

SUDOKU

Dossier de conception et code
S1.02 - Comparaison d'approches
algorithmiques

BODIN Maximilien
JACOB-SAUSSEREAU Maxime
MARTIN Edgar
TD II - TP 4

Année universitaire 2022-2023

1. Equipe

BODIN Maximilien
JACOB-SAUSSEREAU Maxime
MARTIN Edgar
TD II – TP 4

2. Jeu du Sudoku:

TD-II_Sujet_3_1

Table des matières :

S1.02 - Comparaison d'approches algorithmiques	1
1. Equipe	2
2. Jeu du Sudoku:	2
3. Rappel des spécifications du programme	4
3.1 Spécifications initiales	4
3.2 Spécifications complémentaires = extensions traitées	4
4. Jeux d'essais	4
5. Algorithmes du programme	5
5.1 Sudoku	5
5.1.1 But de l'action	5
5.1.2 Stratégie de l'algorithme mise en œuvre et justification	5
5.1.4 Dictionnaire des éléments associés à cet algorithme	5
5.1.5 Dictionnaire des actions (complexes) associées à cet algorithme	6
5.2 initialiserLaPartie	6
5.2.1 But de l'action	6
5.2.2 Stratégie de l'algorithme mise en œuvre et justification	6
5.2.3 Algorithme	7
5.2.4 Dictionnaire des éléments associés à cet algorithme	8
5.2.5 Dictionnaire des actions associées à cet algorithme	8
5.3 jouerLaPartie	8
5.3.1 But de l'action	8
5.3.2 Stratégie de l'algorithme mise en œuvre et justification	8
5.2.3 Algorithme	9
5.3.4 Dictionnaire des éléments associés à cet algorithme	10
5.3.5 Dictionnaire des actions associées à cet algorithme	10
5.4 afficherSudoku	11
5.4.1 But de l'action	11
5.4.2 Stratégie de l'algorithme mise en œuvre et justification	11

5.4.3 Algorithme	12
5.4.4 Dictionnaire des éléments associés à cet algorithme	13
5.4.5 Dictionnaire des actions associées à cet algorithme	13
5.5 saisiVerifJoueur	13
5.5.1 But de l'action	13
5.5.2 Stratégie de l'algorithme	13
5.5.3 Algorithme	14
5.5.4 Dictionnaire des éléments associés à cet algorithme	15
5.5.5 Dictionnaire des actions associées à cet algorithme	15
5.6 rechercheDeValeurPossible	15
5.6.1 But de l'action	15
5.6.2 Stratégie de l'algorithme	15
5.6.3 Algorithme	16
5.6.4 Dictionnaire des éléments associés à cet algorithme	17
5.6.5 Dictionnaire des actions associées à cet algorithme	17
5.7 verifValeur	17
5.7.1 But de l'action	17
5.7.2 Stratégie de l'algorithme	17
5.7.3 Algorithme	18
5.7.4 Dictionnaire des éléments associés à cet algorithme	19
5.7.5 Dictionnaire des actions associées à cet algorithme	19
5.8 tabPlein	19
5.8.1 But de l'action	19
5.8.2 Stratégie de l'algorithme	19
5.8.3 Algorithme	20
5.8.4 Dictionnaire des éléments associés à cet algorithme	21
5.8.5 Dictionnaire des actions associées à cet algorithme	21
5.9 Finaliser la partie	21
5.9.1 But de l'action	21
5.9.2 Stratégie de l'algorithme	21
5.9.3 Algorithme	22
5.9.4 Dictionnaire des éléments associés à cet algorithme	23
5.9.5 Dictionnaire des actions associées à cet algorithme	23
6. Tests réalisés et traces d'exécution	23
7. Remarques	24
8. Code C++	24
9. Annexe 1 – maquettes d'écran prévues dans les spécifications	24
10. Annexe 2 – Traces d'exécution - copies d'écran	24
11. Annexe 3 – Précisions	24

3. Rappel des spécifications du programme

3.1 Spécifications initiales

Comportement des différents scénarios du programme:

- Valeur saisie compatible avec la grille du sudoku :
 - Le programme affiche une validation textuelle puis affiche le nombre sur la grille et précise si celui-ci à été modifié dans le cas où il l'a été.
- Valeur saisie incompatible avec la grille du sudoku :
 - Le programme affiche une erreur et
 - affiche les propositions possibles que l'utilisateur peut entrer.
 - et si l'utilisateur à dépasser son nombre d'erreurs disponibles et le programme lui indique la fin de la partie.
 - Le programme affiche un message légèrement différent si la saisie du joueur à tenter de modifier la grille de départ.
- Erreur dans la saisie :
 - Le programme affiche un message d'erreur mais n'en prend pas compte dans le nombre d'erreurs.
- Abandon :
 - Dans le cas où la saisie est 0 0 0 le joueur indique un abandon le programme affiche alors une confirmation visuel puis ré-affiche la grille sans les chiffres ajoutées par l'utilisateur.
- La grille est remplie :
 - Le programme affiche un message de félicitation et recommence une partie

3.2 Spécifications complémentaires = extensions traitées

Nous avons décidé de traiter le cas d'erreur de la saisie du nombre d'erreur autorisé (cf **5.2 Initialiser la partie**) et pour cela nous avons décidé d'afficher un message demandant la saisie du joueur dans un premier temp puis d'afficher un message d'erreur si la saisie ne convient pas et de recommencer ainsi jusqu'à que la saisie convienne

4. Jeux d'essais

Test programme SUDOKU					
Donnés			Résultat		
indiceLigne	indiceCollone	valeurSaisie	issueDeLaSaisie	modifier	affichage
1	3	1	compatible	faux	cf. Annexe_1 maquette_écran_compatible
8	5	4	incompatible	faux	cf. Annexe_1 maquette_écran_incompatible
1	3	2	compatible	vrai	cf. Annexe_1 maquette_écran_compatible_changement_de_valeur
1	1	6	incompatible	vrai	cf. Annexe_1 maquette_écran_incompatible_changement_de_valeur
1	z	c	erreurDeSaisie	faux	cf. Annexe_1 maquette_écran_erreur_saisie
0	0	0	abandon	faux	cf. Annexe_1 maquette_écran_abandon

Tous les scénarios ne sont ici pas pris en compte puisque géré différemment tels que lorsque la grille est remplie ou que l'utilisateur a dépassé son nombre de fautes.

