

COUPLAGE DANS UN GRAPHE BIPARTI ÉQUILIBRÉ (MINES 2012)

Durée : 3 heures

Préliminaire concernant la programmation

Lorsque le candidat écrira une fonction il pourra faire appel à une autre fonction définie dans les questions précédentes. Si les paramètres d'une fonction à écrire sont supposés vérifier certaines hypothèses, il ne sera pas utile dans l'écriture de cette fonction de tester si les hypothèses sont bien vérifiées.

Dans l'énoncé du problème, un même identificateur écrit dans deux polices de caractères différentes désignera la même entité, mais du point de vue mathématique pour la police en italique (par exemple n) et du point de vue informatique pour celle en romain (par exemple `n`).

Un *graphe* est défini par deux ensembles X et E . L'ensemble X est un ensemble fini d'éléments appelés *sommets*. L'ensemble E est un ensemble de paires de sommets ; un élément $\{x, y\}$ de E est appelé *arête* de G et x et y sont les *extrémités* de l'arête. L'*ordre* d'un graphe G est le nombre de sommets de G . Si $\{x, y\}$ est une arête de G on dit que x et y sont *voisins*. Le *degré* d'un sommet x est le nombre de voisins de x .

On dit que deux arêtes d'un graphe G sont *incidentes* si elles ont une extrémité en commun. On appelle *couplage* dans G un ensemble d'arêtes de G deux à deux non incidentes.

Un graphe G est dit *biparti* si on peut partitionner son ensemble de sommets X en deux sous-ensembles A et B non vides de sorte que toute arête ait une extrémité dans A et une extrémité dans B . Si les ensembles A et B ont même cardinal on dit qu'il s'agit d'un *graphe biparti équilibré*. Dans toute le problème on ne considère que des graphes bipartis équilibrés. On note n le cardinal commun aux ensembles A et B ; l'ordre du graphe est donc égal à $2n$. On suppose qu'on a toujours $n \geq 1$. Les sommets de A sont numérotés de 0 à $n-1$ et nommés $0_A, 1_A, 2_A, \dots, (n-1)_A$; les sommets de B sont numérotés de 0 à $n-1$ et nommés $0_B, 1_B, 2_B, \dots, (n-1)_B$. Une arête de G est toujours écrite en mettant d'abord l'extrémité qui est dans A puis celle qui est dans B .

On représente toujours les graphes bipartis équilibrés par des schémas comme on peut le voir dans la figure 1 avec le graphe G_0 en représentant les sommets de A à gauche et les sommets de B à droite.

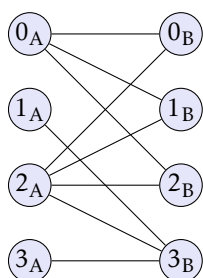


FIGURE 1 – Le graphe G_0 .

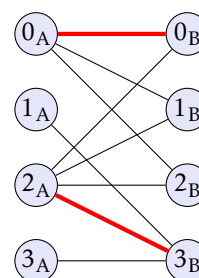


FIGURE 2 – Le graphe G_0 et le couplage C_0 .

Dans le graphe G_0 les arêtes $\{0_A, 0_B\}$ et $\{2_A, 3_B\}$ étant non incidentes elles forment un couplage nommé C_0 dont les arêtes sont dessinées en gras figure 2. On dit que dans ce couplage :

- le sommet 0_A est couplé au sommet 0_B et réciproquement ;
- le sommet 2_A est couplé au sommet 3_B et réciproquement ;
- les sommets $1_A, 3_A, 1_B$ et 2_B sont non couplés.

Le cardinal d'un couplage est le nombre d'arêtes de celui-ci ; par exemple le cardinal de C_0 vaut 2.

Partie I. Généralités

Question 1. Exhiber un couplage de cardinal 3 de G_0 , puis indiquer s'il existe dans G_0 un couplage de cardinal 4. Justifier la réponse.

