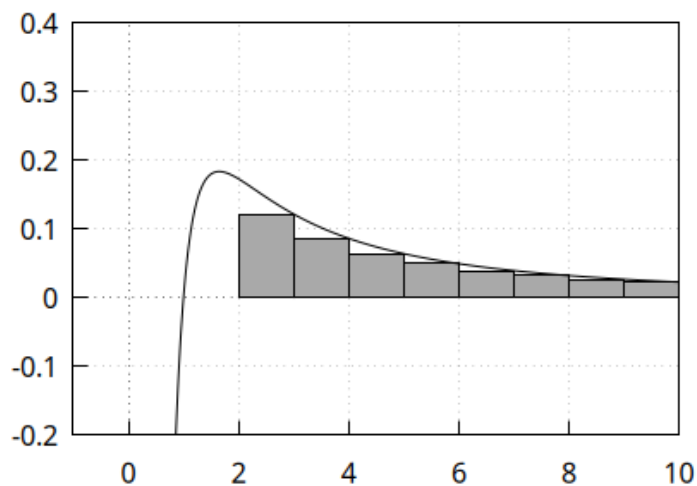
(%o8) 0.18393972058572117

On a donc prouvé que f est positive et décroissante sur $]\sqrt{e}, +\infty[$, donc notamment à partir de $n=2$ si l'on considère la suite associée.

(% i9) `RECTANGLES :makelist(rectangle([i-1,0],[i,f(i)]),i,3,10)$`

(% i10) `wxdraw2d(
grid=true,
xaxis=true,
yaxis=true,
xrange=[-1,10],
yrange=[-0.2,0.4],
title= Majoration par une intégrale de la série ,
border=true,
color=black,
fill_color=dark_grey,
RECTANGLES,
explicit(f(x),x,0.1,11)
)$`

Majoration par une intégrale de la série



(%t10)

(% i11) `S(n) :=sum(f(k),k,2,n);`

(%o11) $S(n) := \sum_{k=2}^n f(k)$

S(n) est donc majorée par l'intégrale de 2 à n de f(x)

(% i12) `integrate(f(x),x,2,inf);`

(%o12) $\frac{\log(2)}{2} + \frac{1}{2}$

Ce qui prouve la convergence de la série (on remarquera que le terme pour k=1 est nul, ce qui nous permet de ne commencer qu'à 2)

7 Exercice 7

(% i13) `kill(all)$`

(% i1) `f(x) :=(x+3)/(x^3+x);`

(%o1) $f(x) := \frac{x+3}{x^3+x}$

Valeur approchée de la somme

(% i2) `Sn:=sum(float(f(k)),k,1,10000);`

(Sn) 3.0921719940422783

(% i3) `bsup :float(integrate(f(x),x,10000,inf));`

(bsup) $1.000149996599475310^{-4}$

(% i4) `binf :float(integrate(f(x),x,10001,inf));`

(binf) $1.000049976624950410^{-4}$

donc $\text{binf} < S - S_n < \text{bsup}$, ce qui équivaut à $\text{binf} + S_n < S < \text{bsup} + S_n$

(% i5) `print(bin+Sn, <S<,bsup+Sn)$`

3.0922719990399408 < S < 3.092272009041938

8 Exercice 8

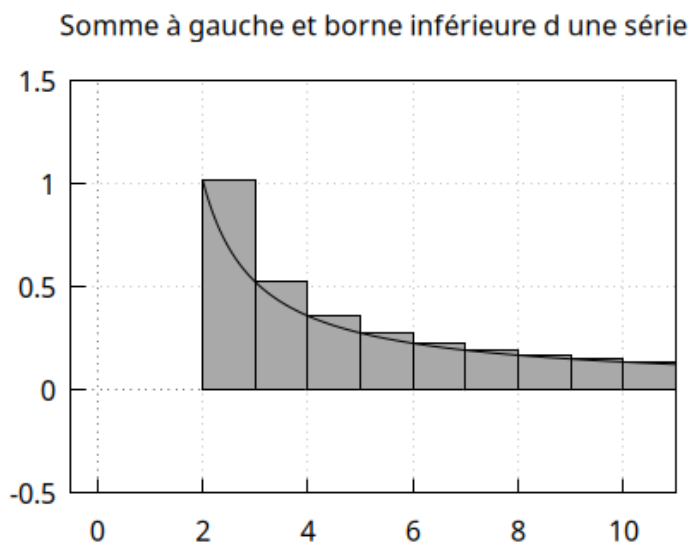
```
(% i6) kill(all)$
```

```
(% i1) f(x) := 1/(sqrt(x)*log(x));
```

```
(%o1)  $f(x) := \frac{1}{\sqrt{x} \log(x)}$ 
```

```
(% i2) RECTANGLES : (makelist(rectangle([i,0],[i+1,f(i)]),i,2,10))$
```

```
(% i3) wxdraw2d(
  grid=true,
  xaxis=true,
  yaxis=true,
  xrange=[-0.5,11],
  yrange=[-0.5,1.5],
  title= Somme à gauche et borne inférieure d une série ,
  color=black,
  border=true,
  fill_color=dark_grey,
  RECTANGLES,
  explicit(f(x),x,2,11)
)$
```



```
(%t3)
```

La série est minorée par l'intégrale de f entre 2 et l'infini

```
-> /*integrate(f(x),x,2,inf);
    defint : integral is divergent */;
```

L'intégrale diverge donc la série est divergente.

