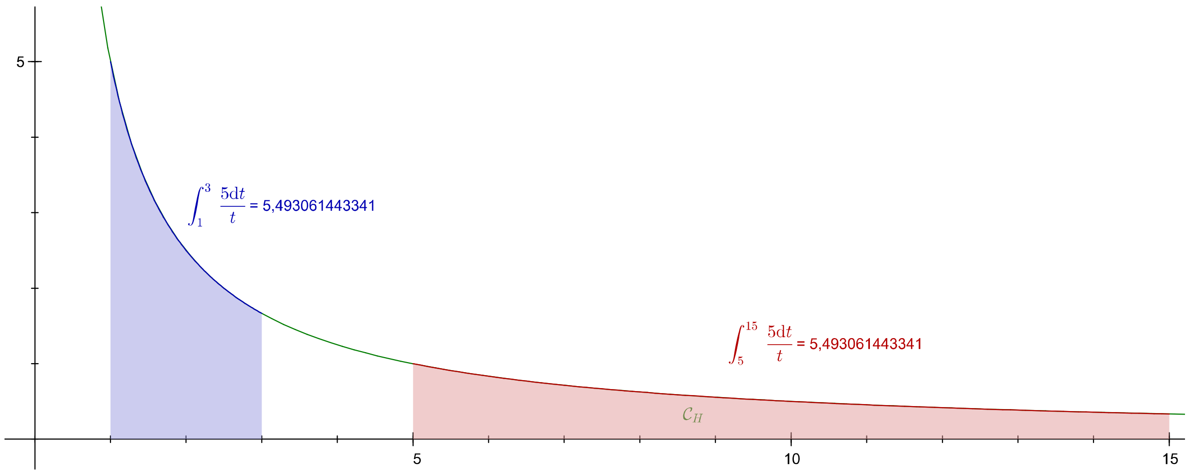
Logarithme népérien

I/ Quadrature de l'hyperbole

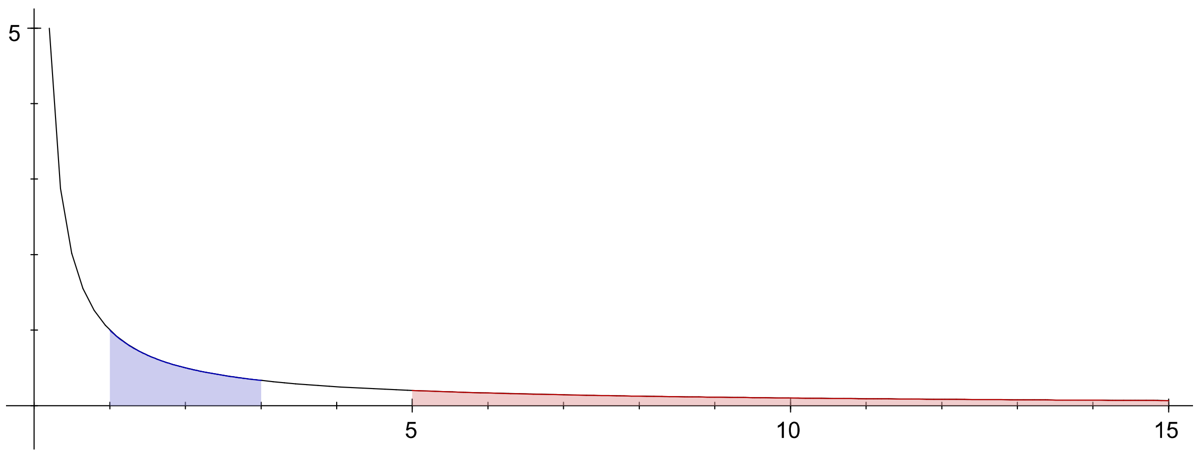
- En 1614, John Neper publie *Mirifici logarithmorum canonis descriptio* dans lequel il décrit les logarithmes (fonctions transformant les multiplications en additions).
- En 1616, Henry Briggs et John Neper commencent à publier les premières tables de logarithmes décimaux.
- En 1647, Grégoire de Saint Vincent (père jésuite) publie *Opus geometricum quadraturae circuli et sectionum coni decem libris comprehensum* dans lequel il constate le phénomène suivant :

1) Égalité d'aires

Les deux aires ci-dessous sont égales :