Un graphe biparti équilibré d'ordre $2n$ est représenté par une matrice carrée de dimension $n \times n$ dont les lignes correspondent aux éléments de A et les colonnes aux éléments de B . Les cases de cette matrice sont indicées par (i, j) avec $0 \leq i \leq n-1$ et $0 \leq j \leq n-1$ et contiennent des valeurs booléennes : la case d'indice (i, j) contient la valeur « vrai » (**true** en CAML) si $\{i_A, j_B\}$ est une arête du graphe ; elle contient la valeur « faux » (**false** en CAML) dans le cas contraire. Le graphe G_0 ci-dessus est donc représenté par la matrice :

i / j	0	1	2	3
0	V	V	V	F
1	F	F	F	V
2	V	V	V	V
3	F	F	F	V

Indications pour la programmation

Un vecteur est par la suite nommé aussi tableau. On nomme *matrice* un tableau à deux dimensions (un vecteur de vecteurs). La matrice représentant un graphe biparti équilibré d'ordre $2n$ est codée en CAML par une matrice $n \times n$ de booléens. La matrice représentant G_0 est ainsi codée par :

```
let g0 = [| [| true; true; true; false |];
            [| false; false; false; true |];
            [| true; true; true; true |];
            [| false; false; false; true |] |] ;;
```

Un couplage est codé par un vecteur de n entiers ; le couplage C_0 est ainsi codé par :

```
let c0 = [| 0; -1; 3; -1 |] ;;
```

Une arête est codée par un vecteur **a** de deux entiers avec **a.(0)** dans A et **a.(1)** dans B.

Question 2. Soit G un graphe biparti équilibré d'ordre $2n$. On considère un tableau C d'entiers de longueur n et contenant dans ses cases indicées de 0 à $n-1$ soit la valeur -1 , soit une valeur comprise entre 0 et $n-1$. Il s'agit de savoir si ce tableau C représente ou non un couplage dans G .

Écrire en CAML une fonction **verifie** telle que si **g** est une matrice codant le graphe G et **c** un vecteur codant le tableau C alors **verifie g c** renvoie **true** si le tableau C représente un couplage dans G et **false** sinon.

Indiquer la complexité de la fonction **verifie**.

Question 3. On considère un tableau C de longueur n codant un couplage d'un graphe G . Il s'agit d'écrire une fonction qui calcule le cardinal de ce couplage.

Écrire en CAML une fonction **cardinal** telle que si **c** est un vecteur codant un couplage alors **cardinal c** renvoie le cardinal de ce couplage.

Indiquer la complexité de la fonction **cardinal**.

Partie II. Un algorithme pour déterminer un couplage maximal

On dit qu'un couplage C dans un graphe G est *maximal* si toute arête de G n'appartenant pas à C est incidente à au moins une arête de C . Par exemple, le couplage C_0 de G_0 est maximal. Un couplage maximal de G n'est pas forcément de cardinal maximum parmi les couplages de G . On cherche à concevoir un algorithme qui détermine un couplage maximal dans un graphe binaire équilibré G .

L'algorithme, nommé *algo_approche*, est le suivant :

- on commence avec un couplage vide C ;
- tant que G possède au moins une arête :
 - on choisit une arête a de G dont la somme des degrés des extrémités soit minimale ;
 - on ajoute l'arête a au couplage C ;
 - on retire de G l'arête a et toutes les arêtes incidentes à a .

On admettra que le résultat est, par construction, un couplage maximal.

Question 4. Appliquer *algo_approche* au graphe G_0 (représenté figure 1).

On considère par la suite le graphe biparti équilibré G_1 d'ordre 12 représenté sur la figure 3.

Question 5. On applique *algo_approche* au graphe G_1 . Détermine la première arête a_1 choisie par *algo_approche* ; tracer le graphe obtenu après suppression de a_1 et des arêtes incidentes à a_1 . Montrer que le couplage obtenu par *algo_approche* est de cardinal au plus 5 et indiquer s'il est de cardinal maximal parmi les couplages de G_1 .

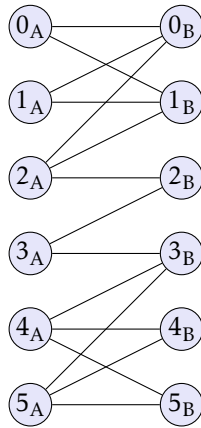


FIGURE 3 – Le graphe G_1 .

