2 Arbres. 2.1 Arbres 2.1.1 2.1.2 2.2 Tas. 2.2.1	Chapitre 21	
2.1 Arbres 2.1.1 2.1.2 2.2 Tas 2.2.1	Structures de données nées, piles et files.	
	binaires. Définitions. Parcours. Définitions. Implémentation. binaires de recherche.	
2.4 Arbres 2.4.1 2.4.2	rouge-noir. Définitions. Implémentation. ations des arbres binaires. res.	
4.2 Résolut	Les propositions marquées de ★ sont au programme de colles.	
Définition 1: Une liste cha • Soit c'es	haînées, piles et files. Liste chaînée. Înée est une structure qu'on peut définir inductivement : t une liste vide. t un noeud, appelé cellule, qui contient une valeur et une référence à une autre liste chaînée	e.
de leur inserti Interface : • Empile • Dépiler	une structure de données qui permet de stocker des éléments et de les retirer dans l'ordre ir	nverse
Définition 3: Une file est uninsertion. Interface:		e leur
DéfilerTesterArbres.	un élément. si la file est vide. binaires.	
Un arbre bir Il est défini de • Soit c'es	Arbre binaire. naire est une structure de données hiérarchique où chaque élément est un noeud. e manière inductive : t une arbre vide. et constitué d'un noeud et de deux arbre binaires disjoints, appelés fils gauche et fils droit	;
	in arbre est le seul noeud qui n'a pas de père.	
Les feuilles d Une branche Les noeuds i	Feuilles, noeuds internes. 'un arbre sont les noeuds qui n'ont pas de fils. est un chemin de la racine à une feuille. enternes sont les noeuds qui ont au moins un fils. euilles + noeuds internes = noeuds	
Un arbre bina Proposition 8	Arbre binaire strict. ire strict est un arbre où tous les noeuds internes ont deux fils. E Dénombrement. e binaire strict non vide, le nombre de feulles est égal au nombre de noeuds internes plus un.	
On dénombre On dénombre En combinant	: $ \text{feuilles} = \text{noeuds} - \text{noeuds internes} $. les liaisons père-fils : $N = 2 \cdot \text{noeuds internes} $ (les noeuds internes ont deux fils). les liaisons fils-père : $N = \text{noeuds} - 1$ (la racine n'a pas de père). les deux : $ \text{noeuds} - 1 = 2 \text{noeuds internes} $. es $ \text{es} = \text{noeuds internes} + 1$.	
La profonde Un niveau d' La hauteur d	Vocabulaire. a arbre est son nombre de noeuds. ur d'un noeud est sa distance à la racine. un arbre est l'ensemble des noeuds de même profondeur. l'un arbre est la profondeur maximale de ses noeuds. La hauteur d'un arbre vide est -1, la hauteur d'un arbre réduit à sa racine est 0.	
	Arbre binaire parfait. aire est parfait si toutes ses feuilles sont de même profondeur et que tous ses noeuds int x fils.	ternes
Un arbre bina De plus, pour Preuve:	1: Nombre de noeuds d'un arbre parfait. \star ire parfait de hauteur h a $2^{h+1}-1$ noeuds. tout $k \in [\![0,h]\!]$, il a 2^k noeuds de profondeur k . ar récurrence sur la hauteur.	
Initialisation Hérédité. Su Soit un arbre Un noeud de g Ainsi, il y a d Alors le nomb La propriété e	n. Un arbre de hauteur 0 est réduit à sa racine et de taille $1 = 2^{0+1} - 1$. apposons la propriété vraie sur les arbres de hauteur h . binaire parfait de hauteur $h + 1$. profondeur $k + 1$ est le fils d'un noeud de profondeur k , qui a deux fils. eux fois plus de noeuds de profondeur $k + 1$ que de noeuds de profondeur k . re de noeuds total vaut $\sum_{k=0}^{h} 2^k = \frac{2^{h+1}-1}{2-1} = 2^{h+1} - 1$. est vraie pour $h + 1$.	
Définition 12	e, elle est vraie pour tout $h \in \mathbb{N}$. Arbre binaire complet. ire de hauteur h est complet si tous ses niveaux sont remplis, sauf peut-être le dernier, que che à droite.	ui est
La taille d'un	: Nombre de noeuds d'un arbre complet. arbre binaire complet de hauteur h est comprise entre 2^h et $2^{h+1}-1$. un arbre complet de taille n est $\lfloor \log n \rfloor$.	
2.1.2 Parcou	e est une valeur associée à un noeud d'un arbre.	
Un parcours Il existe trois • Préfixe • Infixe:	en profondeur est un parcours où on termine d'explorer une branche avant d'en visiter une a types de parcours en profondeur : : on visite le noeud, puis le fils gauche, puis le fils droit. on visite le fils gauche, puis le noeud, puis le fils droit. e : on visite le fils gauche, puis le fils droit, puis le noeud.	autre.
Un parcours profondeur su	en largeur est un parcours où on visite les noeuds de même profondeur avant de passerivante. e file pour implémenter ce parcours.	rà la
2.2.1 Définit Définition 17: Une file de p ordonné. Interface : • Insérer u	File de priorité. riorité est une structure de données qui permet de stocker des éléments d'un ensemble totale un nouvel élément.	ement
 Extraire Modifier Tester s Définition 18	l'élément le plus grand. la valeur d'un élément. i la file est vide.	