5. Algorithmes du programme

5.1 Sudoku

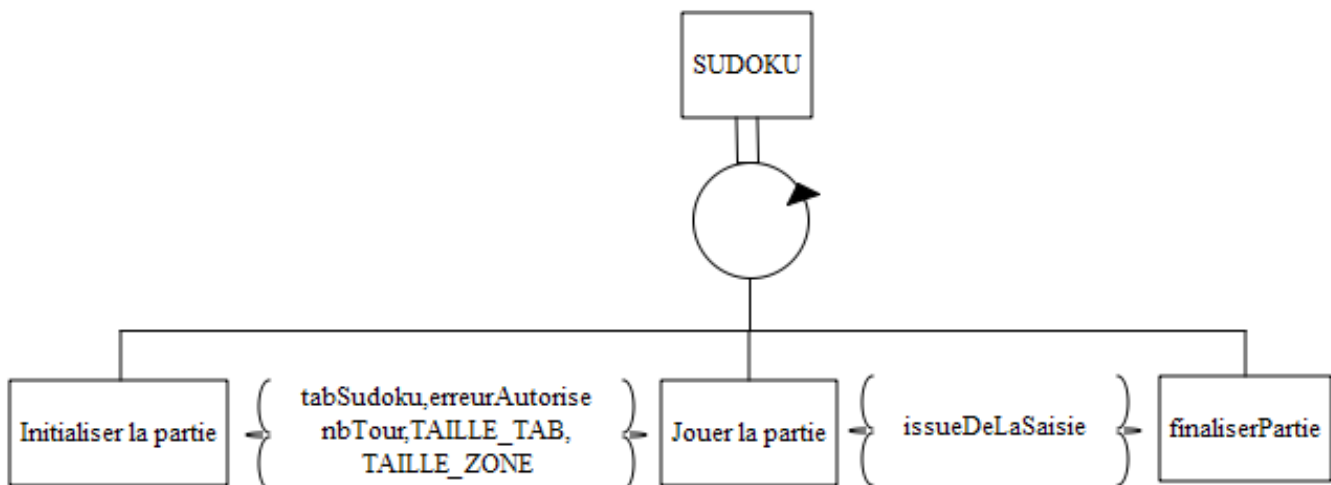
5.1.1 But de l'action

Jouer au Sudoku.

5.1.2 Stratégie de l'algorithme mise en œuvre et justification

Cet algorithme propose une décomposition en 3 parties : "Initialiser la partie", "Jouer la partie", "Finaliser la partie". qui une approche classique dans un jeu puisqu'elle facilite l'implémentation de la boucle de jeu, ici dans: "Jouer la partie". Chaque partie communique directement entre elles.

5.1.3 Algorithme



5.1.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
tabSudoku	Tableau	tableau de nbSudoku, il contient toutes les valeurs du sudoku et est actualisé avec la saisie du joueur en cas de compatibilité.
erreureAutorise	Entier positif court	Le nombre d'erreur autorisée avant la fin de la partie (≥ 3)
nbTour	Entier positif court	Le nombre de tour
TAILLE_TAB	Entier positif court	La taille de la grille du sudoku défini à 9

TAILLE_ZONE	Entier positif court	La taille de chaque zone défini à 3
issueDeLaSaisie	énuméré	Défini chaque situations; {compatible, incompatible, abandon, erreur de saisie}

5.1.5 Dictionnaire des actions (complexes) associées à cet algorithme

Nom Action	But	Données	Résultats
Initialiser la partie	initialise toutes les variables qui doivent l'être.	clavier (pour le nombre d'erreurAutorise)	tabSudoku, erreurAutorise, nbTour, TAILLE_TAB, TAILLE_ZONE
Jouer la partie	jouer au sudoku	tabSudoku, erreurAutorise, nbTour, TAILLE_TAB, TAILLE_ZONE	issueDeLaSaisie
Finaliser la partie	affiche le message de fin de partie	issueDeLaSaisie	affichage

5.2 initialiserLaPartie

5.2.1 But de l'action

Initialiser le nombre d'erreur autorisé, le tableau du sudoku, les variables de tailles, TAILLE_TAB et TAILLE_ZONE et le nombre de tour nbTour

5.2.2 Stratégie de l'algorithme mise en œuvre et justification

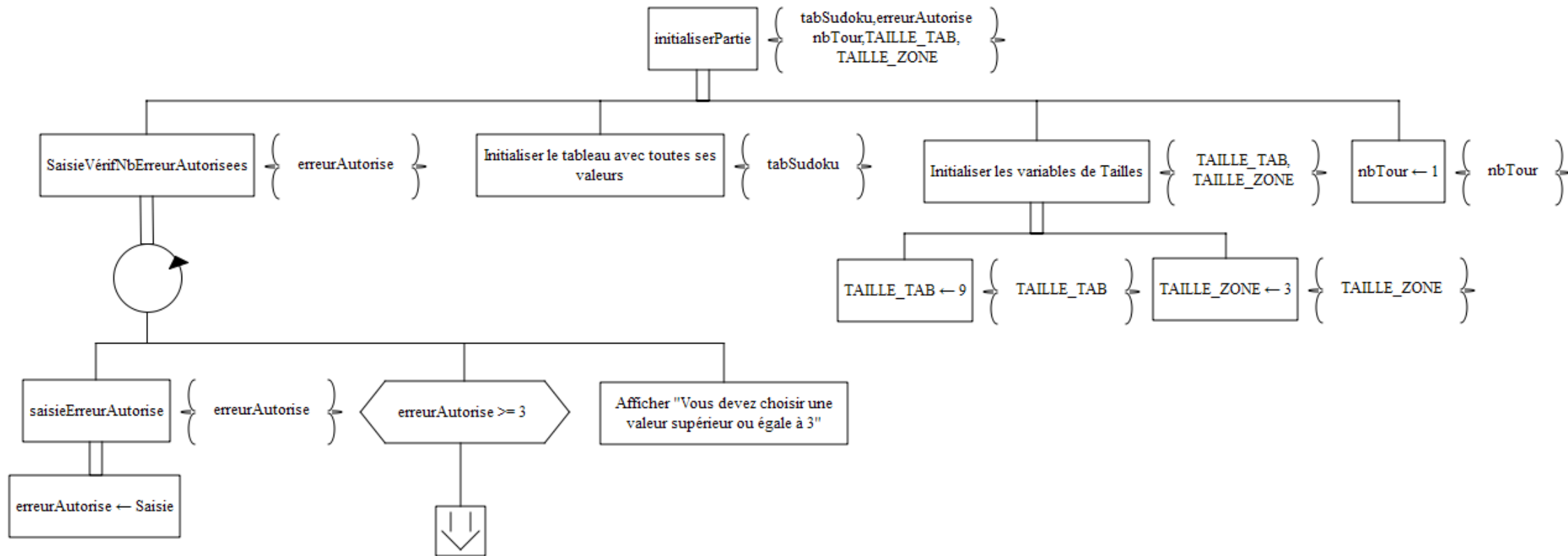
On effectue d'abord une "saisie vérif" pour laisser au joueur le choix du nombre d'erreurs autorisées tout en le limitant à choisir une valeur supérieure ou égale à 3. Puis on initialise le tableau cela dépend énormément du langage utilisé, mais la majorité des langages permet d'initialiser un tableau dès sa déclaration,