Question 6. Soit G un graphe biparti équilibré d'ordre $2n$. Il s'agit de décrire en langage de programmation une fonction `arete_min` qui détermine une arête de G dont la somme des degrés des extrémités soit minimum. Si le graphe possède au moins une arête, cette fonction modifie le tableau a de deux entiers reçu en paramètre pour mettre dans les deux cases de a les numéros des deux extrémités d'une arête qui atteint ce minimum ; dans ce cas la fonction renvoie la valeur « vrai » (**true**) ; sinon elle renvoie la valeur « faux » (**false**).

Écrire en CAML une fonction `arete_min` telle que si g est une matrice codant le graphe G et a un vecteur de deux entiers alors `arete_min g a` effectue les opérations décrites ci-dessus, en modifiant le tableau a dans le cas où G possède au moins une arête.

Indiquer la complexité de la fonction `arete_min`.

Question 7. Il s'agit d'écrire une fonction `supprimer` qui supprime d'un graphe biparti équilibré une arête a donnée et toutes les arêtes incidentes à a .

Écrire en CAML une fonction `supprimer` telle que si g est une matrice codant un graphe biparti équilibré G et a un vecteur de deux entiers codant une arête a de G alors `supprimer g a` modifie g pour que, après modifications, g code le graphe obtenu à partir de G en supprimant a ainsi que toutes les arêtes incidentes à a .

Indiquer la complexité de la fonction `supprimer`.

Question 8. Il s'agit de définir l'algorithme `algo_approche` décrit au début de la deuxième partie.

Écrire en CAML une fonction `algo_approche` telle que si g est une matrice qui code un graphe biparti équilibré G , `algo_approche` effectue `algo_approche` à partir d'une copie de G et renvoie un vecteur codant le couplage obtenu.

Indication : on pourra utiliser sans la définir une fonction `copy_matrix` telle que, si g est une matrice codant un graphe biparti équilibré G , `copy_matrix g` renvoie une matrice identique à g . Cette fonction sera utilisée pour que la fonction `algo_approche` ne modifie pas le contenu de la matrice reçue en paramètre.

Indiquer la complexité de la fonction `algo_approche`.

Partie III. Recherche exhaustive d'un couplage de cardinal maximum

Question 9. Soit G un graphe biparti équilibré d'ordre $2n$. Il s'agit d'écrire une fonction nommée `une_arete` qui recherche une arête quelconque de G . Si G possède au moins une arête la fonction mémorise la première arête rencontrée a dans un vecteur de deux entiers reçu en paramètre ; elle arrête alors sa recherche et renvoie la valeur « vrai » (**true**) ; sinon la fonction renvoie la valeur « faux » (**false**).

Écrire en CAML une fonction `une_arete` telle que si g est une matrice codant un graphe G et a un vecteur de deux entiers destiné à coder l'arête a alors `une_arete g a` effectue les opérations décrites ci-dessus, en modifiant le vecteur a dans le cas où G possède au moins une arête.

Question 10. On cherche à établir un algorithme **récuratif** nommé `meilleur_couplage` qui permette de déterminer un couplage de cardinal maximal dans un graphe biparti équilibré. Le principe est le suivant.

Si le graphe courant ne contient aucune arête, le cardinal maximum d'un couplage est 0 et aucun sommet n'est couplé.

Dans le cas contraire, l'algorithme considère une arête quelconque a du graphe courant et recherche successivement :

- un couplage de cardinal maximal parmi les couplages du graphe courant ne contenant pas a ;
- un couplage de cardinal maximal parmi les couplages du graphes contenant a .

L'algorithme déduit alors un couplage de cardinal maximal.

Écrire en CAML une fonction récursive **meilleur_couplage** telle que, si **g** est une matrice codant un graphe biparti équilibré **G**, **meilleur_couplage g** renvoie un vecteur codant un couplage de cardinal maximal dans **G**. La fonction utilisera le principe décrit plus haut.

Indication : on pourra utiliser sans la définir une fonction **copy_matrix** telle que, si **g** est une matrice codant un graphe biparti équilibré **G**, alors **copy_matrix g** renvoie une matrice identique à **g**.

Partie IV. L'algorithme hongrois

On considère un graphe biparti équilibré **G** et un couplage **C**. Une *chaîne* de **G** est une suite $x_0, x_1, \dots, x_p$ ($p \geq 0$) de sommets distincts telle que, pour $k \in \llbracket 0, p-1 \rrbracket$ $\{x_k, x_{k+1}\}$ est une arête de **G**. Le sommet x_0 s'appelle l'*origine* de la chaîne et le sommet x_p l'*extrémité* de la chaîne.