2.2.2 Implén Définition 19:	nentation. Percoler vers le bas. \star e 1 : Percoler vers le bas Un tas A , un indice i et une valeur v	
Sorties: 1 $A[i] \leftarrow v$. 2 $\max \leftarrow \inf$ 3 $\sin \max \neq i$ 4 $A[i] \leftarrow$ 5 $percol$ 6 fin	Un tas similaire à A , tel que $A[i] = v$ dice du noeud d'étiquette maximale entre $A[i]$ et ses fils. i alors $A[\max]$. $er_{vers_le_bas}(A, \max, v)$.	
Complexité. Notons $T(h)$ l On a $T(0) = 0$ Or $2^h \le n \le 2$	3,4 se terminent. Le variant d'appel est la hauteur du noeud d'indice i .	-
Correction. Soit h la haut Si $h = 0$, alors Supposons qu On prend un : Si $max = i$ alors Si $max \neq i$ alors	eur d'un noeud d'indice i . s c'est une feuille et $max = i$: il n'y a pas d'appel. e l'appel est correct pour une certaine hauteur $h - 1$. Montrons que l'appel sur h fonctionne indice i d'un noeud de hauteur h . ors par hypothèse de récurrence, les sous-arbres de i sont des tas et l'appel est correct. ors on remplace $A[i]$ par le max et la condition est donc vérifiée entre i et ses fils. e de récurrence, on sait que l'appel récursif est correct.	·.
Algorithme Entrées :	e 2 : Percoler vers le haut Un tas A , un indice i et une valeur v Un tas similaire à A , tel que $A[i] = v$	
$2 \min \leftarrow inc$ $3 \text{ si } \max \neq i$ $4 \mid A[i] \leftarrow i$ $5 \mid \text{ percol} i$ 6 fin	lice du noeud d'étiquette minimale entre $A[i]$ et son père. i alors $A[\max]$.er_vers_le_haut(A, min, v) étés que pour percoler_vers_le_bas.	
Algorithm Entrées: Sorties: $1 \text{ si } v > A[i]$	**Modification. **e 3 : Modification Un tas A , un indice i et une valeur v Un tas similaire à A , tel que $A[i] = v$ **alors **er_vers_le_bas(A, i, v)	
3 fin 4 sinon 5 percol 6 fin	$\operatorname{der_{vers}}$ $\operatorname{le_{haut}}(A, i, v)$ $\operatorname{decoule}$ decoule de celles de $\operatorname{percoler_{vers}}$ $\operatorname{le_{haut}}$.	
Entrées : Sorties : Ajouter v	Example 1. Insertion. Let A : Insertion Un tas A et une valeur v La valeur v a été insérée dans le tas A . à la fin du tas. Let A :	
Entrées :	e 5 : Extraction	
 2 suppring 3 fin 4 Garder en 5 Remplacer 6 Le percole 	et de taille 1 alors mer son élément et le renvoyer. mémoire le plus grand élément du tas (le premier). e le premier élément par le dernier. r vers le bas. e le plus grand élément.	
	e 6 : Construire un tas Un tableau T T tel qu'il vérifie la condition de tas. $\lfloor n/2 \rfloor \ \text{à 1 faire}$.er_vers_le_bas $(T, i, T[i])$.	
Soit H la hau À une certain De plus, ces n La complexité	teur du tas. On sait que percoler_vers_le_bas est en $O(H)$. e profondeur p , il y a au plus 2^p noeuds. oeuds sont à hauteur soit $h=H-p$, soit $h=H-p-1$, donc $p=H-h$ ou $p=H-h-1$ e de percoler_vers_le_bas sur un noeud de hauteur h est donc de αh , $\alpha \in \mathbb{R}$. u plus 2^{H-h} noeuds de hauteur h . pire des cas :	
De plus, on p	$\sum_{h=0}^{H} \alpha h \cdot 2^{H-h} = \alpha 2^{H} \sum_{h=0}^{H} \frac{h}{2^{h}}$ $\mapsto \sum_{h=0}^{H} x^{h}, \text{ alors } f(x) = \sum_{h=0}^{H} x^{h} = \frac{x^{H+1}-1}{x-1}.$ $\operatorname{ose} g: x \mapsto x f'(x), \text{ et pour } x < 1: \ f'(x) = \frac{Hx^{H+1}-Hx^{H}+x^{H}+1}{(x-1)^{2}} = \frac{1}{(x-1)^{2}} + Hx^{H} \frac{x-1-\frac{1}{H}}{(x-1)^{2}} \to \frac{x}{(x-1)^{2}} + \beta \text{ en particulier pour } \frac{1}{2}.$	$\frac{1}{x-1)^2}$.
	$\sum_{h=0}^{H}\alpha h 2^{H-h} \leq \alpha 2^H(2+\beta)$ $\sum_{h=0}^{H}\frac{h}{2^h}=O(2^H=n).$ binaires de recherche.	
Définition 25: Un arbre bir que l'étiquette étiquettes de s	Arbre binaire de recherche. naire de recherche (ABR) a des noeuds étiquettés dans un ensemble totalement ordonné de de chaque noeud interne est supérieur aux étiquettes de son sours-arbre gauche, et inférieur son sous-arbre droit. res n'est pas nécéssairement strict.	
	ge trouve à l'extrémité gauche de l'arbre.	
Initialisation	e sur la hauteur h de l'arbre. e: Pour $h=0$, un seul élément, c'est l'extrémité gauche.	
Par récurrence Initialisation Hérédité: Su Soit un arbre 1er cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin		·.
Par récurrence Initialisation Hérédité: Su Soit un arbre 1er cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit \(\mathscr{A}\) un AF Preuve: Initialisation	Pour $h = 0$, un seul élément, c'est l'extrémité gauche. upposons que la propriété est vraie pour les arbres de hauteur $h - 1$. binaire de recherche de hauteur h . e sous-arbre gauche, la racine est le minimum. a un sous-arbre gauche, le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sous-arbre. num est à l'extrémité gauche de l'arbre. e, on a bien la propriété. 27: Parcours infixe. \bigstar BR de hauteur h et P_h : «Le parcours infixe de $\mathscr A$ donne une liste triée».	