9 Exercice 9

```
(% i4) kill(all)$
```

Pour les grandes valeurs de n, le terme général se rapproche de $\sqrt{3n^2}/n^3$ soit $\sqrt{3}/n^2$. On va donc comparer à la série $1/n^2$ (série de Riemann classique).

```
(% i6) /* Définition du terme général et de la série de comparaison */
a_n : sqrt(3*n^2-1)/(n^3+5*n);
b_n : 1/n^2;

/* Calcul de la limite du rapport a_n/b_n */
L : limit(a_n/b_n, n, inf);
print( Limite du rapport a_n/b_n = , L);

/* Vérification de la convergence de la série de comparaison */
print( La série sum(1/n^2) converge car p = 2 > 1 (série de Riemann) );

/* Conclusion du critère de comparaison par passage à la limite */
if L > 0 and L < inf then
print( Comme la limite est finie et positive, sum(a_n) converge aussi )
else
print( Le critère ne permet pas de conclure );
```

$$(a_n) \quad \frac{\sqrt{3n^2-1}}{n^3+5n}$$

$$(b_n) \quad \frac{1}{n^2}$$

$$(L) \quad \sqrt{3}$$

Limite du rapport $a_n/b_n = \sqrt{3}$

```
(%o4)  \sqrt{3}
La série sum(1/n^2) converge car p = 2 > 1 (série de Riemann)
(%o5)  La série sum(1/n^2) converge car p = 2 > 1 (série de Riemann)
```

Comme la limite est finie et positive, $\sum(a_n)$ converge aussi

```
(%o6)  Comme la limite est finie et positive, sum(a_n) converge aussi
```

10 Exercice 10

```
(% i7)  kill(all)$

(% i10) /* Serie 1 : critere du rapport */
a_n : n/2^n;
ratio : ratsimp(subst(n+1, n, a_n)/a_n);
L1 : limit(ratio, n, inf);
print( Limite du critere du rapport = , L1);
if L1 < 1 then
print( Comme L1 < 1, la serie sum(n/2^n) converge );

/* Serie 2 : critere de comparaison par passage a la limite */
a2_n : n/(2^n-1);
b2_n : n/2^n;
L2 : limit(a2_n/b2_n, n, inf);
print( Limite du rapport a2_n/b2_n = , L2);
if L2 > 0 and L2 < inf then
(print( Comme la limite est finie et positive, et sum(b2_n) converge, ),
print( alors sum(a2_n) = sum(n/(2^n-1)) converge aussi ));
```

$$(a_n) \quad \frac{n}{2^n}$$

(ratio) $\frac{n+1}{2n}$

(L1) $\frac{1}{2}$

Limite du critere du rapport = $\frac{1}{2}$

(%o4) $\frac{1}{2}$

Comme $L1 < 1$, la serie $\sum(n/2^n)$ converge

(%o5) Comme $L1 < 1$, la serie $\sum(n/2^n)$ converge

(a2_n) $\frac{n}{2^n - 1}$

(b2_n) $\frac{n}{2^n}$

(L2) 1

Limite du rapport $a2_n/b2_n = 1$

(%o9) 1

Comme la limite est finie et positive, et $\sum(b2_n)$ converge,
alors $\sum(a2_n) = \sum(n/(2^n-1))$ converge aussi