Précision capitale :

Il faut noter que le Tableau a 91 cases, qui sont elles même remplies chacune de 2 informations : la **valeur de la case** et si ce nombre fait partie de la **grille de départ** du sudoku.

Et en dernier on initialise TAILLE_TAB et TAILLE_ZONE qui sont comme indiqué par leur nom des variable et nbTour qu'on initialise à 1 et qui sera incrémenté de 1 à chaque coup du joueur excepté dans le cas d'une erreur de saisie voir [5.3 JouerLaPartie](#) .

5.2.3 Algorithme



5.2.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
tabSudoku	Tableau	tableau de nbSudoku, il contient toutes les valeurs du sudoku et actualisé avec la saisie du joueur en cas de compatibilité
erreureAutorise	Entier positif court	Le nombre d'erreur autorisée avant la fin de la partie (≥ 3)
nbTour	Entier positif court	Le nombre de tour
TAILLE_TAB	Entier positif court	La taille de la grille du sudoku défini à 9
TAILLE_ZONE	Entier positif court	La taille de chaque zone défini à 3

5.2.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
saisie Vérif bErreur Autorisees	Saisie vérif de la saisie par le joueur du nombre d'erreur qui lui sera accordées pour cette manche	aucune	erreurAutorise
Initialiser le tableau avec toutes ses valeurs	Comme expliqué dans la partie 5.2.2, le tableau est initialisé avec toute ses valeurs	aucune	tabSudoku
Initialiser les variables de Tailles	Initialiser le valeurs TAILLE_TAB à 3 et TAILLE_ZONE à 9	aucune	TAILLE_TAB , TAILLE_ZONE

5.3 jouerLaPartie

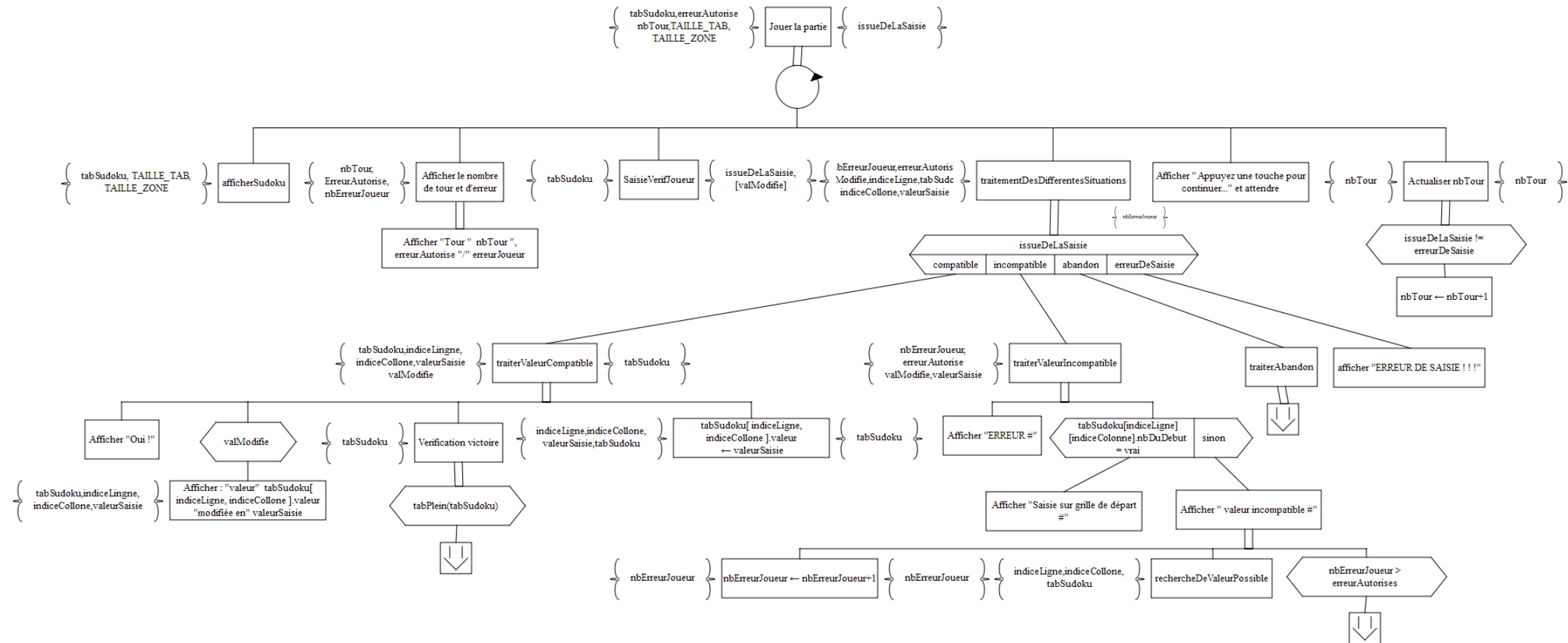
5.3.1 But de l'action

jouerLaPartie est l'algorithme centrale qui contient la boucle de jeu c.a.d la boucle qui vas afficher le jeu et prendre en compte les actions du joueur

5.3.2 Stratégie de l'algorithme mise en œuvre et justification

Pour comprendre la stratégie utilisée ici il est vivement conseillé d'avoir lu l'annexe 1 avant, puisque la plupart des traitements interne se trouve dans les autres algorithmes, ici on voit surtout la partie affichage et répartition des tâches que le programme doit réaliser. La partie en arbre et celle qui gère toutes les situations et c'est dans les 3 situations (Abandon, Victoire, Défaite) qui font sortir de la boucle que la partie se termine et c'est ensuite l'algorithme **5.9 finaliserLaPartie** qui se lance.