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit & un AF Preuve: Initialisation Hérédité. Su On note & le Par suppositio Le parcours in On a que tout	a: Pour $h = 0$, un seul élément, c'est l'extrémité gauche. Apposons que la propriété est vraie pour les arbres de hauteur $h = 1$. Binaire de recherche de hauteur h . Binaire de l'extrémité gauche de l'arbre. Binaire de l'extrémité gauche de l'arbre. Binaire de l'extrémité de l'extrémité gauche de l'arbre. Binaire de l'extrémité de l'extrémi	
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit \(\mathscr{A}\) un AF Preuve: Initialisation Hérédité. Su On note \(\mathscr{A}\) g le Par suppositio Le parcours ir On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e • Chaque	a: Pour $h = 0$, un seul élément, c'est l'extrémité gauche. Apposons que la propriété est vraie pour les arbres de hauteur $h - 1$. binaire de recherche de hauteur h . e sous-arbre gauche, la racine est le minimum. a un sous-arbre gauche, le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sous-arbre. hum est à l'extrémité gauche de l'arbre. e, on a bien la propriété. 7: Parcours infixe. ★ BR de hauteur h et P_h : «Le parcours infixe de \mathcal{A} donne une liste triée». 1. Pour $h = 0$ c'est trivial car l'arbre est réduit à sa racine. Apposons P_h vrai pour toute hauteur strictement inférieure à h . Montrons P_h . e fils gauche de \mathcal{A} , \mathcal{A}_d son fils droit et r sa racine. In, le parcours est correct sur \mathcal{A}_g et \mathcal{A}_d car ils sont de hauteurs strictement inférieures à h . Infixe parcourt d'abord \mathcal{A}_g , puis r , puis \mathcal{A}_g . elément de \mathcal{A}_g est inférieur à r et que tout élément de \mathcal{A}_d est supérieur à r par propriété des urs de \mathcal{A}_g puis r puis \mathcal{A}_d est dans l'ordre croissant.	ABR.
Par récurrence Initialisation Hérédité: Su Soit un arbre 1er cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit \(\mathscr{A}\) un AF Preuve: Initialisation Hérédité. Su On note \(\mathscr{A}\) g le Par suppositio Le parcours in On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e	n: Pour h = 0, un seul élément, c'est l'extrémité gauche. upposons que la propriété est vraie pour les arbres de hauteur h − 1. binaire de recherche de hauteur h. e sous-arbre gauche, la racine est le minimum. a un sous-arbre gauche, le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sous-arbre. num est à l'extrémité gauche de l'arbre. e, on a bien la propriété. 17: Parcours infixe. ★ BR de hauteur h et P _h : «Le parcours infixe de A donne une liste triée». 18. Pour h = 0 c'est trivial car l'arbre est réduit à sa racine. puposons P _h vrai pour toute hauteur strictement inférieure à h. Montrons P _h . e fils gauche de A, A _d son fils droit et r sa racine. on, le parcours est correct sur A _g et A _d car ils sont de hauteurs strictement inférieures à h. effixe parcourt d'abord A _g , puis r, puis A _g . élément de A _g est inférieur à r et que tout élément de A _d est supérieur à r par propriété des urs de A _g puis r puis A _d est dans l'ordre croissant. rouge-noir. ions. Arbre rouge-noir. ige-noir (ARN) est un ABR strict dont toutes les feuilles sont vides, dont chaque noeud est en noir, tel que : feuille est noire.	ABR.
Par récurrence Initialisation Hérédité: Su Soit un arbre 1er cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit Initialisation Hérédité. Su On note Je le parcours in On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e Chaque La racin Les fils i Tout che Preuve : Proposition 29: La hauteur re Preuve :	in Pour $h=0$, un seul élément, c'est l'extrémité gauche. pposons que la propriété est vraie pour les arbres de hauteur $h-1$. binaire de recherche de hauteur h . e sous-arbre gauche, la racine est le minimum. a un sous-arbre gauche, le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sous-arbre. un est à l'extrémité gauche de ce sous-arbre. e, on a bien la propriété. 77: Parcours infixe. \star 8R de hauteur h et P_h : «Le parcours infixe de \mathcal{A} donne une liste triée». 18. Pour $h=0$ c'est trivial car l'arbre est réduit à sa racine. upposons P_h vrai pour toute hauteur strictement inférieure à h . Montrons P_h . fils gauche de \mathcal{A} , \mathcal{A}_d son fils droit et r sa racine. P_h , le parcours est correct sur \mathcal{A}_g et \mathcal{A}_d car lis sont de hauteurs strictement inférieures à h . diffixe parcourt d'abord \mathcal{A}_g , puis r , puis \mathcal{A}_g est inférieur à r et que tout élément de \mathcal{A}_d est supérieur à r par propriété des urs de \mathcal{A}_g puis r puis \mathcal{A}_d est dans l'ordre croissant. 19. Arbre rouge-noir. igne-noir (ARN) est un ABR strict dont toutes les feuilles sont vides, dont chaque noeud est en noir, tel que : feuille est noire. est noire. est noire la racine à une feuille contient le même nombre de noeuds noirs. 10. Hauteur noire. noire d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds noirs.	ABR.