(%o10) alors $\sum(a2_n) = \sum(n/(2^n-1))$ converge aussi

11 Exercice 11

(% i1) kill(all)\$

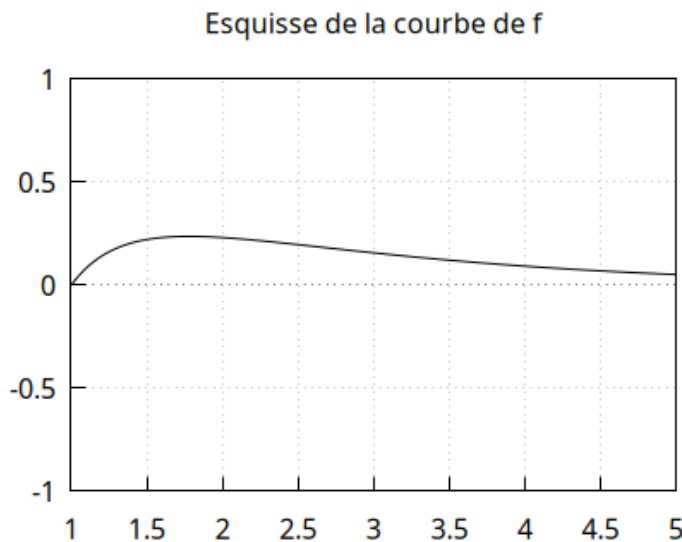
Critère de l'intégrale : on, définit la fonction f associée. Est-elle positive, décroissante sur $[1, +\infty[$ et son intégrale converge-t-elle ?

(% i1) f(x) := log(x)/(2^x - 1);

(%o1) $f(x) := \frac{\log(x)}{2^x - 1}$

La fonction f semble décroissante à partir d'une certaine valeur.

(% i2) wxdraw2d(
grid=true,
xaxis=true,
yaxis=true,
xrange=[1,5],
yrange=[-1,1],
title= Esquisse de la courbe de f ,
color=black,
explicit(float(f(x)),x,1,5)
)\$



(%t2)

Pour le prouver on calcule la dérivée, mais l'étude du signe est trop compliquée.

(% i3) `diff(f(x),x);`

(%o3)
$$\frac{1}{(2^x - 1)x} - \frac{2^x \log(2) \log(x)}{(2^x - 1)^2}$$

(% i4) `integrate(f(x),x,1,inf);`

(%o4)
$$\int_1^{\infty} \frac{\log(x)}{2^x - 1} dx$$

L'intégrale de f n'est pas non plus calculable, donc on ne peut appliquer le critère de d'Alembert. Calculons donc les sommes partielles :

(% i5) `S(p) := sum(float(f(k)),k,1,p);`

(%o5)
$$S(p) := \sum_{k=1}^p \text{float}(f(k))$$

(% i6) `for i : 100 thru 2000 step 100 do print( S( ,i, ) = ,S(i))$`

S(100)= 0.5932202191397986
 S(200)= 0.5932202191397986
 S(300)= 0.5932202191397986
 S(400)= 0.5932202191397986
 S(500)= 0.5932202191397986
 S(600)= 0.5932202191397986
 S(700)= 0.5932202191397986
 S(800)= 0.5932202191397986
 S(900)= 0.5932202191397986
 S(1000)= 0.5932202191397986
 S(1100)= 0.5932202191397986
 S(1200)= 0.5932202191397986
 S(1300)= 0.5932202191397986
 S(1400)= 0.5932202191397986
 S(1500)= 0.5932202191397986
 S(1600)= 0.5932202191397986
 S(1700)= 0.5932202191397986
 S(1800)= 0.5932202191397986
 S(1900)= 0.5932202191397986

$S(2000) = 0.5932202191397986$

La série semble converger vers une valeur proche de 0.593

```
(% i7)  g(x) := log(x)-x ;  
(%o7)  g(x) := log(x) - x
```

```
(% i8)  define(gprime(x),factor(diff(g(x),x)));
```

```
(%o8)  gprime(x) := - (x-1/x)
```

Sur $[1, +\infty[$, $g'(x)$ est négatif donc g est décroissante. Comme $g(1) = -1$, on en déduit que $g(x)$ est négatif et donc que $\log(x) < x$. On en déduit donc la majoration $f(n) < n/(2^n - 1) < n/2^{n-1}$. On étudie donc la série $n/2^{n-1}$:

```
(% i9)  u(n) := n/2^(n-1);
```

```
(%o9)  u(n) := n/2^(n-1)
```

```
(% i10) limit(u(n+1)/u(n),n,inf);
```

```
(%o10) 1/2
```

$u(n)$ converge donc le critère de comparaison prouve la convergence de la série $\sum(f(n))$

12 Exercice 12

```
(% i11) kill(all)$
```

On définit la série des valeurs absolues des termes :

```
(% i1)  u(n) := n^10 / %e^n ;
```

```
(%o1)  u(n) := n^10 / %e^n
```

```
(% i2)  l : limit(u(n+1)/u(n),n,inf);
```

```
(l)  %e^-1
```

```
(% i3)  float(l-1);
```

```
(%o3)  -0.6321205588285577
```

Le rapport $u(n+1)/u(n)$ est inférieur à 1 pour cette série des valeurs absolues, ce qui prouve la convergence absolue de la série.