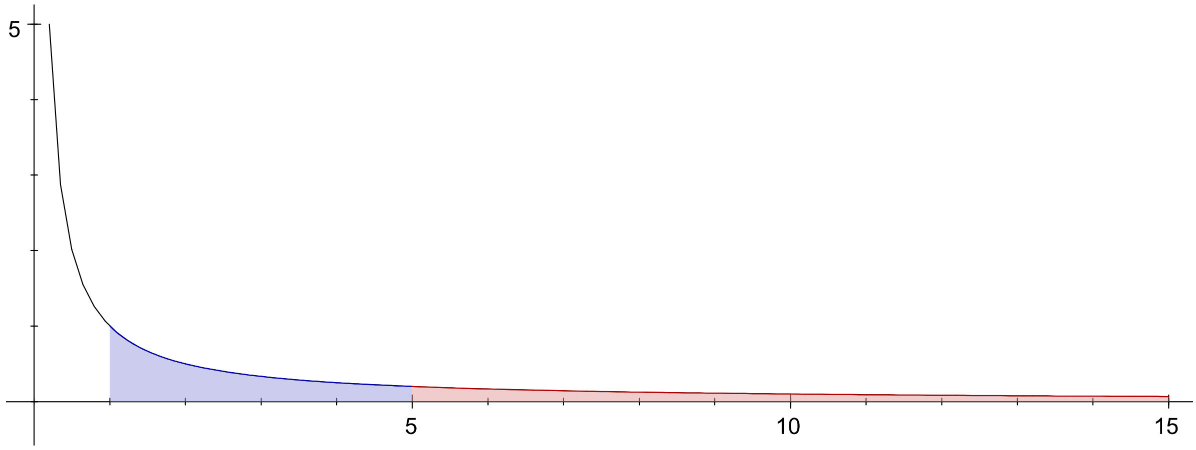


En effet l'aire bleue est l'intégrale de 5/x entre 1 et 3 et l'aire rouge s'obtient en quintuplant les abscisses ce qui a pour effet de diviser les ordonnées par 5 (puisque c'est la fonction 5/x). En fait toute fonction de la forme k/x a la même propriété. Par exemple 1/x :



2) Somme d'aires

Grégoire de Saint-Vincent en déduit une propriété importante de l'aire sous l'hyperbole (d'équation y=1/x). Par exemple si on regarde l'aire entre 1 et 15, elle peut être décomposée en deux morceaux, bleu et rouge :



Mais le morceau rouge est l'aire entre 5 et 15, qui, comme l'avait remarqué Grégoire de Saint-Vincent, est égale à l'aire entre 1 et 3. Donc l'aire entre 1 et 15 est formée de deux morceaux :

- l'aire entre 1 et 5 (en bleu)
- l'aire entre 1 et 3 (en rouge)

et elle est donc la somme des aires entre 1 et 5, et entre 1 et 3. La fonction qui, à x, associe l'aire entre 1 et x, est donc un ***logarithme*** (elle transforme les multiplications comme 5×3 en additions).

3) Le nom du logarithme

Les frères Bernoulli appellent ce logarithme « hyperbolique » (puisque c'est l'aire sous l'hyperbole) et Euler reprend ce nom. Puis il l'appelle « naturel » voire sans adjectif :

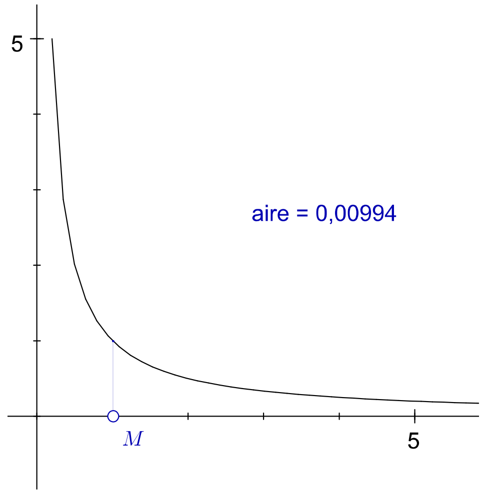
poſant e pour marquer le nombre, dont le logarithme eſt $= 1$, on ſait que $\frac{1}{e}$ exprime cette dernière ſerie.

Aujourd'hui on appelle « népérien » ce logarithme bien que Neper ne l'ait pas connu.