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Initialisation Hérédité. Su On note Je le parcours in On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e Chaque Les fils n Les fils n Tout che Preuve: Montrons qu'n Initialisation Hérédité. Su On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e Chaque La racin Les fils n Tout che Preuve: Montrons qu'n Initialisation Hérédité. Su Les deux fils o Donc A a au Alors k ≤ logge	ix Pour h = 0, un seul élément, c'est l'extrémité gauche. pposons que la propriété est vraie pour les arbres de hauteur h − 1. binaire de recherche de hauteur h. e sous-arbre gauche, la racine est le minimum. a un sous-arbre gauche, le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sous-arbre. aum est à l'extrémité gauche de l'arbre. e, on a bien la propriété. 77: Parcours infixe. BR de hauteur h et P _h : «Le parcours infixe de ad donne une liste triées. 18. Parcours infixe. BR de hauteur h et P _h : «Le parcours infixe de ad donne une liste triées. 19. Pour h = 0 c'est trivial car l'arbre est réduit à sa racine. pposons P _h viai pour toute hauteur strictement inférieure à h. Montrons P _h . fils gauche de A, Ad son fils droit et r sa racine. n. le parcours est correct sur Ag et Ad, car ils sont de hauteurs strictement inférieures à h. fifixe parcourt d'abord Ag, puis r, puis Ag. elément de Ag est inférieur à r et que tout élément de Ag est supérieur à r par propriété des urs de Ag, puis r puis Ag est dans l'ordre croissant. rouge-noir. lons. Arbre rouge-noir. loge-noir (ARN) est un ABR strict dont toutes les feuilles sont vides, dont chaque noeud est en noir, tel que : feuille est noire. e est noire. couges d'un noeud noir sont noirs. en noir, tel que : feuille est noire. e est noire. ouge-noir de hauteur het de taille n, on a h ≤ 2 log ₂ (n + 1). In ARN de hauteur noire k a au moins 2 ^k − 1 noeuds. 10. Pour k = 0, un arbre de hauteur noire 0 est réduit à sa racine, donc un seul noeud. pposons la propriété vraie pour tout arbre de hauteur noire inférieure à k. Montrons le pour le la racine sont donc de hauteur noire k : ils ont au moins 2 ^k − 1 noeuds. (n + 1) et h ≤ 2k donc h ≤ 2 log ₂ (n + 1)	ABR.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas de 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit \mathscr{A} un AF Preuve: Initialisation Hérédité. Su On note \mathscr{A}_g le Par supposition Le parcours in On a que tout Donc le parcours in On a que tout Donc le parcours in On a fue tout Donc le parcours in On a fue tout Donc le parcours in On a que tout Donc le parcours in On a que tout Donc le parcours in On a que tout Donc le parcours in Les fils in Eas fils in Eas fils in Tout che and I and	n: Pour h = 0, un seuf élément, c'est l'extrémité gauche. pposons que la propriété est vraie pour les arbres de hauteur h − 1. binaire de rederche de hauteur h. sons-arbre gauche, la racine est le minimum. a un sous-arbre gauche, la racine est le minimum. a un sous-arbre gauche, le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sous-arbre. num est à l'extrémité gauche de l'arbre. e, on a bien la propriété. 77: Parcours infixe. ★ Bit de hauteur h et Ph: «Le parcours infixe de d' donne une liste triée». 18. Pour h = 0 c'est trivial car l'arbre est réduit à sa racine. pposons Ph vrai pour toute hauteur strictement inférieure à h. Montrons Ph. fils gauche de d', sd' son fils droit et r sa racine. n, le parcours est correct sur sd', et d', car ils sont de hauteurs strictement inférieures à h. filse parcourt d'à bord sd', su pis r, puis sd', elément de d', g est inférieur à r et que tout élément de d', est supérieur à r par propriété des urs de d', puis r puis sd', est dans l'ordre croissant. rouge-noir. ions. Arbre rouge-noir. ions. Arbre rouge-noir. ions. Arbre rouge-noir. ione d'un noeud noir sont noirs. entin de la racine à une feuille contient le même nombre de noeuds noirs. Hauteur noire. ione d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds noirs. Hauteur noire. noire d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds noirs. Hauteur noire. noire d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds noirs.	ABR.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas de 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit \mathscr{A} un AF Preuve: Initialisation Hérédité. Su On note \mathscr{A}_g le Par supposition 2 2.4 Arbres 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rouse nouge, ou en couge, ou en couge, ou en couge, ou en rouge,	is Pour h = 0, un seul défement, c'est l'extrémité gauche. proposons que la propriété et vraie pour les arbres de hauteur h − 1. binaire de recherche de hauteur h. a un sous arbre gauche, la racine est le minimum. a un sous arbre gauche, la racine est le minimum. a un sous arbre gauche, la racine est le minimum. a un sous arbre gauche, le minimum y appartient, or ce sous arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sous-arbre. un est à l'extrémité gauche de l'arbre. 7. Parcours infixe. ★ BR de hauteur h et P _h : «Le parcours infixe de s' donne une liste triées. 1. Pour h = 0 c'est trivial car l'arbre est réduit à sa racine. ppossas P _h vai pour toute banteur strictement inférieure à h. Montrons P _h . fils gauche de s', sd son fis fortie et r sa racine. n, le parcours est correct sur s' _t et s' _t car ils sont de hauteurs strictement inférieure à h. fils parcourt d'abord s' _t , puis s' _t , elément de s' _t est inférieur à n et que tout élément de s' _t est supérieur à n par propriété des mes de s' _t guis r puis s' _t est dans l'ordre croissant. **rouge-noir** fons. **Arbre rouge-noir** ions. **Arbre rouge-noir** ions. **Arbre rouge-noir* ions. **Arbre rouge-noir* ions d'un noud noir sont noirs. amin de la racine à une feuille contient le même nombre de noeuds noirs. **Hauteur noire** our d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de ha racine à une feuille contient le même nombre de noeuds noirs. **Hauteur noire** our d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de ha racine à une for cest noire d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de ha racine à une for cest noire d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de ha racine à une for cest noire d'un arbre rouge-noir de hauteur noire k a un moins 2 - 1 noeuds. 