5.2.3 Algorithmme



5.3.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
tabSudoku	Tableau	tableau de nbSudoku, il contient toutes les valeurs du sudoku et actualisé avec la saisie du joueur en cas de compatibilité
erreureAutorise	Entier positif court	Le nombre d'erreur autorisée avant la fin de la partie
nbErreurJoueur	Entier positif court	Le nombre d'erreur du joueur
nbTour	Entier positif court	Le nombre de tour
TAILLE_TAB	Entier positif court	La taille de la grille du sudoku défini à 9
TAILLE_ZONE	Entier positif court	La taille de chaque zone défini à 3
issueDeLaSaisie	énuméré	Défini chaque situations; {compatible, incompatible, abandon, erreur de saisie}
indiceLigne	Entier positif court	Élément qui permet l'itération de la boucle pour le traitement verticale du sudoku
indiceColonne	Entier positif court	Élément qui permet l'itération de la boucle pour le traitement horizontale du sudoku

5.3.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
afficherSudoku	5.4 afficherSudoku	tabSudoku, TAILLE_ZONE, TAILLE_TAB	affichage
afficher le nombre de tour et d'erreur	afficher le nombre de tour et d'erreur	nbTour, nbErreurJoueur, erreureAutorise	affichage
saisieVerifJoueur	5.5 saisieVerifJoueur	tabSudoku	issueDeLaSaisie, valModifie
traitementDesDifferentesSituations	Afficher le message adapté pour chaque situations cf Annexe 1	issueDeLaSaisie, nbErreurJoueur, erreureAutorise, TAILLE_TAB TAILLE_ZONE,	nbErreurJoueur

5.4 afficherSudoku

5.4.1 But de l'action

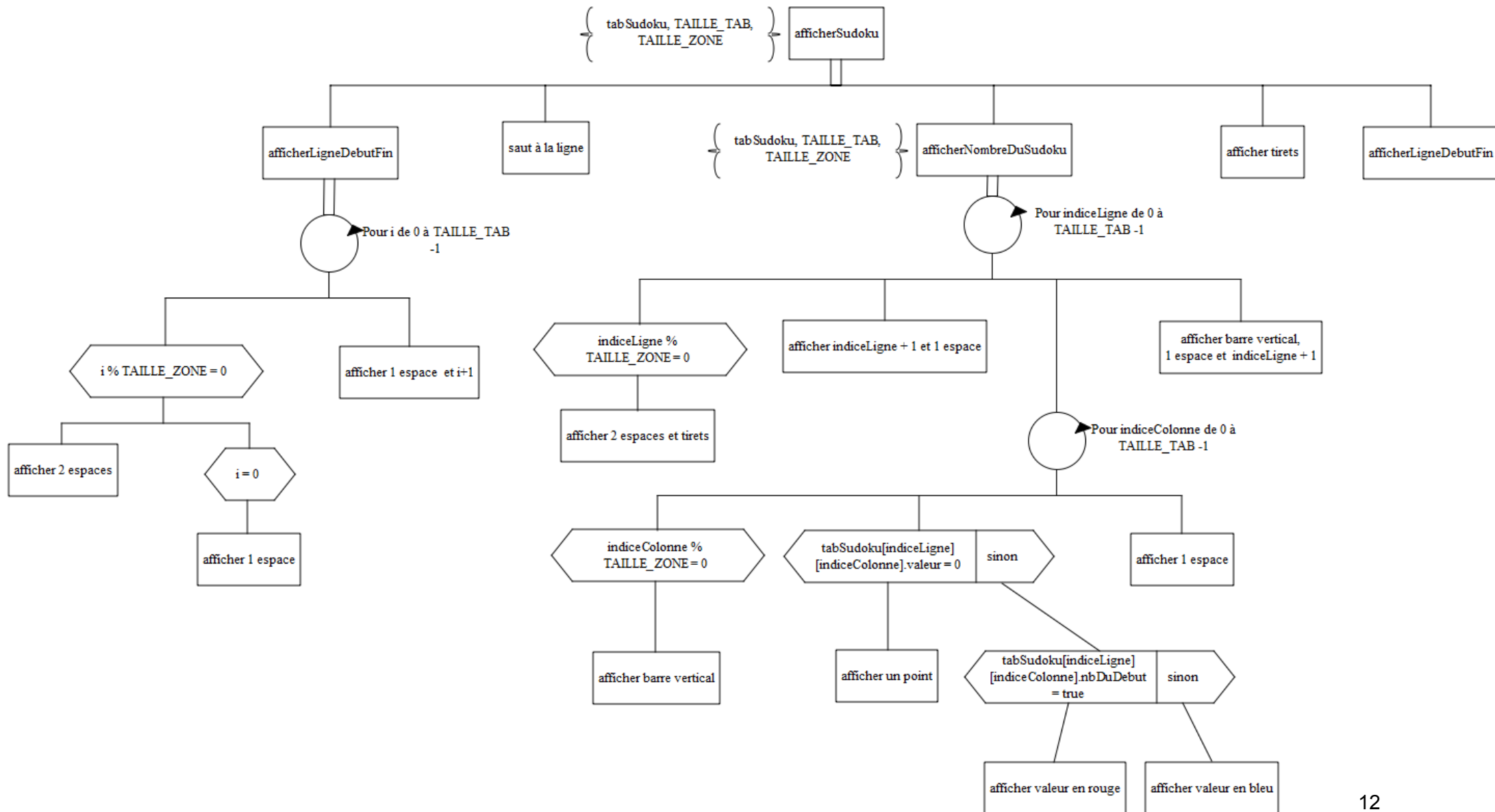
Afficher le tableau sudoku

5.4.2 Stratégie de l'algorithme mise en œuvre et justification

L'algorithme respecte un parcours complet par traitement systématique de chaque valeur du tableau et à sa s'ajoute 2 boucles supplémentaires qui viennent afficher l'entête et le pied du tableau. Si l'algorithme paraît complexe c'est parce que l'affichage l'est aussi et doit respecter des critères précis.