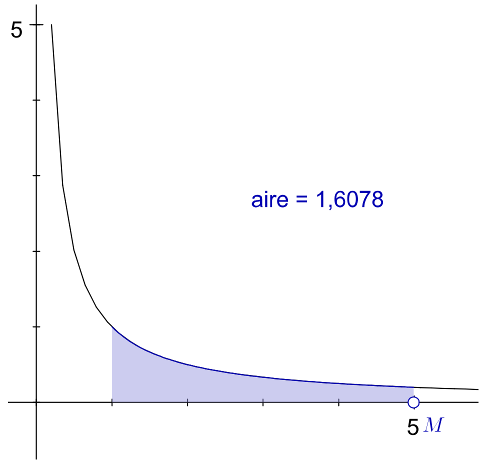
II/ Logarithme comme intégrale

Avec CaRMetal, on crée la représentation graphique de la fonction 1/x, puis on crée un point M attaché à l'axe des abscisses, et l'expression `sum(1/x(M)*d(x(M)))`. Alors si on promène M lentement sur l'axe des abscisses, l'expression « somme des dx/x » (appelée E1 par CaRMetal) est une valeur approchée du logarithme de x.

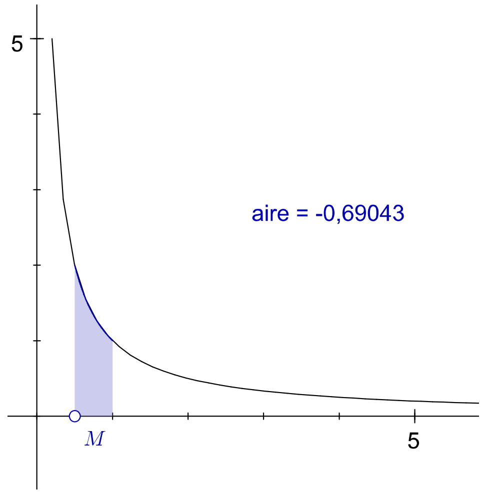
Au début x est en 1 et n'ayant pas bougé, la valeur de dx est donc restée égale à 0 et la somme de 0 est 0 :



Si on fait aller M jusqu'à x=5, on a une valeur approchée du logarithme de 5 :

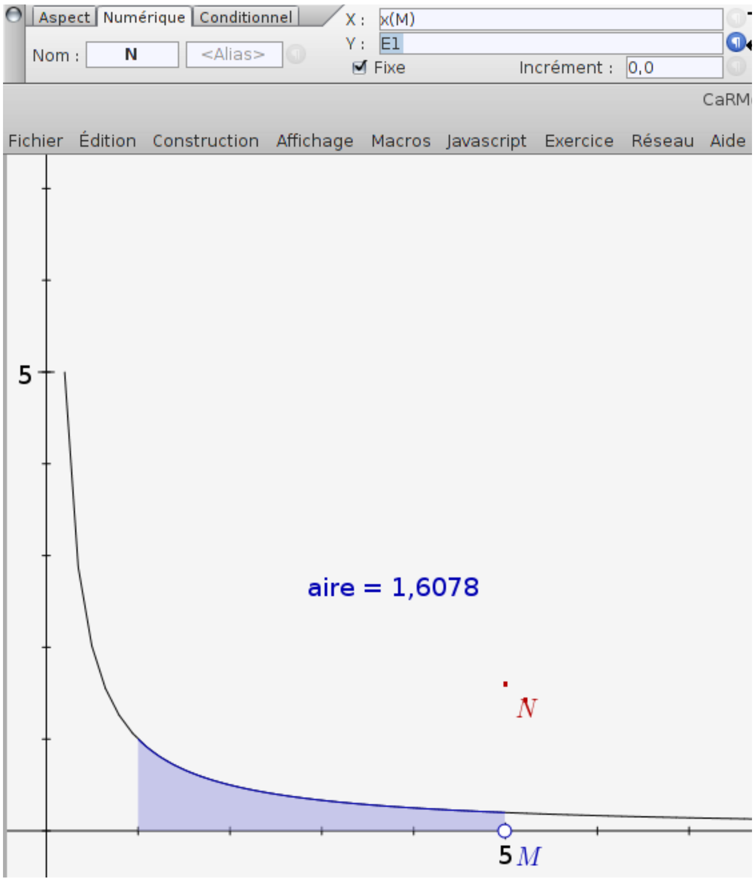


Mais si on fait aller M vers la gauche, dx est négatif et l'intégrale aussi sera négative (ici le logarithme de 0,5) :