1. Pour k = 0, un arbre de hauteur noire k is lis ont an moins 2 - 1 noeuds. 1. (n + 1) et h ≤ 2k done h ≤ 2 log ₂ (n + 1) ** un des arbres binaires.	ABR. $k+1$.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas de 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit \mathscr{A} un AF Preuve: Initialisation Hérédité. Su On note \mathscr{A}_g le Par supposition 1. Le parcours in On a que tout Donc le parcours in On a que tout Definition 28: 1.4.1 Définition 28: 1.5.1 Les fils in Tout che les fils in Théorème 31: 1.5.2 Applica Théorème 31: 1.5.3 Applica Théorème 32: 2.5 Applica Théorème 32: 2.5 Applica Théorème 32: 2.6 Applica Théorème 32: 2.7 Applica Théorème 32: 2.8 Applica Théorème 32: 2.9 Control on Théorème 32: 2.1 Applica Théorème 32: 2.1 Applica Théorème 32: 2.2 Applica Théorème 32: 2.3 Applica Théorème 32: 2.4 Arbres Les fils in Théorème 32: 2.5 Applica Théorème 32: 2.6 Applica Théorème 32: 2.7 Applica Théorème 32: 2.8 Applica Théorème 32: 2.9 Applica Théorème 32: 2.1 Applica Théorème 32: 2.1 Applica Théorème 32: 2.2 Applica Théorème 32: 2.3 Applica Théorème 32: 2.4 Arbres Théorème 32: 2.5 Applica Théorème 32: 2.6 Applica Théorème 32: 2.7 Applica Théorème 32: 2.8 Applica Théorème 32: 2.9 Applica Théorème 32: 2.1 Applica Théorème 32: 2.1 Applica Théorème 32: 2.2 Applica Théorème 32: 2.3 Applica Théorème 32: 2.4 Arbres Théorème 34: 2.5 Applica Théorème 34: 2.6 Arbres Théorème 34: 2.7 Arbres Théorème 34: 2.8 Arbres Théorème 34: 2.9 Arbres Théorème 34: 2.0 Arbres Théorème 34: 2.1 Arbres Théorème 34: 2.2 Arbres Théorème 34: 2.3 Arbres Théorème 34: 2.4 Arbres Théorème 34: 2.5 Applica Théorème 34: 2.6 Arbres Théorème 34: 2.7 Arbres Théorème 34: 2.8 Arbres Théorème 34: 2.9 Arbres Théorème 34: 2.0 Arbres Théorème 34: 2.1 Arbres Théorème 34: 2.1 Arbres Thé	is Pour h = 0, un seul défenent, c'est Pearfointé gauche. proposons que la propriété est vaie pour les arbres de hauteur h − 1. binaire de recherche de hauteur h. sous-arbre gauche, la racine est le minimum. a un sons-arbre gauche, la racine est le minimum. a un sons-arbre gauche, de unimum y appartient, or ce sons-arbre est de hauteur inférieure le minimum est à l'extrémité gauche de ce sons-arbre. um est à l'extrémité gauche de l'arbre. e, on a bien la propriété. 7: Parcours infixe. ★ BR de hauteur h et P _b : «Le parcours infixe de A donne une liste triées. 1. Pour h = 0 e'est trivial car l'arbre est réduit à sa racine. ppessons l'h, vroi pour toute hauteur strictement inférieure à h. Montrons l'h. tils gauche de A', A'g son fis droit et ra arcine. n, le parcours est correct sur A'g et A'g car ils sont de hauteurs strictement inférieure à h. diac parcourt d'abord A'g, puis r, puis A'g. et dans l'ordre croissant. rouge-noir puis A'g est dans l'ordre croissant. rouge-noir. ions. Arbre rouge-noir. ions. Arbre rouge-noir. ge-noir (ARR) est un ABR strict dont toutes les feuilles sont vides, dont chaque noeud est en roir, let que. et st noire. et st noire. et st noire. et toure. ione d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds moirs. Hauteur noire. ione d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds moirs. Hauteur noire. ione d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds moirs. Hauteur noire. ionire d'un arbre rouge-noir est le nombre de noeuds noirs sur un chemin de la racine à une feuille contient le même nombre de noeuds moirs le pour le la racine sont donc de hauteur noire v.: ils ont au moins 2 ^k − 1 noeuds. n. Pour k = 0, un arbre de hauteur noire v.: ils ont au moins 2 ^k − 1 noeuds. (a) H = 1 + 1 + 2 ^{k+1} − 1 - 2 ^{k+1} − 1 noeuds.	ABR. $k+1$.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas de le le cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit \mathscr{A} un AF Preuve: Initialisation Hérédité. Su On note \mathscr{A}_g le Par supposition 2 2.4 Arbres 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rouge, ou ele Chaque La racin el Les fils results	 if Pour h = 0, un seul élement, e'est l'extremité gauche. proposons que la propriété est visage pour les arbres de hauteur h = 1. binaire de recherche de hauteur h. a un sois-arbre gauche, le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum et l'extrémité gauche de le minimum y appartient, or ce sous-arbre est de hauteur inférieure le minimum et l'extrémité gauche de l'arbre. c, on a bien la propriété. 