13 Exercice 13

```
(% i4)  kill(all)$
```

On définit le terme général de la série des valeurs absolues pour étudier la convergence absolue :

```
(% i1)  a(n) := %pi/2 - atan(n);
```

```
(%o1)  a(n) := pi/2 - atan(n)
```

(% i2) assume(n>0);

(%o2) [n > 0]

On démontre l'égalité classique $\pi/2 - \text{atan}(n) = \text{atan}(1/n)$

(% i7) h(x) := %pi/2 - atan(x) - atan(1/x);

dh : diff(h(x), x);

dh : ratsimp(dh);

dh;

h(1);

(%o3) $h(x) := \frac{\pi}{2} - \text{atan}(x) - \text{atan}\left(\frac{1}{x}\right)$

(dh) $\frac{1}{\left(\frac{1}{x^2} + 1\right)x^2} - \frac{1}{x^2 + 1}$

(dh) 0

(%o6) 0

(%o7) 0

La fonction h est à dérivée nulle et prend la valeur 0 en 1, donc il s'agit de la fonction nulle, ce qui prouve l'égalité attendue. La série à étudier se simplifie donc

(% i8) b(n) := atan(1/n);

(%o8) $b(n) := \text{atan}\left(\frac{1}{n}\right)$

Etude de la convergence absolue : En $+\infty$, b(n) équivaut à $1/n$ qui est une série divergente, donc la série n'est pas absolument convergente. On étudie maintenant la série $(-1)^n b(n)$ avec le critère des séries alternées : b(n) positif et décroissant.

(% i9) limit(b(n),n,inf);

(%o9) 0

D'après ce critère des séries alternées, la série est donc convergente et non absolument convergente.

14 Exercice 14

(% i10) kill(all)\$

(% i1) taylor(log(x), x, 1, 10);

(%o1/T/) $x - 1 - \frac{(x-1)^2}{2} + \frac{(x-1)^3}{3} - \frac{(x-1)^4}{4} + \frac{(x-1)^5}{5} - \frac{(x-1)^6}{6} + \frac{(x-1)^7}{7} - \frac{(x-1)^8}{8} + \frac{(x-1)^9}{9} - \frac{(x-1)^{10}}{10}$

Formule explicite du ième terme

(% i2) a(n) := (-1)^(n-1) * (x-1)^n / n;

(%o2) $a(n) := \frac{(-1)^{n-1} (x-1)^n}{n}$

On applique le critère de d'Alembert pour cette série alternée :

(% i3) limit(abs(a(n+1)/a(n)),n,inf);

(%o3) |x - 1|

La série converge pour $\text{abs}(x-1) < 1$, soit $0 < x < 2$. Etudions les cas limites :

(% i4) subst(0,x,a(n));

(%o4) $\frac{(-1)^{2n-1}}{2n}$

Comme $(-1)^{2n} = 1$, la série a pour terme général $-1/n$ qui diverge

```
(% i5) subst(2,x,a(n));
```

```
(%o5) 
$$\frac{(-1)^{n-1}}{n}$$

```

On reconnaît la série harmonique alternée qui converge, et qui est bien égale à $\log(2)$

```
(% i6) sum( subst(2,x,a(n)) ,n,1,inf),simpsum;
```

```
(%o6) 
$$\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}$$

```

On remarque que Maxima ne sait pas calculer directement la somme de cette série.

15 Exercice 15

```
(% i7) kill(all)$
```

```
(% i1) taylor((1+x)^ n, x, 0, 10);
```

```
(%o1/T/) 
$$1 + nx + \frac{(n^2 - n) x^2}{2} + \frac{(n^3 - 3n^2 + 2n) x^3}{6} + \frac{(n^4 - 6n^3 + 11n^2 - 6n) x^4}{24} + \frac{(n^5 - 10n^4 + 35n^3 - 50n^2 + 24n) x^5}{120} + \dots$$

```

Approximation linéaire (deux premiers termes obtenus précédemment)

```
(% i2) L(x) := 1 + n*x;
```

```
(%o2) 
$$L(x) := 1 + nx$$

```

La fonction étudiée et l'erreur commise

```
(% i4) f(x) := (1+x)^ n;E(x) := f(x) - L(x);
```

```
(%o3) 
$$f(x) := (1+x)^n$$

```

```
(%o4) 
$$E(x) := f(x) - L(x)$$

```

Développement de l'erreur par Taylor

```
(% i5) taylor(E(x), x, 0, 5);
```

```
(%o5/T/) 
$$\frac{(n^2 - n) x^2}{2} + \frac{(n^3 - 3n^2 + 2n) x^3}{6} + \frac{(n^4 - 6n^3 + 11n^2 - 6n) x^4}{24} + \frac{(n^5 - 10n^4 + 35n^3 - 50n^2 + 24n) x^5}{120} + \dots$$