Chaque élément du tableau est une

5.4.3 Algorithmme



5.4.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
nbSudoku	Enregistrement	structure pour chaque nombre du sudoku
tabSudoku	Tableau	tableau de nbSudoku
indiceLigne	Entier positif court	Élément qui permet l'itération de la boucle pour l'affichage verticale du sudoku
indiceColonne	Entier positif court	Élément qui permet l'itération de la boucle pour l'affichage horizontale du sudoku

5.4.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
afficherLigneDebutFin	afficher l'entête et le pied du sudoku	aucune	Affichage
afficherNombreDuSudoku	afficher l'ensemble des valeurs du tableau	tabSudoku, TAILLE_TAB, TAILLE_ZONE	Affichage

5.5 saisieVerifJoueur

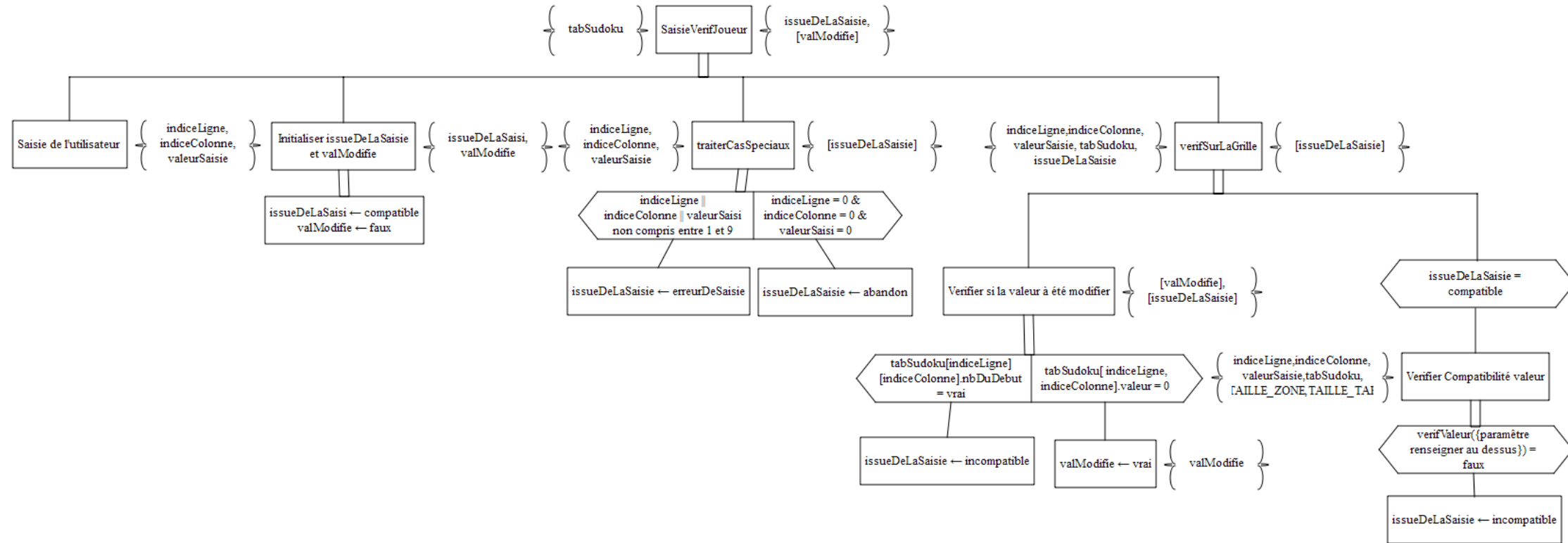
5.5.1 But de l'action

Vérifier si la valeur saisie par le joueur est compatible ou non avec la grille.

5.5.2 Stratégie de l'algorithme

L'algorithme saisieVerifJoueur traite d'abord toutes les situations éliminatoires tels que erreurDeSaisie, Abandon, et modification sur la grille de départ. Puis après avoir écarté tous ces cas et confirmer que la valeur peut potentiellement être compatible, l'algorithme vérifie si la valeur saisie par l'utilisateur est compatible avec la grille voir [5.6 verifValeur](#).

5.5.3 Algorithmme



5.5.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
tabSudoku	Tableau	tableau de nbSudoku
indiceLigne	Entier positif court	Élément qui permet l'itération de la boucle pour la vérification verticale du sudoku
indiceColonne	Entier positif court	Élément qui permet l'itération de la boucle pour la vérification horizontale du sudoku
issueDeLaSaisie	énuméré	Définit chaque situation; {compatible, incompatible, abandon, erreur de saisie}
valeurSaisie	caractère	La valeur saisie par le joueur
valModifie	Booléen	Vérifie si la valeur saisie modifie une valeur déjà existante

5.5.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
Saisie de L'utilisateur	Récupérer la saisie de l'utilisateur pour obtenir les valeurs résultats	clavier	indiceLigne, indiceColonne, valeurSaisie
traiterCasSpeciaux	traiter les cas qui empêchent la vérification de la valeur sur la grille	indiceLigne, indiceColonne, valeurSaisie	[issueDeLaSaisie]
verifSurLaGrille	Vérifier si la valeur modifie une ancienne valeur et si elle est compatible avec la grille	indiceLigne, indiceColonne, valeurSaisie, tabSudoku, issueDeLaSaisie	[issueDeLaSaisie]