77. Parcours infixe. ★ Bit de hauteur h et h'_b : « le parcours infixe de s' donne une fiste triées. In Pour h = 0 c'est trivial car l'arbre est yédnit à sa racine. proposos P, vini pour toute hauteur strictement inférieure à h. Montrons P_b. in. Pour h = 0 c'est trivial car l'arbre est yédnit à sa racine. proposos P, vini pour toute hauteur strictement inférieure à h. Montrons P_b. in. Pour s'est course sur s'est est s'est s'est de la mateur strictement inférieure à h. Montrons P_b. in. Pour s'est course sur s'est est s'est s'est de la mateur strictement inférieure à h. Montrons P_b. in. Pour s'est course sur s'est est s'est s'est de la mateur strictement inférieure à h. Montrons P_b. in. Pour s'est course sur s'est est s'est s'est de la mateur strictement inférieure à h. pur propriété des ura de s'_d puis r pais s'_d est dans l'ordre croissant. rouge-noir. ions. Arbre rouge-noir. ions. Arbre rouge-noir. ions. Arbre rouge-noir. ions. Arbre rouge-noir. ions. ions. Arbre rouge-noir. ions. ions de nacteur à et de taille contient le même nombre de noeuda noir	ABR. $k+1$.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas de 2ème cas: Il y Par hyothèsel, Donc le minim Par récurrence le soit \mathscr{A} un AF Preuve: Initialisation Hérédité. Su On note \mathscr{A}_g le Par supposition 28: On note \mathscr{A}_g le Par supposition 29: Le parcours in On a que tout Donc le parcours in Eas fils in Tout che le soit un arbre le les fils in Tout che le	 10 m h = 0, un sent blement, riest Destroinie gauche. 11 ministe de techerche de hauteur h. 12 m ous arbre gauche, le missimum y appariente, no ce rous arbre set de hauteur inférieure le unimismum et à l'extrémit quache de ce sous-arbre. 12 m ous arbre gauche, le missimum y appariente, no ce rous arbre set de hauteur inférieure le unimismum et à l'extrémit quache de ce sous-arbre. 13 m ous arbre gauche, le missimum y appariente, no ce rous arbre set de hauteur inférieure le unimismum et à l'extrémit quache de ce sous-arbre. 14 m ou le la propriété. 15 Parcours infixe. ★ 16 R de hauteur h et P_k: ster parcours infixe de af donne une liste trices. 16 Parcours infixe. ★ 17 Parcours infixe. ★ 18 R de hauteur h et P_k: ster parcours infixe de af donne une liste trices. 19 Parcours infixe. ★ 18 R de hauteur h et P_k: ster parcours infixe de af donne une liste trices. 19 Parcours infixe. ★ 19 Parcours infixe. ★ 10 Parcours infixe. ★ 10 Parcours infixe. ★ 10 Parcours infixe. ★ 11 Parcours infixe. ★ 12 Parcours infixe. ★ 12 Parcours infixe. ★ 12 Parcours infixe. ★ 13 Parcours infixe. ★ 14 Parcours infixe. ★ 14 Parcours infixe. ★ 15 Parcours infixe. ★ 16 Parcours infixe. ★ 16 Parcours infixe. ★ 17 Parcours infixe. ★ 18 Parcours infixe. ★ 19 Parcours infixe. ★ 19 Parcours infixe. ★ 10 Parcours infixe. ★ 10 Parcours infixe. ★ 10 Parcours infixe. ★ 11 Parcours infixe. ★ 12 Parcours infixe. ★ 13 Parcours infixe. ★ 14 Parcours infixe. ★ 15 Parcours infixe. ★ 16 Parcours infixe. ★ 16 Par	ABR. $k+1$.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas de 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence Proposition 2 Soit ω un AF Preuve: Initialisation Hérédité. Su On note ω gle Par suppositic Le parcours in On a que tout Donc le parcours in Earlie Initialisation Hérédité. Su Les fils in Tout che Initialisation Hérédité. Su Les deux fils conc A a aux Alors k ≤ log. 2.4.2 Implémentation Preuve: Montrons qu'n Initialisation Hérédité. Su Les deux fils conc A a aux Alors k ≤ log. 2.5 Applica Théorème 31 Implémentation Preuve: L'action de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information de l'a de ses élément feuille si le tri On note h la information d	is Pour $k=0$, un sent défennen, c'est l'extrémité gauche. proposeus que la propriét rel vaire que le native de hauteur $h=1$, bindré de recherche de auteure h . son andré gauche, le minime est le minimum. a un sons-arbre gauche, le minime six le minimum. a un sons-arbre gauche, le minimum y appareient, or ce sons-arbre est de hauteur inférience le minimum est à fearthrinte gauche de l'arbre. on a bien la propriéte. 7: Parcours infixe. ** 16: Parcours infixe. ** 17: Parcours infixe. ** 18: de hauteur h et P : 4 de percours infixe de a^{\prime} donne une liste tribes. 19: Parcours infixe. ** 19: Parcours infixe. ** 10: Parcours infixe. 10: Parcours infixe. ** 10	ABR. $k+1$.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Soit Initialisation Hérédité. Su On note Par supposition Le parcour Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e	is thur $h = 0$, on seal different x -real positions of gaussian proposes up the largeridide started part is active at the statem $h = 1$, behinder de rectarche de hauteur $h = 1$, and so substitute quality of the content of the statem $h = 1$ indicated gaussian	ABR. $k+1$.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas de 2ème cas: Il y Par hyothèse, Donc le minim Par récurrence de soit of un AF Preuve: Initialisation Hérédité. Su On note of le parcours in On a que tout Donc le parcours in Earlie Définition 28: Un arbre rout en rouge, ou en roug	 From N = 0, on seed distance, c'est l'externation grounds. proposes que la propriété des varia pour les arbors de l'autre A = 1 friedre de vortente de hauteur A = 1 friedre de vortente de l'autre de vortente est d'he friedre goulde de l'autre constitute. 7. Parcours influe. ★ 18. de hauteur A et A = 1 de partours influe de al dome une flot critice. 7. Parcours influe. ★ 18. de hauteur A et A = 1 de partours influe de al dome une flot critice. 3. Pour A = 0 c'est trivial est l'altre est poulle à est sociale. processe A = 0 c'est trivial est l'altre est poulle de se sociale de hauteurs strictement autérieure à l'apprende de la freque de de l'apprende de l'ap	ABR. $k+1$.