Une chaîne $x_0, x_1, \dots, x_p$ ($p \geq 0$) de **G** est dite *alternée relativement à C* si :

- pour tout indice k pair, x_k est dans **A** ;
- pour tout indice k impair, x_k est dans **B** ;
- le sommet x_0 n'est pas couplé ;
- pour tout i vérifiant $0 \leq 2i \leq p-1$ l'arête $\{x_{2i}, x_{2i+1}\}$ n'appartient pas à **C** ;
- pour tout i vérifiant $0 \leq 2i \leq p$ l'arête $\{x_{2i-1}, x_{2i}\}$ appartient à **C**.

Autrement dit :

- l'origine de la chaîne est dans **A** et n'est pas couplée ;
- la première arête de la chaîne n'est pas dans **C**, la deuxième est dans **C**, la troisième n'est pas dans **C** et ainsi de suite.

Une chaîne $x_0, x_1, \dots, x_p$ alternée relativement au couplage **C** est dite *chaîne alternée augmentante relativement à C* si $p \geq 1$ et si x_p n'est pas couplé, ce qui entraîne que x_p est dans **B**. Par exemple, dire qu'une chaîne $x_0, x_1, x_2, x_3, x_4, x_5$ constitue une chaîne alternée augmentante relativement à un couplage **C** signifie que :

- x_0, x_2 et x_4 sont des sommets de **A**, x_1, x_3 et x_5 sont des sommets de **B** ;
- $\{x_0, x_1\}, \{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_4\}, \{x_4, x_5\}$ sont des arêtes de **G** ;
- x_0 n'est pas couplé dans **C**, x_5 n'est pas couplé dans **C** ;
- x_1 est couplé avec x_2 , x_2 n'est pas couplé avec x_3 et x_3 est couplé avec x_4 .

Question 11. On considère le graphe **G**₁ et le couplage **C**₁ constitué des arêtes $\{0_A, 0_B\}, \{1_A, 1_B\}, \{3_A, 2_B\}, \{4_A, 3_B\}, \{5_A, 5_B\}$ représentés sur la figure 4 en gras.

Après avoir indiqué le seul sommet de **A** qui puisse être l'origine d'une chaîne alternée augmentante relativement à **C**₁ et le seul sommet **B** qui puisse être l'extrémité d'une chaîne alternée augmentante relativement à **C**₁, déterminer une chaîne alternée augmentante relativement à **C**₁.

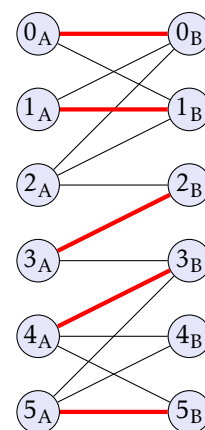


FIGURE 4 – Le graphe **G**₁ et le couplage **C**₁.

Question 12. On considère un graphe biparti équilibré **G** et un couplage **C** dans **G**. Montrer que s'il existe une chaîne alternée augmentante relativement à **C** alors il existe dans **G** un couplage dont le cardinal est égal au cardinal de **C** augmenté de 1.

On admet le théorème suivant : *un couplage **C** d'un graphe biparti équilibré **G** est de cardinal maximal si et seulement si il n'existe pas dans **G** de chaîne alternée augmentante relativement à **C**.*

Soit G un graphe biparti équilibré. L'algorithme hongrois s'appuie sur le théorème ci-dessus et détermine un couplage de cardinal maximal dans G . L'algorithme débute avec un couplage C de cardinal nul ; tant qu'il existe une chaîne augmentante relativement à C , l'algorithme modifie C pour incrémenter de 1 le cardinal du couplage en utilisant une telle chaîne augmentante.

La suite du problème a pour objectif de programmer cet algorithme. On commence par étudier la recherche d'une chaîne alternée augmentante.

On considère un graphe biparti équilibré G et un couplage C dans G . On va chercher s'il existe une chaîne alternée augmentante relativement à C . Pour cela, on recherche les sommets qui sont extrémités de chaînes alternées en procédant de proche en proche à partir des sommets de A non couplés.