```

Etude de l'erreur en 0.001 pour les premières valeurs de n

```
(% i6) for k : 1 thru 10 do (  
  print( n = , k , , Erreur E(0.001) = ,  
  float(subst([x=0.001, n=k], E(x)))  
  )$
```

n = 1 , Erreur E(0.001) = $-1.101549407245272510^{-16}$

n = 2 , Erreur E(0.001) = 9.9999999697423510^{-7}

n = 3 , Erreur E(0.001) = 3.00099999727520210^{-6}

n = 4 , Erreur E(0.001) = $6.004000999485642510^{-6}$

n = 5 , Erreur E(0.001) = $1.001000500034255510^{-5}$

n = 6 , Erreur E(0.001) = $1.502001500531223410^{-5}$

n = 7 , Erreur E(0.001) = $2.103503502016209510^{-5}$

n = 8 , Erreur E(0.001) = 2.80560700551893410^{-5}

n = 9 , Erreur E(0.001) = 3.60841261249980610^{-5}

n = 10 , Erreur E(0.001) = $4.512021025116809410^{-5}$

Etude de l'erreur en prenant comme approximation l'erreur quadratique

```
(% i8) approx_err(n, x) := n*(n-1)/2 * x^2;
```

```
for k : 1 thru 10 do (
  print( n = , k,
    , Erreur exacte = , float(subst([x=0.001, n=k], E(x))),
    , Approx n(n-1)/2*x^2 = , float(subst([x=0.001, n=k], approx_err(n,x)))
  )$
```

```
(%o7) approx_err(n, x) :=  $\frac{n(n-1)}{2}x^2$ 
```

```
n = 1 , Erreur exacte = - 1.101549407245272510-16 , Approx n(n-1)/2· x^2 = 0.0
n = 2 , Erreur exacte = 9.99999999697423510-7 , Approx n(n-1)/2· x^2 = 1.010-6
n = 3 , Erreur exacte = 3.00099999727520210-6 , Approx n(n-1)/2· x^2 = 3.010-6
n = 4 , Erreur exacte = 6.004000999485642510-6 , Approx n(n-1)/2· x^2 = 6.010-6
n = 5 , Erreur exacte = 1.001000500034255510-5 , Approx n(n-1)/2· x^2 = 9.999999999999910-6
n = 6 , Erreur exacte = 1.502001500531223410-5 , Approx n(n-1)/2· x^2 = 1.4999999999999910-5
n = 7 , Erreur exacte = 2.103503502016209510-5 , Approx n(n-1)/2· x^2 = 2.10000000000000210-5
n = 8 , Erreur exacte = 2.80560700551893410-5 , Approx n(n-1)/2· x^2 = 2.810-5
n = 9 , Erreur exacte = 3.60841261249980610-5 , Approx n(n-1)/2· x^2 = 3.610-5
n = 10 , Erreur exacte = 4.512021025116809410-5 , Approx n(n-1)/2· x^2 = 4.510-5
```

16 Exercice 16

```
(% i9) kill(all)$
```

```
(% i1) taylor(exp(%i*x),x,0,8);
```

```
(%o1/T/)  $1 + ix - \frac{x^2}{2} - \frac{ix^3}{6} + \frac{x^4}{24} + \frac{ix^5}{120} - \frac{x^6}{720} - \frac{ix^7}{5040} + \frac{x^8}{40320} + \dots$ 
```

On isole la série réelle (après avoir reconnu la série de Taylor de cosinus en 0)

```
(% i2) taylor(cos(x),x,0,8);
```

```
(%o2/T/)  $1 - \frac{x^2}{2} + \frac{x^4}{24} - \frac{x^6}{720} + \frac{x^8}{40320} + \dots$ 
```

On isole la série des parties imaginaires (après avoir reconnu la série de Taylor de sinus en 0)

```
(% i3) taylor(sin(x),x,0,8);
```

```
(%o3/T/)  $x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{5040} + \dots$ 
```

```
(% i4) rectform(exp(%i*x));
```

```
(%o4) %i sin(x) + cos(x)
```

17 Exercice 17

```
(% i5) taylor(cosh(x),x,0,12);
```

```
(%o5/T/)  $1 + \frac{x^2}{2} + \frac{x^4}{24} + \frac{x^6}{720} + \frac{x^8}{40320} + \frac{x^{10}}{3628800} + \frac{x^{12}}{479001600} + \dots$ 
```

```
(% i6)  taylor(cos(%i*x),x,0,12);
(%o6/T/) 1 +  $\frac{x^2}{2}$  +  $\frac{x^4}{24}$  +  $\frac{x^6}{720}$  +  $\frac{x^8}{40320}$  +  $\frac{x^{10}}{3628800}$  +  $\frac{x^{12}}{479001600}$  + ...
```