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Initialisation Hérédité. Su On note Par suppositic Le parcours in On a que tout Donc le parco 2.4 Arbres 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e	 Four he = 0, on and démant, et an Tourtande packet Infinite de reversale de touter de l'Antonia de reversale de l'Antonia de l'An	ABR. $k+1$.
Par récurrence Initialisation Hérédité: St Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Initialisation Hérédité. St On note Par suppositic Le parcours in On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e	 From A — On an earl Schment. A See "Ferning September 1. In the processing the properties are variety per the arbitrar do handman A — In the first or the variety of the function of the variety of the first of the properties. A manufacture of a Ferning September 1. In the control of the first of the fir	ABR. coloré euille.
Pair récurrence Initialisation Hérédité: St. Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Initialisation Hérédité. St. On note Par suppositic Le parcours in On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou Chaque Les fils in Tout che Preuve: Montrons qu'n Initialisation Hérédité. St. Les deux fils c Donc A a au Alors k ≤ log; 2.4.2 Implém Théorème 31 Implémentation Preuve: Montrons qu'n Initialisation Hérédité. St. Les deux fils c Donc A a au Alors k ≤ log; 2.4.2 Implém Théorème 31 Implémentation Algorithme Soit un algori est en Ω(n log 2.5 Applica Théorème 31 Implémentation Algorithme Théorème 31 Implémentation Algorithme Entrées: Sorties: construir al pu é chang fin chang (fin fun echange (fin Complexité: Sorties: sorties: sorties: construir al pu é chang fin fin echange (fin Entrées: Sorties: sorties: si debut < li>linf fin si inf si inf chechange (fin Complexité: Les apperlain Les boucles we externe se ter Alagorithme Entrées: Sorties: so	 It form b = 0, an and observation of the contract of	ABR. coloré euille. k+1. while while
Par récurrence Initalisation Hérédité: Sh Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Soit Initalisation Hérédité. Sh Con note Par suppositi Le parcours in On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e e Chaque e La racin e Les fils i en Tout che Définition 29: La hauteur re Preuve: Montrons qu' Initialisation Hérédité. Sh Les deux fils o Donc A a au : Alors k ≤ log; 2.4.2 Implém Théorème 31 Implémentation Preuve: L'action de l', de ses élément feuille si le tri On note h la : Alors k ≤ log; 2.4.2 Implém Théorème 31 Implémentation Algorithm Entrées: Sorties: construire pour étalle; de percole fin Complexité: Sorties: construire pour étalle; de percole fin Con en conclut Définition 34: Algorithm Entrées: Sorties: construire pour étalle; de percole fin Complexité: Sorties: construire pour étalle; de percole fin Con en conclut Définition 34: Algorithm Entrées: Sorties: construire pour étalle; de percole fin Con plexité: Sorties: construire pour étalle; de percole fin Con plexité: Sorties: construire pour étalle; de percole fin Con plexité: Sorties: construire pour étalle; de percole fin Con echonger(') reny tri raj fin Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle; con echanger(') reny externe se ter. Algorithm Entrées: Sorties: construire pour étalle	a None A - On a word element. An international grounder. The procurage of largeriche to wave pour his desirations. A man assessmelt prographic to a state of the plantitume. A man assessmelt prographic to a man may appropriate to one accessorate prost for leaders in information in administration of a Procurage information of a contract of the prographic to a man may appropriate to the prographic to a prographic to the p	ABR. coloré euille. k+1. while while
Par récurrence Par récurrence Hérédité: Su Soit un arbre 1 er cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Freuve: Initialisation Hérédité. Su On note Par suppositi Le parcours in On a que tout Donc le parcou 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou en rouge,	is found to 1. Cut a most definions of which is the control of models of the control of the processor and to provide the visible point of the balliations. The processor of the control of product for the control of	ABR. coloré euille. k+1. while adices
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Initialisation Hérédité. Su On note Par suppositi Le parcours in On a que tout Donc le parcou 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou en rouge, ou en chaque Les fils in Tout che Entréedité. Su Les deux fils in On a du tout Initialisation Hérédité. Su Les deux fils in Tout che Preuve: Montrons qu'n Initialisation Hérédité. Su Les deux fils in Donc A a a un Alors k ≤ log; 2.4.2 Implén Théorème 31 Implémentation Preuve: L'action de l'ade seus élément foun rous le change de sus elément foun rous le change in fin Entrées: Sorties: Construire pour in la did L'échange des La boucle for On en conclut Définition 34: Algorithm Entrées: Sorties: Construire pour in la did L'échange des La boucle for On en déduit Définition 34: Algorithm Entrées: Sorties: Construire pour in la did L'échange des L'éc	where $L = 0$, one of delineary design to study in the form of a particle present goal by profession study in particle of the particle present of a particle present of the p	ABR. coloré euille. k+1. while adices
