Problème du prêt I

Je souhaite emprunter de l'argent, pour acheter une maison et une voiture. J'ai contacté divers organismes prêteurs qui m'ont proposé différentes combinaisons de délais et de taux ; d'autres n'annoncent pas de taux mais directement le montant de la mensualité. Dans les rares cas où le taux et la mensualité étaient annoncés, j'ai recalculé la mensualité moi-même ... et abouti à un montant inférieur à celui exigé. Curieusement, la différence tend à être plus importante pour les taux "voiture" que pour les taux "maison". D'où viennent ces divergences, variables d'une banque à l'autre mais systématiquement en ma défaveur ? Comment puis-je vérifier, et comparer différentes propositions ?

Problème du prêt II

L'emprunteur reçoit du prêteur une somme S et s'engage à la rembourser, accrue des intérêts, sous forme de "périodicités" constantes. On fixe un taux t et un nombre de périodes n. Supposons pour fixer les idées que la période est le mois et que le taux fixé est mensuel. On observe d'abord qu'une somme S, au terme d'un nombre n de mois, vaudra

$$S' = S(1+t)^n\,; \tag{ic1}$$

c'est la formule classique des intérêts composés. Dans le cas de remboursements mensuels constants, la formule devient

$$S' = M(1+t)^{n-1} + M(1+t)^{n-2} + \cdots + M(1+t) + M\,; \tag{ic2}$$

le $i$ème terme $M(1+t)^{n-i}$ représente la valeur à l'échéance (au terme du $n$ième mois) de la mensualité M payée au terme du $i$ème mois, et qui s'est donc valorisée pendant $(n-i)$ mois. On utilise la formule bien connue

$$\sum_{i=0}^{n-1} b^i = \frac{b^n - 1}{b - 1},$$

où $b = 1 + t$, pour effectuer la somme des valorisations des n mensualités et, par élimination de S' entre ic1 et ic2, on tire

$$S(1+t)^n = M\frac{(1+t)^n - 1}{t} \tag{ic3}$$

ou encore

$$M = \frac{St(1+t)^n}{(1+t)^n - 1} \tag{ic4}$$

Problème du prêt III

```
(define periodicite
  (lambda (S t n)
    (/ (* S t (expt (+ 1 t) n)) (- (expt (+ 1 t) n) 1))))

(define periodicite
  (lambda (S t n)
    (let ((aux (expt (+ 1 t) n))) (/ (* S t aux) (- aux 1)))))
```

La fonction `periodicite` permet de calculer la mensualité en fonction de la somme à emprunter, du taux d'intérêt mensuel et de la durée du prêt en mois.
En pratique, l'emprunteur qui connaît la mensualité requise, ou sa capacité maximale de remboursement, peut se poser les trois questions suivantes :

Etant donné que ma capacité de remboursement mensuel est de M euros,
- quelle somme maximale puis-je emprunter au taux mensuel t, pour n mois ?
- à quel taux mensuel maximal t puis-je emprunter la somme S, remboursable en n mois ?
- en combien de mois minimum puis-je rembourser la somme S empruntée au taux mensuel t ?

Problème du prêt IV

Les trois programmes d'inversion sont construits de la même manière :

```
(define periodicite<-somme
  (lambda (S tn) (periodicite S (car tn) (cadr tn))))

(define somme<-periodicite (inv+ periodicite<-somme))

(define somme
  (lambda (M t n) (somme<-periodicite M (list t n))))

(periodicite 100000 0.004 240)         648.957469845475
(somme 648.957469845475 0.004 240)     100000.0
```

Problème du prêt V

```
(define periodicite<-taux
  (lambda (t Sn) (periodicite (car Sn) t (cadr Sn))))
(define taux<-periodicite (inv+ periodicite<-taux))
(define taux (lambda (S M n) (taux<-periodicite M (list S n))))

(define periodicite<-nombre-periodes
  (lambda (n St) (periodicite (car St) (cadr St) n)))
(define nombre-periodes<-periodicite
  (inv- periodicite<-nombre-periodes))
(define nombre-periodes
  (lambda (S t M) (nombre-periodes<-periodicite M (list S t))))

;; (periodicite 100000 0.004 240)                   648.957469845475
;; (somme 648.957469845475 0.004 240)               100000.0
;; (taux 100000 648.957469845475 240)               0.0039999999997
;; (nombre-periodes 100000 0.004 648.957469845475)  240.00000000001
```

Problème du prêt VI

Première source de divergence : “simplifier” la conversion entre données annuelles et données mensuelles. Deux variantes existaient (et ont récemment été rendues illégales) :

- Considérer que le taux mensuel à appliquer est le douzième du taux annuel affiché ;
- Considérer que la mensualité à payer est le douzième de l’annuité calculée sur base du taux annuel affiché.