5.6 rechercheDeValeurPossible

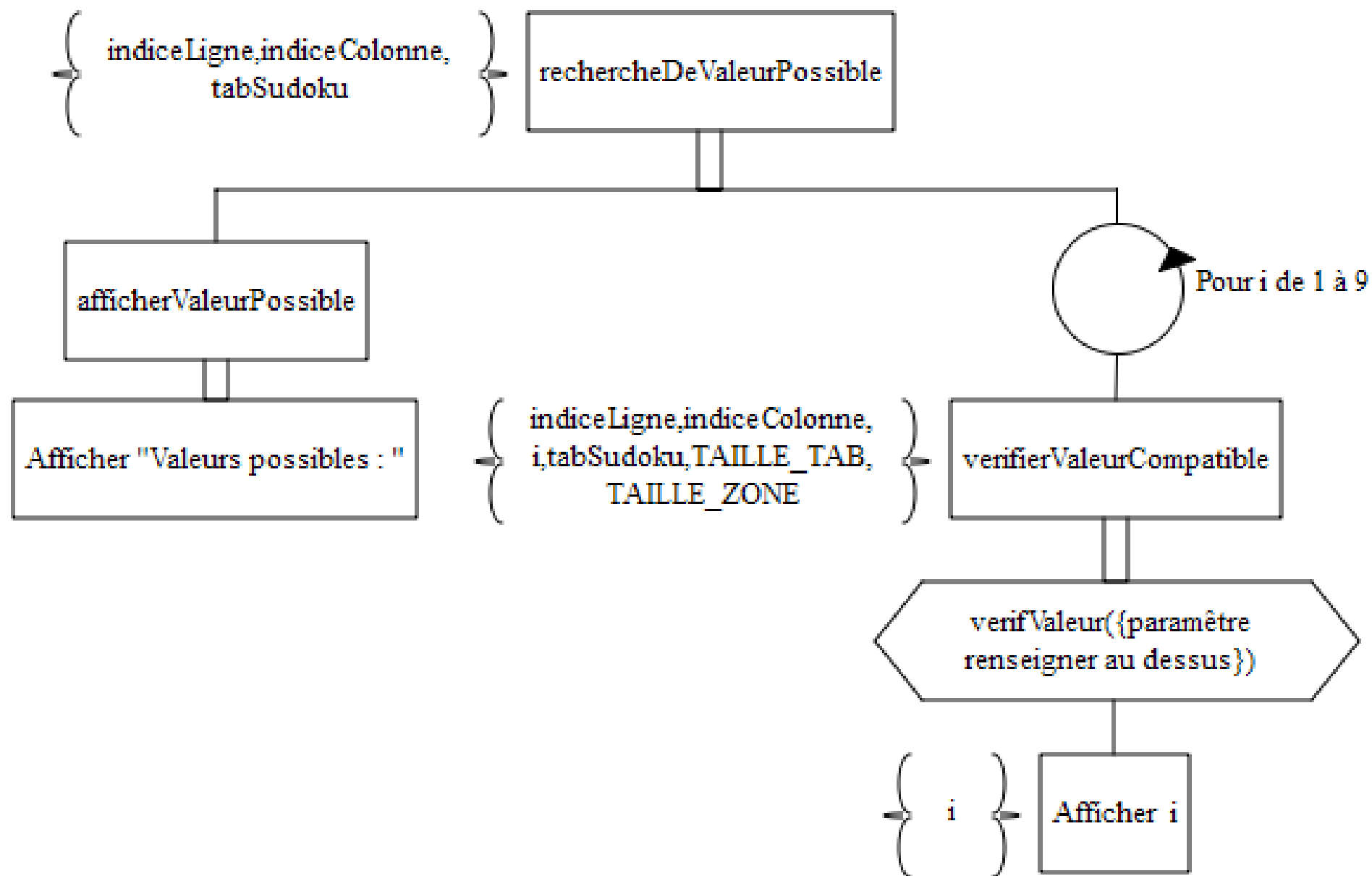
5.6.1 But de l'action

Rechercher toutes les valeurs possibles à la position de la saisie du joueur et les afficher

5.6.2 Stratégie de l'algorithme

La stratégie consiste simplement à balayer toutes les possibilités (de 1 à 9) que la case peut contenir, et afficher

5.6.3 Algorithme



5.6.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
tabSudoku	Tableau	tableau de nbSudoku
indiceLigne	Entier positif court	Élément qui permet l'itération de la boucle pour la vérification verticale du sudoku
indiceColonne	Entier positif court	Élément qui permet l'itération de la boucle pour la vérification horizontale du sudoku
TAILLE_TAB	Entier positif court	La taille de la grille du sudoku défini à 9
TAILLE_ZONE	Entier positif court	La taille de chaque zone défini à 3

5.6.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
afficherValeurPossible	Afficher "Valeurs possibles : "	aucune	Affichage
verifierValeurCompatible	Rechercher et afficher toutes les valeurs possibles à la position de la saisie du joueur	indiceLigne, indiceColonne, tabSudoku, i, TAILLE_TAB, TAILLE_ZONE	Affichage