18 Exercice 18

```
(% i7)  f(x):=1/(1+x^2);
(%o7)  f(x):=  $\frac{1}{1+x^2}$ 
```

```
(% i8)  taylor(f(x),x,0,15);
(%o8/T/) 1 - x^2 + x^4 - x^6 + x^8 - x^{10} + x^{12} - x^{14} + ...
```

```
(% i9)  sum((-1)^ (i)*x^ (2*i),i,0,7);
(%o9)  - x^{14} + x^{12} - x^{10} + x^8 - x^6 + x^4 - x^2 + 1
```

On peut donc écrire l'égalité

```
(% i10) eq :f(x)=sum((-1)^ (i)*x^ (2*i),i,0,inf);
(eq)   $\frac{1}{x^2+1} = \sum_{i=0}^{\infty} (-1)^i x^{2i}$ 
```

```
(% i11) eq2 :integrate(eq,x);
(eq2)  atan(x) =  $\left( \sum_{i=0}^{\infty} \frac{(-1)^i x^{2i+1}}{2i+1} \right) + \%c1$ 
```

Recherchons la valeur de la constante avec x=0

```
(% i12) solve(subst(0,x,lhs(eq2))=subst(0,x,rhs(eq2)),%c1);
(%o12)  [%c1 = 0]
```

Remplaçons %c1 par 0 dans eq2

```
(% i13) eq3 :subst(0,%c1,eq2);
(eq3)  atan(x) =  $\sum_{i=0}^{\infty} \frac{(-1)^i x^{2i+1}}{2i+1}$ 
```

On calcule l'expression précédente pour x=1

```
(% i14) eq4 :subst(1,x,lhs(eq3))=subst(1,x,rhs(eq3));
(eq4)   $\frac{\pi}{4} = \sum_{i=0}^{\infty} \frac{(-1)^i}{2i+1}$ 
```

```
(% i15) eq5 :4*eq4;
(eq5)   $\pi = 4 \sum_{i=0}^{\infty} \frac{(-1)^i}{2i+1}$ 
```

Nous avons donc trouvé l'expression d'une série infinie donnant π . On calcule la somme partielle à 100

```
(% i16) float(4*sum((-1)^ i/(2*i+1),i,0,100));
(%o16)  3.1514934010709905
```

Erreur commise par cette approximation

```
(% i17) float(%pi-4*sum((-1)^ i/(2*i+1),i,0,100));
(%o17) - 0.009900747481197403
L'erreur est de l'ordre d  $10^{-2}$ 
```

19 Exercice 19

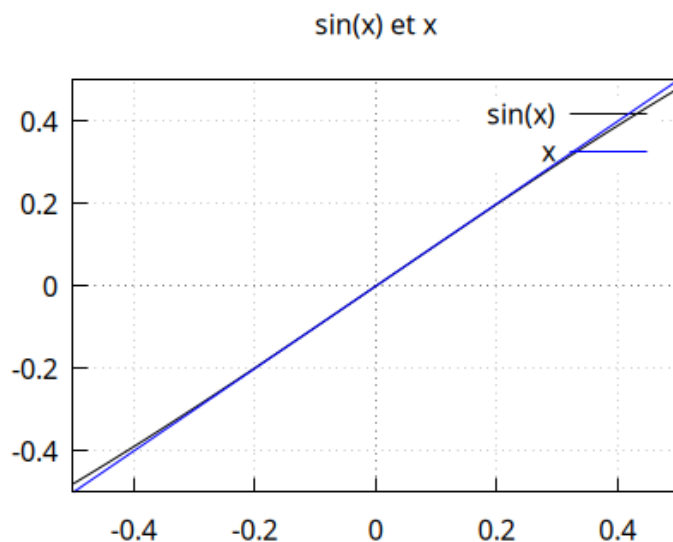
```
(% i18) kill(all)$
```

```
(% i1)  Taylor(sin(theta),theta,0,3);
```

```
(%o1/T/)  $\theta - \frac{\theta^3}{6} + \dots$ 
```

Une approximation linéaire de $\sin(\theta)$ est donc θ au voisinage de 0.

```
(% i2)  wxdraw2d(
title= sin(x) et x ,
grid=true,
xaxis=true,
yaxis=true,
xrange=[-0.5,0.5],
yrange=[-0.5,0.5],
color=black,
line_width=1,
key= sin(x) ,
explicit(sin(x),x,-0.5,0.5),
color=blue,
line_width=1,
key= x ,
explicit(x,x,-0.5,0.5)
)$
```



```
(%t2)
```