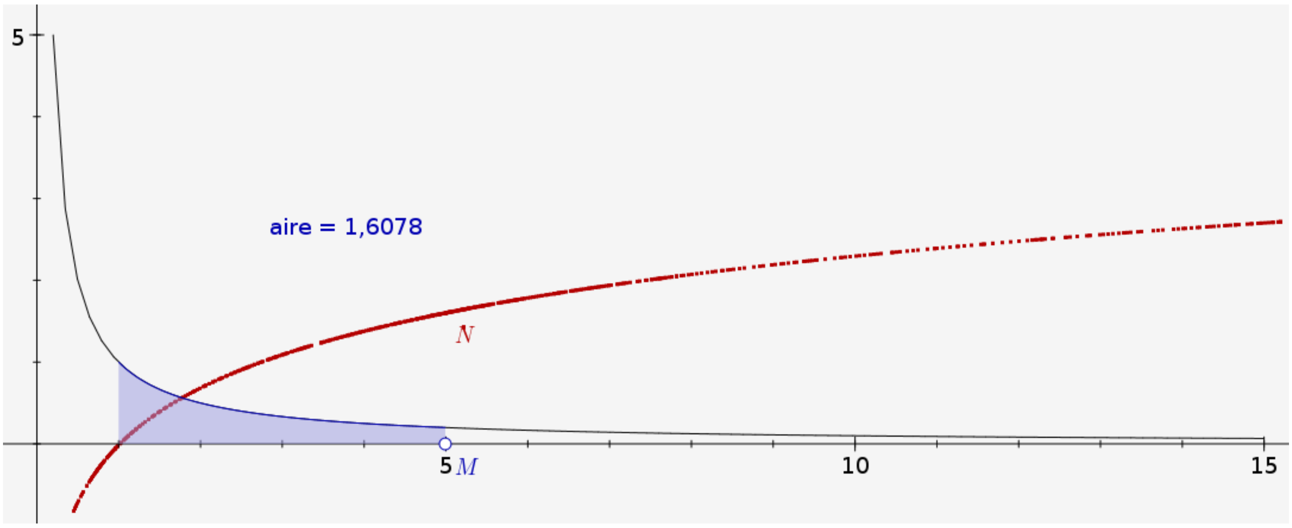


Si maintenant on crée un point N ayant pour abscisse, celle de M, et pour ordonnée, l'expression

E1 :



En activant la trace de N et en bougeant lentement M, on voit apparaître une courbe :

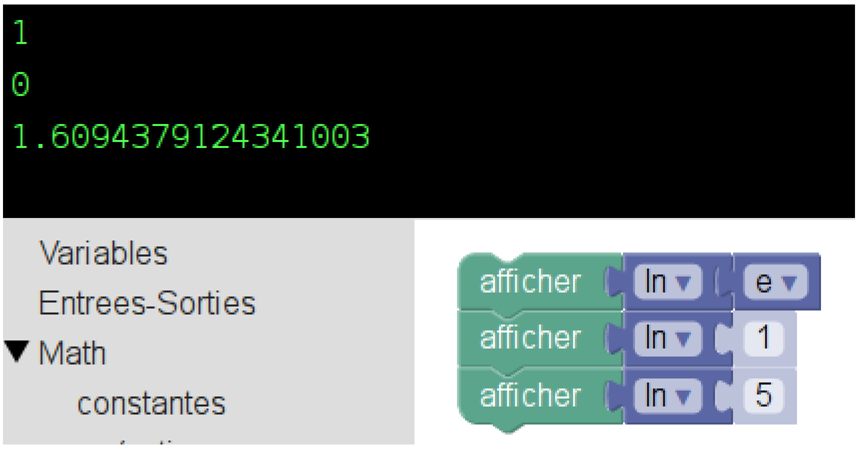


C'est la représentation graphique du logarithme, c'est-à-dire la fonction qui, à x, associe l'aire sous l'hyperbole entre 1 et x.

III/ Calcul

1) Avec SofusPy974

Le logarithme fait partie des fonctions. On vérifie que le logarithme de 5 est proche de 1,6 mais aussi que le logarithme de 1 est égal à 0 (aire nulle) et que le logarithme de e est égal à 1 :



2) En Python

Le logarithme s'appelle log en Python[1] :

```
1.0
0.0
1.60943791243
```

```python
from math import log
from math import e

print(log(e))
print(log(1))
print(log(5))
```

3) Calculatrices

Les calculatrices ont un bouton ln qui donne le logarithme népérien. Par exemple sur Ti on peut entrer ln 2nde ÷) entrer pour vérifier que le logarithme de e est 1.