5.7 verifValeur

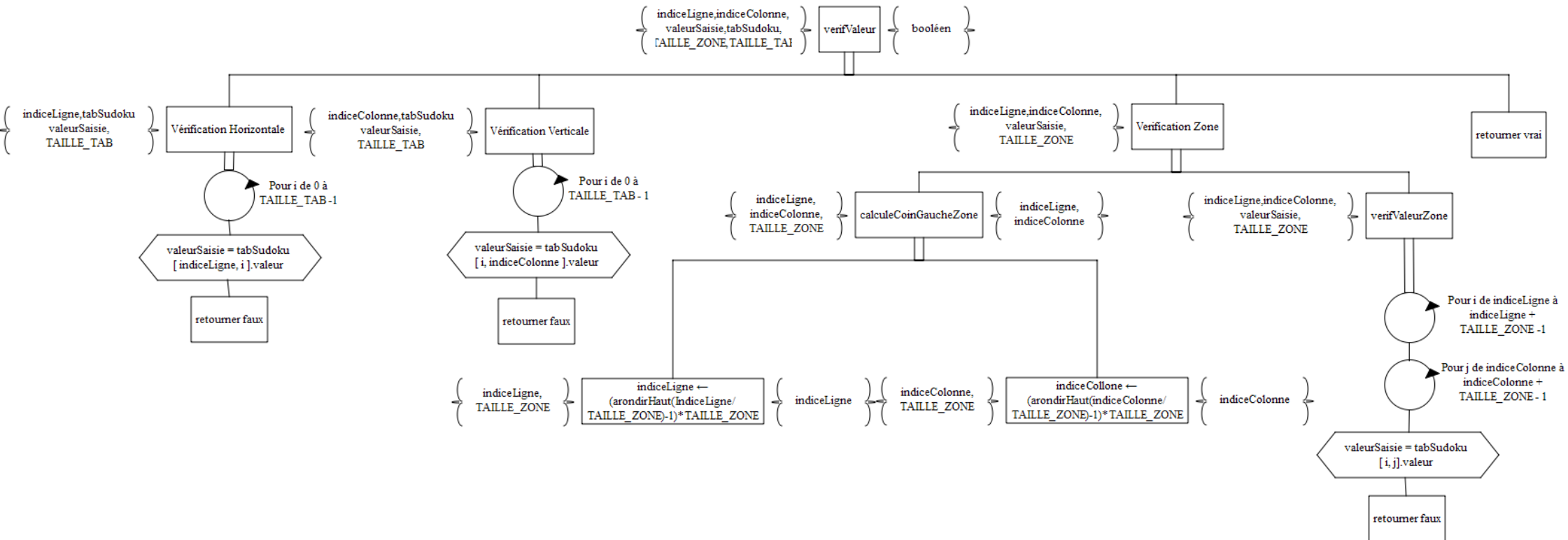
5.7.1 But de l'action

Vérifier si la valeur est compatible avec la grille

5.7.2 Stratégie de l'algorithme

Cette fonction (puisque c'est une fonction) se décompose en 3 grands traitements **Verification Horizontale**, **Verification Verticale** et **Verification Zone**. Les deux premiers servent à vérifier respectivement la compatibilité sur la ligne et sur la colonne, et le dernier traitement sert à vérifier la compatibilité dans la zone où se trouve la valeur saisie d'abord on identifie le coin haut gauche de la zone en faisant le calcul suivant [cf. Annexe 3](#) : $(\text{arondirHaut}(\text{Indice}/\text{TAILLE_ZONE})-1)*\text{TAILLE_ZONE}$ sur l'indice ligne et l'indice colonne puis avec cela il nous suffit de passer à travers les valeurs de la zone et de les comparer pour voir si la valeur est compatible.

5.7.3 Algorithmme



5.7.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
valeurSaisie	caractère	La valeur saisie par le joueur
tabSudoku	Tableau	tableau de nbSudoku
indiceLigne	Entier positif court	Élément qui permet l'itération de la boucle pour la vérification verticale du sudoku
indiceColonne	Entier positif court	Élément qui permet l'itération de la boucle pour la vérification horizontale du sudoku
TAILLE_TAB	Entier positif court	La taille de la grille du sudoku défini à 9
TAILLE_ZONE	Entier positif court	La taille de chaque zone défini à 3

5.7.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
Initialiser compatible	initialise compatible à vrai	aucune	compatible
Verification Horizontale	Vérifier si la valeur est compatible avec la ligne dans laquelle elle se trouve de la grille	indiceLigne, tabSudoku, valeurSaisie, TAILLE_ZONE	peut retourner faux
Verification Verticale	Vérifier si la valeur est compatible avec la colonne dans laquelle elle se trouve dans la grille	indiceColonne, tabSudoku, valeurSaisie, TAILLE_ZONE	peut retourner faux
Verification Zone	Vérifier si la valeur est compatible avec la Zone dans laquelle elle se trouve dans la grille	indiceColonne, indiceLigne, tabSudoku, valeurSaisie, TAILLE_TAB	peut retourner faux