Sur cet intervalle, on constate que l'approximation est de qualité, les deux courbes étant très proches.

```
(% i3)  erreur(x) :=float(abs(sin(x)-x)/sin(x));
```

```
(%o3)  erreur(x) := float  $\left( \frac{|\sin(x) - x|}{\sin(x)} \right)$ 
```

```
(% i4) for i :0.1 thru 1 step 0.1 do print( pour x= ,i, l'erreur vaut ,erreur(i))$
```

pour x= 0.1 l'erreur vaut 0.001668613163477696

pour x= 0.2 l'erreur vaut 0.006697909534468891

pour x= 0.30000000000000004 l'erreur vaut 0.015159008547236834

pour x= 0.4 l'erreur vaut 0.027172982219113205

pour x= 0.5 l'erreur vaut 0.04291482146674408

pour x= 0.6 l'erreur vaut 0.06261931801263516

pour x= 0.7 l'erreur vaut 0.08658922886997274

pour x= 0.7999999999999999 l'erreur vaut 0.11520625551090898

pour x= 0.8999999999999999 l'erreur vaut 0.14894559211300595

pour x= 0.9999999999999999 l'erreur vaut 0.18839510577812124

On voit que pour $x > 0.6$ l'erreur est supérieure à 5%. On peut en chercher une valeur plus précise :

```
(% i5) find_root(erreur(x)=0.05,x,0.1,1);
```

```
(%o5) 0.538411672338011
```

Donc $x=0.54$ et l'intervalle cherché est de $[-0.54, 0.54]$ soit en degrés

```
(% i6) float(180*0.54/%pi);
```

```
(%o6) 30.939720937064457
```

Résolution de l'équation différentielle :

```
(% i8) assume(L>0)$assume(g>0)$
```

```
(% i9) ode2('diff(y(x),x,2)=-g/L*y(x),y(x),x);
```

```
(%o9)  $y(x) = \%k1 \sin\left(\frac{\sqrt{g}x}{\sqrt{L}}\right) + \%k2 \cos\left(\frac{\sqrt{g}x}{\sqrt{L}}\right)$ 
```

La période est donc de $2\pi \sqrt{L/g}$

20 Exercice 20

```
(% i10) kill(all)$
```

```
(% i6) TIMEDEP(x,t,n) :=sin(n*%pi*x)*cos(n*%pi*t)$
```

```
TIMEDEP3 :TIMEDEP(x,t,3)$
```

```
load(draw)$
```

```
wxanimate_autoplay :true;
```

```
wxanimate_framerate :20;
```

```
with_slider_draw(
```

```
i, makelist(0.05*i, i, 0, 200),
```

```
title=concat( t= ,0.05*i),
```

```
grid=true,
```

```
xaxis=true,
```

```
yaxis=true,
```

```
nticks=120,
```

```
xrange=[0,1],
```

```
yrange=[-1,1],
```

```
color=black,
```

```
line_width=2,
```

```
explicit(subst(0.05*i,t,TIMEDEP3),x,0,1)
```

```
)$
```

```
(wxanimate_autoplay) true
```

```
(wxanimate_framerate) 20
```

(%t6) [animated graphics - not shown in TeX export]

(%t7)

```
(% i12) TIMEDEP(x,t,n) :=sin(n*%pi*x)*cos(n*%pi*t)$  
TIMEDEP5 :TIMEDEP(x,t,5)$  
wxanimate_autoplay :true;  
wxanimate_framerate :20;  
with_slider_draw(  
i, makelist(0.05*i, i, 0, 200),  
title=concat( t= ,0.05*i),  
grid=true,  
xaxis=true,  
yaxis=true,  
nticks=120,  
xrange=[0,1],  
yrange=[-1,1],  
color=black,  
line_width=2,  
explicit(subst(0.05*i,t,TIMEDEP5),x,0,1)  
)$
```

(wxanimate_autoplay) true

(wxanimate_framerate) 20

(%t12) [animated graphics - not shown in TeX export]

(%t13)

```
(% i20) TIMEDEP(x,t,n) :=sin(n*%pi*x)*cos(n*%pi*t)$  
TIMEDEP3 :TIMEDEP(x,t,3)$  
TIMEDEP5 :TIMEDEP(x,t,5)$  
COMBO :TIMEDEP3 + (1/3)*TIMEDEP5$  
wxanimate_autoplay :true;  
wxanimate_framerate :20;  
with_slider_draw(  
i, makelist(0.05*i, i, 0, 200),  
title=concat( t= ,0.05*i),  
grid=true,  
xaxis=true,  
yaxis=true,  
nticks=120,  
xrange=[0,1],  
yrange=[-1.4,1.4],  
color=black,  
line_width=2,  
explicit(subst(0.05*i,t,COMBO),x,0,1)  
)$
```

(wxanimate_autoplay) true

(wxanimate_framerate) 20

(%t20) [animated graphics - not shown in TeX export]

(%t21)