La notation officielle pour le logarithme est ln comme sur les calculatrices.

1 Attention à ne pas confondre avec le bouton log des calculatrices, qui donne le logarithme décimal. La notation Python n'est pas standard.

IV/ Étude du logarithme népérien

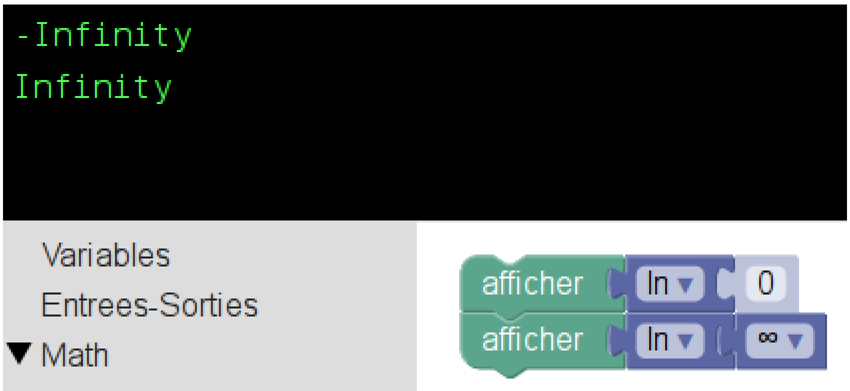
1) Propriétés algébriques

Comme ln(a×b)=ln(a)+ln(b), ln(1)=0, ln(1/a)=-ln(a) et ln(x^n)=n×ln(x). Par exemple ln(0,5) et ln(2) sont opposés, ainsi que ln(0,2) et ln(5).

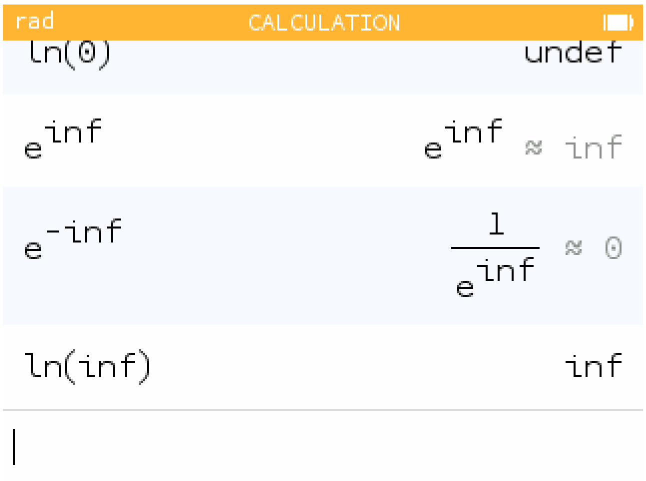
2) Limites

Ln(0)=-∞ et ln(∞)=∞

SofusPy974 le sait :



La Numworks le sait à moitié :



et Python le sait aussi à moitié : en entrant `log(float('inf'))` on a `'inf'` mais en entrant log(0) on a un message d'erreur.