Un sommet y est dit *atteint* si une chaîne alternée relativement à C et d'extrémité y est mise en évidence. Au départ tous les sommets non couplés de A (un tel sommet est extrémité d'une chaîne alternée réduite à ce sommet) sont considérés comme atteints ; aucun autre sommet n'est considéré comme atteint. Un sommet y **non encore atteint** peut être atteint à partir d'un de ses voisins x déjà atteint si on a :

- soit y est dans B (et donc x est dans A) et l'arête $\{x, y\}$ n'est pas dans le couplage C ;
- soit y est dans A (et donc x est dans B) et l'arête $\{y, x\}$ est dans le couplage C .

On utilise des *marques attribuées aux sommets*. Ces marques sont des entiers initialisés à -1 pour tous les sommets. Lorsqu'un sommet y est atteint à partir d'un sommet x la marque de y devient égale au numéro de x (on rappelle que le numéro d'un sommet i_A ou d'un sommet i_B vaut i) ; x est alors l'avant-dernier sommet dans la chaîne alternée $Ch(y)$ d'extrémité y mise en évidence. La chaîne $Ch(y)$ peut être retrouvée à l'envers, de proche en proche, grâce aux marques ; dans cette chaîne, seule l'origine porte une marque de valeur -1 .

Si un sommet non couplé y de B est atteint, $Ch(y)$ est une chaîne alternée augmentante relativement à C .

Dans le cas où simultanément :

- il n'y a plus de sommet non encore atteint qui puisse être atteint ;
- aucun sommet non couplé de B n'est atteint

on admet qu'il n'existe pas de chaîne alternée augmentante relativement à C .

Remarque : les valeurs des marques peuvent dépendre de l'ordre dans lequel on atteint les sommets, sans que cela ait d'importance pour la suite du problème.

Question 13. On considère le graphe G_1 et un couplage nommé C'_1 constitué des arêtes $\{0_A, 0_B\}$, $\{2_A, 1_B\}$, $\{3_A, 2_B\}$, $\{4_A, 4_B\}$, $\{5_A, 5_B\}$, en gras sur la figure 5.

Certains sommets ont été atteints ; sur la figure 5, les marques attribuées sont portées à côté des sommets, les sommets atteints sont encadrés.

Utiliser les marques pour reconstituer la chaîne alternée arrivant dans le sommet 3_B et correspondant aux marques. Indiquer s'il s'agit d'une chaîne alternée augmentante relativement à C'_1 .

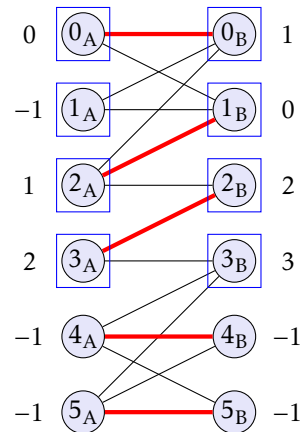


FIGURE 5 – Le graphe G_1 et le couplage C'_1 .

On considère quatre tableaux :

- un tableau nommé c codant un couplage C ;
- un tableau nommé r (pour réciproque) indicé par $0, 1, \dots, n-1$; soit j vérifiant $0 \leq j \leq n-1$; si le sommet j_B n'est pas couplé dans C , la case d'indice j du tableau r contient la valeur -1 ; si le sommet j_B est couplé avec le sommet i_A la case d'indice j du tableau r comporte la valeur i ;
- un tableau nommé m_A indicé par $0, 1, \dots, n-1$ servant à contenir les marques des sommets de A ;
- un tableau nommé m_B indicé par $0, 1, \dots, n-1$ servant à contenir les marques des sommets de B .

Question 14. On suppose qu'une chaîne $Ch(x_p) = x_0, x_1, \dots, x_p$ alternée augmentante relativement à un couplage C a été déterminée et codée grâce aux marques. D'après la question 13 il existe un couplage de cardinal égal à celui de C augmenté de 1. La fonction **actualiser** reçoit en paramètres les quatre tableaux décrits ci-dessus ainsi que le numéro de x_p ; elle transforme alors les tableaux **c** et **r** pour qu'ils correspondent à un couplage, obtenu à partir de C et de $Ch(x_p)$, dont le cardinal est celui du couplage C augmenté de 1.