La première méthode est systématiquement défavorable à l’emprunteur car

$$(1 + t_a) = (1 + t_m)^{12} > 1 + 12t_m\,.$$

Les passages du taux annuel au taux mensuel (exact) et réciproquement se programment aisément :

```
(define tm<-ta
  (lambda (ta) (- (expt (+ ta 1) 1/12) 1)))

(define ta<-tm
  (lambda (tm) (- (expt (+ tm 1) 12) 1)))
```

L'organisme prêteur qui applique un taux mensuel effectif égal au douzième de son taux annuel nominal utilise en fait un taux annuel effectif supérieur au taux nominal annoncé ; pour savoir quel taux annuel lui sera réellement appliqué, le client pourra utiliser le programme suivant :

```
(define ta<-ta_1 (lambda (t) (ta<-tm (/ t 12))))
; exemple:  (ta<-ta_1 0.06)   .0616778
```

Problème du prêt VIII

La deuxième variante est, elle aussi, défavorable à l'emprunteur, qui rembourse chaque mensualité avec une avance de 1 à 11 mois, c'est-à-dire avec, en moyenne, cinq mois et demi d'avance. Ici, le désavantage dépend non seulement du taux mais aussi de la durée du prêt. On utilise le programme

```
(define compose
  (lambda (f g) (lambda (x) (f (g x)))))

(define compose-list
  (lambda (f-list)
    (if (null? f-list)
        (lambda (x) x)
        (compose (car f-list)
                 (compose-list (cdr f-list))))))
```

Problème du prêt IX

```
(define ta<-ta-2
  (compose-list
    (list ta<-tm tm<-mensu mensu<-annu annu<-ta-2)))

(define annu<-ta-2
  (lambda (t) (periodicite *S* t *n*)))

(define mensu<-annu
  (lambda (a) (/ a 12.0)))

(define tm<-mensu
  (lambda (M) (taux *S* M (* 12 *n*))))

(define ta<-tm
  (lambda (tm) (- (expt (+ tm 1) 12) 1)))
```

C'est pire que précédemment, surtout pour les courtes durées :

```
; *n* 20 :  (ta<-ta-2 .06)   .063523
; *n* 10 :  (ta<-ta-2 .06)   .066294
; *n*  3 :  (ta<-ta-2 .06)   .079255
```

Problème du prêt X

Solution sans variables globales gênantes :

```
(define make-ta<-ta-2
  (lambda (n)
    (compose-list
 ; ta<-tm ; tm<-mensu ; mensu<-annu ; annu<-ta-2
      (list
        (lambda (tm) (- (expt (+ tm 1) 12) 1))
        (lambda (M) (taux *S* M (* 12 n)))
        (lambda (a) (/ a 12.0))
        (lambda (t) (periodicite *S* t n))))))
```

On notera que `make-ta<-ta-2` est une fonction qui à tout nombre positif (représentant la durée du prêt) associe une fonction équivalente à `ta<-ta-2` donnée plus haut.

Problème du prêt XI

En reprenant les mêmes exemples que précédemment, on obtient

```
(define *S* 1234567)    ... ;; S quelconque
((make-ta<-ta-2 20) .06)    .063523
((make-ta<-ta-2 10) .06)    .066294
((make-ta<-ta-2  3) .06)    .079255
```

La technique consistant à écrire un “générateur de fonction” permet d’éviter les variables globales gênantes sans que l’on doive modifier le nombre d’arguments des fonctions impliquées.

Problème du prêt XII

Deuxième cause de divergence : la notion de taux d'intérêt est parfois remplacée par la notion "voisine" de taux de chargement. L'emprunteur d'une somme S remboursable en n mois au taux de chargement mensuel t_c rembourse chaque mois la somme $M = S'/n = (1/n + t_c)S$.

Calcul de la mensualité à partir de la somme à emprunter, du taux de chargement ou du taux d'intérêt et de la durée du prêt en mois :

```
(define mensualite<-tm periodicite)

(define tm<-mensualite taux)

(define mensualite<-tc
  (lambda (S tc n) (* (+ (/ 1 n) tc) S)))

(define tc<-mensualite
  (lambda (S M n) (- (/ M S) (/ 1 n))))
```

On en tire immédiatement les fonctions de conversion :

```
(define tc<-tm
  (lambda (tm n) (tc<-mensualite *S* (mensualite<-tm *S* tm n) n)))

(define tm<-tc
  (lambda (tc n) (tm<-mensualite *S* (mensualite<-tc *S* tc n) n)))
```

Problème du prêt XIII

On calcule d'abord quel taux d'intérêt mensuel correspond à un taux de chargement de 0.5 %, pour des durées de prêt de 1, 12, 30 et 240 mois ; on obtient

```
(tm<-tc 0.005   1)   0.005000
(tm<-tc 0.005  12)   0.009080
(tm<-tc 0.005  30)   0.009265
(tm<-tc 0.005 240)   0.007719
```

On calcule de même quel taux de chargement correspond à un taux d'intérêt mensuel de 0.5 % :

```
(tc<-tm 0.005   1)   0.005000
(tc<-tm 0.005  12)   0.002733
(tc<-tm 0.005  30)   0.002646
(tc<-tm 0.005 240)   0.002998
```

On voit que la confusion entre les deux notions de taux peut coûter cher !