3) Dérivée

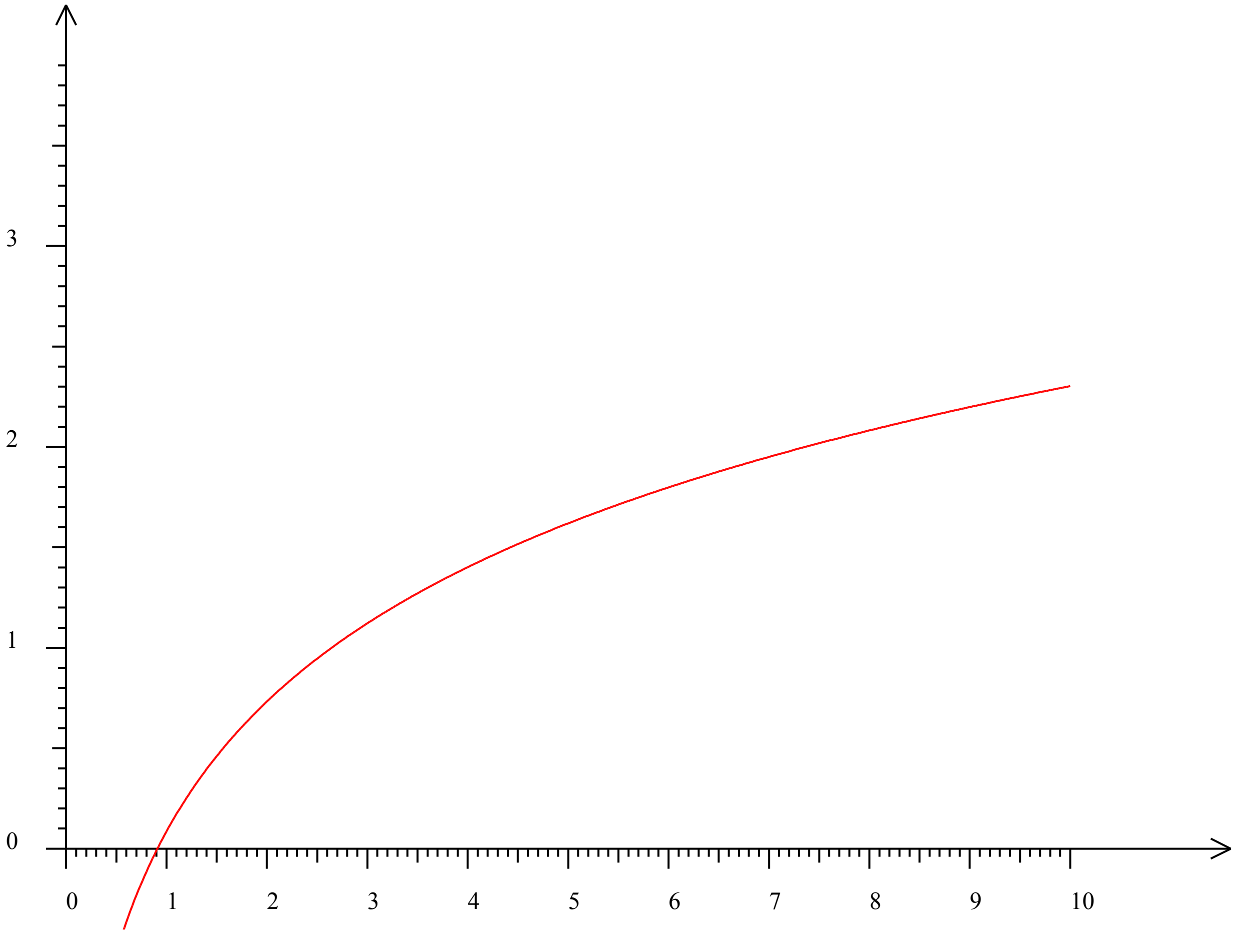
Puisque ln est une intégrale, sa dérivée est 1/x : d(ln(x))=dx/x avec la notation de Leibniz. Comme 1/x est positif sur [0,+∞], la fonction ln est croissante sur [0,+∞].

Avec la notation de Leibniz, si u est une fonction de x, d(ln(u))=d(ln(u))/du×du=1/u×du. Avec la notation de Lagrange, la dérivée de ln(u(x)) est u'/u.

Le fait que d(ln(u))=du/u facilite la résolution d'équations différentielles.

4) Représentation graphique

Comme ln(0)=-∞, la courbe admet une asymptote verticale, d'équation x=0 :



5) Application : recherche de seuil

C'est l'inflation : un biscuit qui au départ ne coûtait que 3 roubles, voit son prix doubler toutes les semaines. Au bout de combien de semaines son prix dépassera-t-il 100 roubles ?

La suite des prix du biscuit est géométrique de raison 2, donc on sait que son terme général est 3×2^n. On demande à partir de quand cette expression dépasse 100, on veut donc résoudre l'inéquation $3\times2^n > 100$. La fonction ln étant croissante, cela revient à $\ln(3\times2^n) > \ln 100$. Or le premier membre est $\ln(3) + n\times\ln(2)$, donc on résout $\ln(3)+ n\times\ln(2) > 100$ ce qui équivaut à $n\times\ln 2 > \ln(100)-\ln(3)$ puis $n> (\ln(100)-\ln(3)\ ) / \ln(2)$. En appuyant sur ln 100 ÷ 3) ÷ ln 2) entrer on a un peu plus de 5 : il faudra donc 6 semaines pour que le biscuit coûte plus de 100 roubles.