Par récurrence Hatelisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Soit Preuve: Initialisation Hérédité. Su On note Par suppositi Le parcours in On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou en rouge,	In the control of the control of the control of particles of the particle of the control of the	ABR. coloré euille. k+1. while adices
Proposition 2 Proposition 2 Soit with a sign and sign a	If the the real is an another contract of the	ABR. coloré euille. k+1. while adices
Par récurrence Initialisation Hérédité: Su Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Soit a un AF Preuve: Initialisation Hérédité. Su On note ag le Par suppositic Le parcours in On a que to to Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou en Chaque • La racin • Les fils is • Tout ch Théorème 31 Implémentation Preuve: Montrons qu' Intialistion Hérédité. Su Onn A a au Alors k ≤ log; 2.4.2 Implém Théorème 31 Implémentation Preuve: Montrons qu' Intialistion Hérédité. Su Con a que to to pour a la le decay fils Donc A a au Alors k ≤ log; 2.4.2 Implémentation Preuve: L'action de l'end feuille ri Con note h la le decay fils Connote h la le change des La boucle for On en déduit Définition 34: Algorithm Entrées: Sorties: sorties: sorties: la boucle for On en déduit Définition 34: Algorithm Entrées: Sorties: l'end fin	where the control of	ABR. coloré euille. k+1. while adices
Par récurrence Initialisation Hérédité: St Soit un arbre ler cas: Pas d 2ème cas: Il y Par hyothèse, Donc le minin Par récurrence Preuve: Initialisation Hérédité. St On note Preuve: Initialisation Herédité. St On note Preuve: Initialisation On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou en Chaque • La racin • Les rils • Tout ch Définition 29: La hauteur re Preuve: Montrons qu'n Initialisation Hérédité. St Con ca a a man and	Let have $\lambda = 0$ and continuous contribution of $\lambda = 0$. The contribution of $\lambda = 0$ is a second to account of the solution of $\lambda = 0$ is a second to account of $\lambda = 0$ in the contribution of $\lambda = 0$ is a second to account of $\lambda = 0$ in the contribution of $\lambda = 0$ in the contri	ABR. coloré euille. k+1. while adices
Par récurrence Initialisation Hérédité: St. Soit un arbre ler cas: Pas il y Par hyothèse, Donc le minin Par récurrence Proposition 2 Soit Jun Al- Preuve: Initialisation Hérédité. St. On note Par supposition On a que tout Donc le parco 2.4 Arbres 2.4.1 Définit Définition 28: Un arbre rou en rouge, ou e La racin e Les fils e Tout che Preuve: Montrons qu', Initialisation Hérédité. St. La deux fils e Tout che Preuve: Montrons qu', Initialisation Hérédité. St. La deux fils e Tout che Preuve: Montrons qu', Initialisation Hérédité. St. Les deux fils e Tout che Preuve: Nontrons qu', Initialisation Hérédité. St. Les deux fils e Tout che Preuve: L'action de l', de ses élément feuiles il et ri On note h la il On en conclut Définition 34: Algorithm Entrées: Sorties: construire pour i all, échang percole fin fin Soit na la tails L'échange des La boucle for On en déduit Définition 34: Algorithm Entrées: Sorties: construire pour i all, échang percole fin fin si inf: inf: inf: inf: inf: inf: inf: inf:	From A. D. In seculation of the process of the proc	ABR. coloré euille. k+1. while adices
Par récurrence Initialisation Hérédité: Sto Soi un arbre de le manner	From A = 10, an order between the production of particles of the part of the particle of the p	ABR. coloré euille. k+1. while adices
Par récurrence Initialisation Hefétité: Sr Soit un arbe ler cas: Pay Darnothèsein, Dar récurrence Proposition 2 Soit Freuve: Initialisation Herédité. Sr On a que four Donc le parcours in La hauteur in Preuve: Montrons qu'n Initialisation Herédité. Sr Care de la de se demendent in Definition 29: La hauteur in Preuve: Montrons qu'n Initialisation Herédité. Sr Care de la de se démendent in Les diux s' ≤ log; 2.4.2 Implém Preuve: Montrons qu'n Initialisation Herédité. Sr Ces de se démendent in Les diux s' ≤ log; 2.4.2 Implém Preuve: Montrons qu'n Initialisation Les diux s' ≤ log; Alors k ≤ log; 2.4.2 Implém Preuve: Montrons qu'n Initialisation Les diux s' ≤ log; Alors k ≤ log; 2.4.2 Implém Preuve: Montrons qu'n Initialisation Les diux s' ≤ log; Alors k ≤ log; 2.4.2 Implém Preuve: Montrons qu'n Initialisation Les diux s' ≤ log; Alors k ≤ log; 2.4.2 Implém Preuve: Montrons qu'n Initialisation Les diux s' ≤ log; Alors k ≤ log; Algorithm Entrées : Sorties : pivot de de se de condent in to Complexité: Sorties : pivot de de se de condent in to Con en déduit Définition 36: Les cles son On appelle cle Les cles son On appelle cle Les cles son On appelle cle Les cles son On parle de tr Les appels à p Correction. Montrons la cle Prédiction 39: Algorithm Entrées : Sorties : pivot de de se de condent in to Condent in to Con prend in change in the condent in to Con definition 36: Les cles son On appelle cle Le meilleur de Les cles son On parle de tr Algorithm Entrées : Sorties : Les cles son On parle de tr Algorithm Entrées : Sorties : Les cles son On parle de tr Algorithm Entrées : Correction. Algorithm Ent	The first of the control of the cont	able and ices while while addices
Proposition 3 Algorithm Preuve: All Definition 33 Can rough Les ale Les ale Les ale Les ale La survey All Definition 34 Algorithm Entrées: Con rote band Les ale	The Late of the Common and allowers and the Common	able and ices while while addices