Écrire en CAML une fonction **actualiser** telle que si **c**, **r**, **mA**, **mB** sont des vecteurs de n entiers qui correspondent à la description donnée plus haut et si **numero** est un entier donnant le numéro de x_p alors **actualiser c r mA mB numero** modifie les vecteurs **c** et **r** pour obtenir un couplage de cardinal égal à celui de C augmenté de 1.

Question 15. On considère le graphe G_2 ci-contre et un couplage nommé C_2 constitué des arêtes $\{0_A, 0_B\}$, $\{2_A, 2_B\}$, $\{3_A, 3_B\}$, $\{4_A, 4_B\}$ en gras sur la figure 6.

Recopier la figure, encadrer tous les sommets qui peuvent être atteints et préciser à côté des sommets les marques obtenues. Indiquer s'il existe dans G_2 une chaîne alternée augmentante relativement à C_2 .

Indication : on procédera de proche en proche à partir du seul sommet non couplé de A, c'est-à-dire à partir du sommet 1_A . L'ordre dans lequel on atteint les sommets n'a pas d'importance.

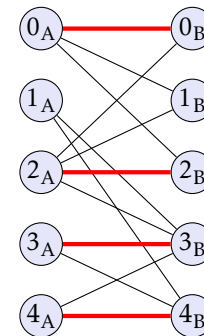


FIGURE 6 – Le graphe G_2 et le couplage C_2 .

Question 16. On suppose que certains sommets de G peuvent déjà avoir été atteints et les marques calculées en conséquence. On note x un sommet de A ou de B déjà atteint.

On définit deux nouvelles fonctions, les fonctions *chercherA* et *chercherB*.

Appliquée au sommet x , appelé sommet de départ, la fonction *chercherA* (si x est dans A) ou la fonction *chercherB* (si x est dans B) détermine des sommets non encore atteints qui peuvent être atteints, de voisin en voisin, récursivement, à partir de x , modifie les marques de ces sommets nouvellement atteints et s'arrête dans les cas suivants :

- un sommet y de B non couplé a été atteint ; elle renvoie alors le numéro du sommet y ;
- tous les sommets qui peuvent être atteints de voisin en voisin à partir de x l'ont été et aucun sommet non couplé de B n'est atteint ; elle renvoie alors la valeur -1 .

Il s'agit d'écrire les deux fonctions *chercherA* et *chercherB* en utilisant une récursivité croisée, chacune des deux fonctions pouvant faire appel à l'autre.

Définir ce qu'on appelle récursivité croisée et indiquer comment elle peut être implémentée en CAML, puis écrire en CAML les deux fonctions **chercherA** et **chercherB** ; chacune de ces deux fonctions reçoit en paramètres une matrice **g** codant le graphe G , quatre vecteurs de longueur n pour les quatre tableaux décrits plus haut : **c**, **r**, **mA**, **mB** et un entier codant le numéro du sommet de départ de la recherche.

Ces deux fonctions modifient les vecteurs **mA** et **mB** conformément à la description ci-dessus. Elles renvoient le numéro d'un sommet non couplé de B ou la valeur -1 selon le cas.

Question 17. On suppose donnés un graphe biparti équilibré G d'ordre $2n$ et un couplage C dans G . Tous les sommets de G possèdent une marque égale à -1 .

La fonction *chaîne_alternee* cherche s'il existe une chaîne alternée augmentante en appliquant la fonction *chercherA* successivement à partir des sommets non couplés de A.

Ecrire en CAML la fonction **chaîne_alternee** telle que si **g** est une matrice codant le graphe G , **c**, **r**, **mA** et **mB** des vecteurs correspondant à la description donnée précédemment, toutes les cases de **mA** et **mB** étant initialisées à -1 , alors **chaîne_alternee g c r mA mB** renvoie :

- -1 s'il n'existe pas de chaîne alternée augmentante ;
- le numéro de l'extrémité d'une chaîne alternée augmentante dans le cas contraire.

De plus, la fonction modifie les vecteurs **mA** et **mB** pour qu'ils contiennent les marques des sommets à la fin de l'exécution de la fonction.

Question 18. Dans cette question on programme l'algorithme hongrois.

Ecrire en CAML la fonction **algorithme_hongrois** telle que si **g** est une matrice codant un graphe biparti équilibré G , alors **algorithme_hongrois g** renvoie un vecteur codant le couplage obtenu par l'algorithme hongrois.