21 Exercice 21

(% i22) kill(all)\$

Puisque $f(x)$ est nulle sur $[1,2]$, il suffit de scinder l'intégrale en deux morceaux, sur $[0,0.5]$ et $[0.5,1]$:

```
(% i5)  f1 : x;
        f2 : 1-x;
        L : 2;
        bn : (2/L)*(integrate(f1*sin(n*%pi*x/L), x, 0, 1/2)
        + integrate(f2*sin(n*%pi*x/L), x, 1/2, 1));
        bn : trigsimp(bn);

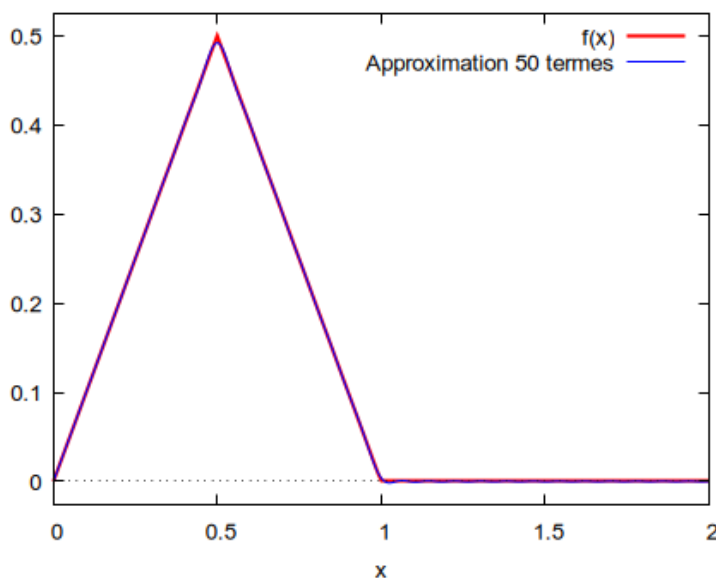
(f1)  x
(f2)  1 - x
(L)  2
(bn)  - (4 sin(πn/2) / (π²n²)) + (4 sin(πn/4) + πn cos(πn/4) / π²n²) + (4 sin(πn/4) - πn cos(πn/4) / π²n²)
(bn)  - (4 sin(πn/2) - 8 sin(πn/4) / π²n²)
```

On construit ensuite l'approximation et le graphe demandé

```
(% i8)  fapprox(x) := sum(bn*sin(n*%pi*x/L), n, 1, 50);
        f(x) := if x <= 0.5 then x
        elseif x <= 1 then 1-x
        else 0;
        wxplot2d([f(x), fapprox(x)], [x, 0, 2],
        [legend, f(x), Approximation 50 termes ],
        [style, [lines, 2, red], [lines, 1, blue]]);

(%o6)  fapprox(x) := ∑n=150 bn sin( nπx / L )

(%o7)  f(x) := if x <= 0.5 then x elseif x <= 1 then 1 - x else 0
```



(%t8)

(%o8)

On constate que l'approximation est quasiment égale à la courbe de f . Pour la suite, on additionne les 50 modes avec leur dépendance temporelle

```
(% i14) v : 5;
ytotal(x,t) := sum(bn*sin(n*%pi*x/L)*cos(n*%pi*v*t/L), n, 1, 50);
```

```
load(draw)$
wxanimate_autoplay : true;
wxanimate_framerate : 20;

with_slider_draw(
i, makelist(0.2*i, i, 0, 100),
title = concat( t= , 0.1*i),
grid = true,
xaxis = true,
yaxis = true,
nticks = 150,
xrange = [0, 2],
yrange = [-0.3, 0.3],
color = black,
line_width = 2,
explicit(ytotal(x, 0.1*i), x, 0, 2)
)$
```

(v) 5

```
(%o10)  $ytotal(x, t) := \sum_{n=1}^{50} b_n \sin\left(\frac{n\pi x}{L}\right) \cos\left(\frac{n\pi vt}{L}\right)$ 
```

```
(wxanimate_autoplay) true
```

```
(wxanimate_framerate) 20
```

```
(%t14) [animated graphics - not shown in TeX export]
```

```
(%t15)
```

Animation avec amortissement :

```
(% i17) ytotal_damped(x,t) := sum(bn*exp(-0.1*n*t)*sin(n*%pi*x/L)*cos(n*%pi*v*t/L), n, 1, 50);
```

```
with_slider_draw(
i, makelist(0.2*i, i, 0, 100),
title = concat( t= , 0.02*i),
grid = true,
xaxis = true,
yaxis = true,
nticks = 150,
xrange = [0, 2],
yrange = [-0.3, 0.3],
color = black,
line_width = 2,
explicit(ytotal_damped(x, 0.02*i), x, 0, 2)
)$
```

```
(%o16)  $ytotal\_damped(x, t) := \sum_{n=1}^{50} b_n \exp(-0.1nt) \sin\left(\frac{n\pi x}{L}\right) \cos\left(\frac{n\pi vt}{L}\right)$ 
```

```
(%t17) [animated graphics - not shown in TeX export]
```

```
(%t18)
```