5.8 tabPlein

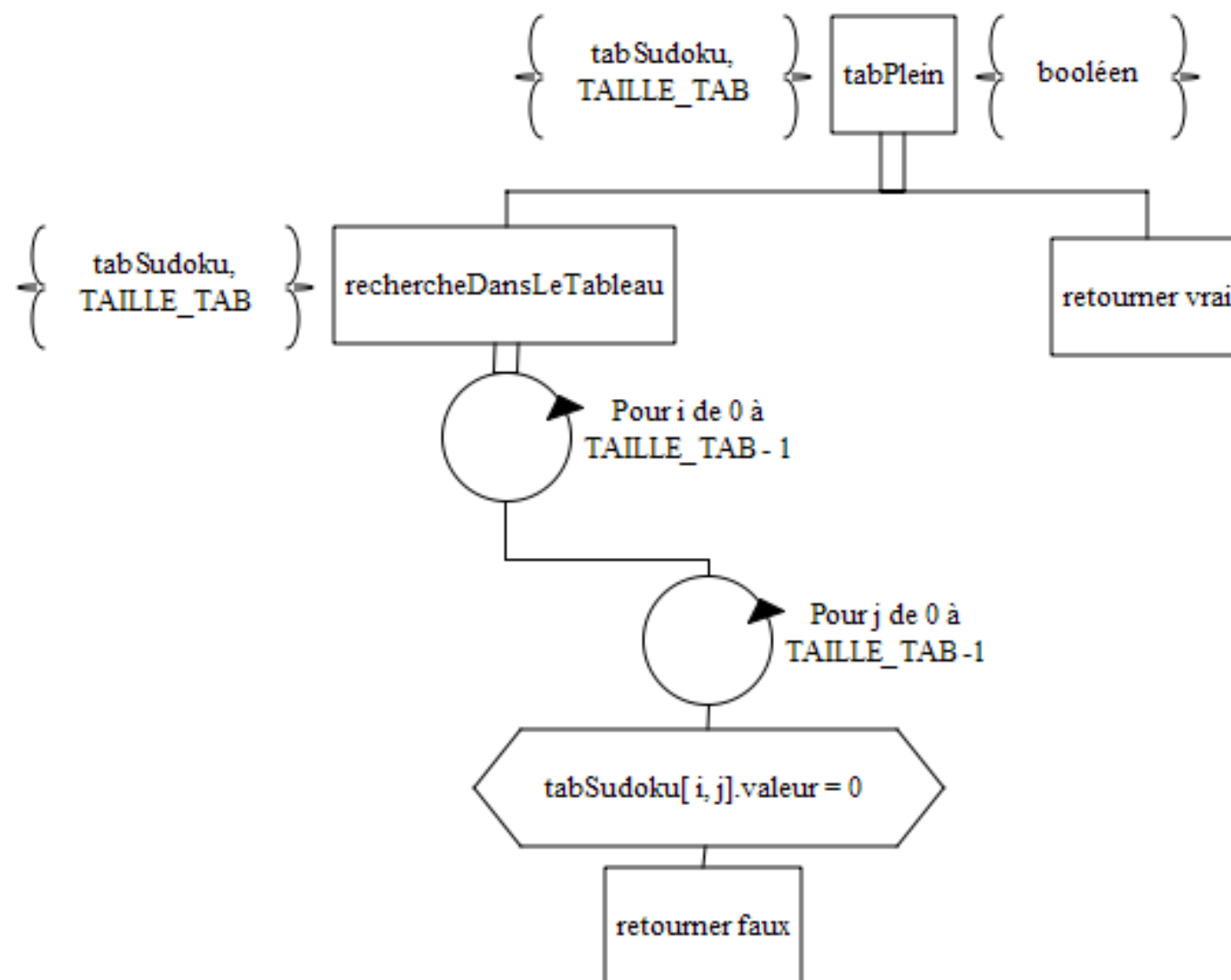
5.8.1 But de l'action

Vérifier si la condition de victoire est vrai

5.8.2 Stratégie de l'algorithme

La fonction tabPlein vérifie si le tableau est rempli de valeur, s' il trouve un nbSudoku dans la grille dont la valeur est 0 alors elle renvoie faux.

5.8.3 Algorithme



5.8.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
tabSudoku	Tableau	tableau de nbSudoku
TAILLE_TAB	Entier positif court	La taille du tableau tabSudoku
booléen	booléen	si le tableau est plein ou non

5.8.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
rechercheDansLeTableau	recherche dans le tableau si un nbSudoku dans la grille à pour valeur est 0	tabSudoku, TAILLE_TAB	booléen

5.9 Finaliser la partie

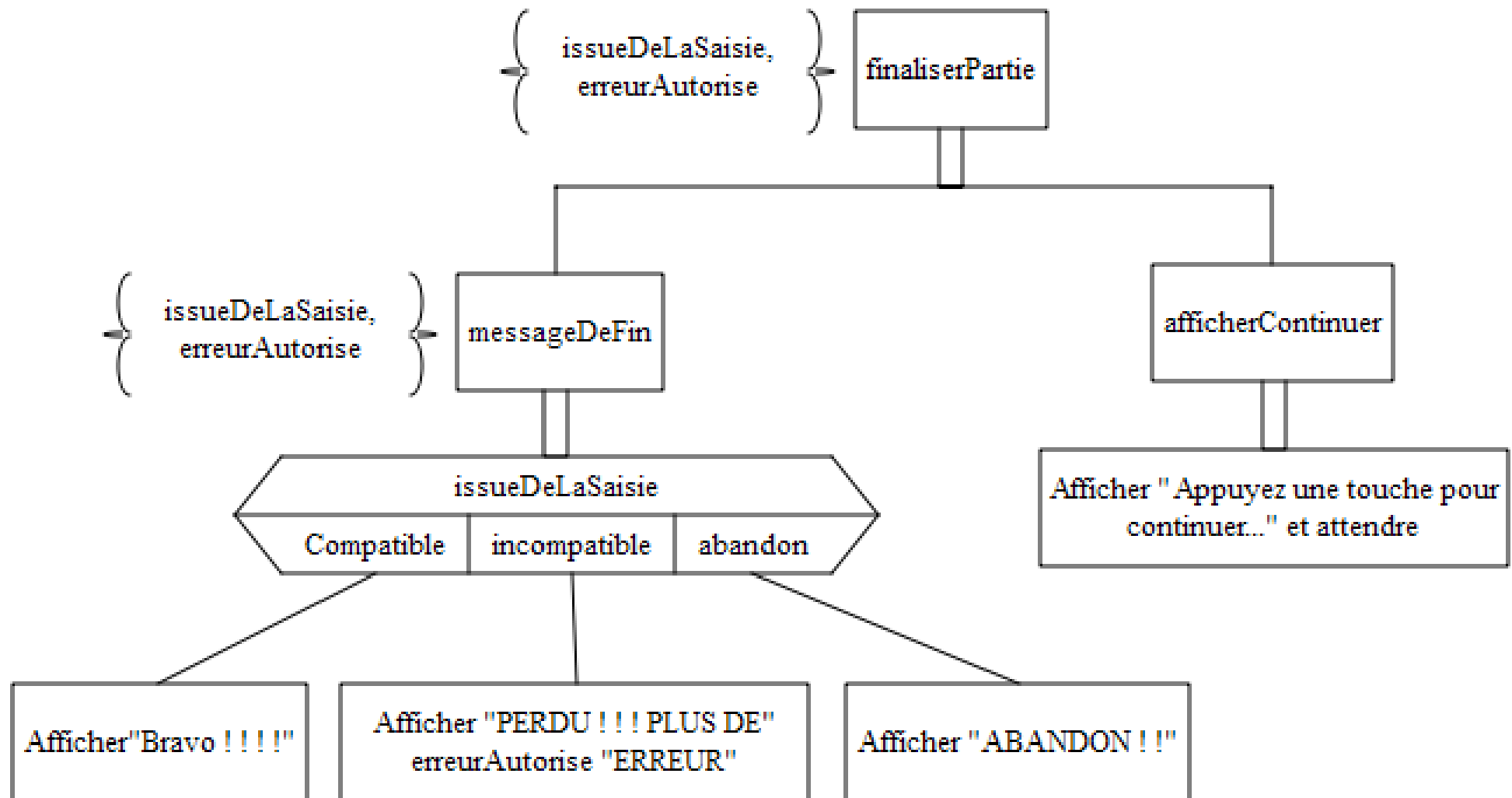
5.9.1 But de l'action

Vérifier si la condition de victoire est vraie

5.9.2 Stratégie de l'algorithme

En utilisant un switch on distingue les 3 manières de finir la partie et affiche un message approprié.

5.9.3 Algorithme



5.9.4 Dictionnaire des éléments associés à cet algorithme

Nom	Type	Signification
issueDeLaSaisie	énuméré	Défini chaque situations; {compatible, incompatible, abandon}
erreurAutorise	Entier positif court	Le nombre d'erreurs autorisée

5.9.5 Dictionnaire des actions associées à cet algorithme

Nom Action	But	Données	Résultats
messageDeFin	Afficher le résultat de la saisie	issueDeLaSaisie, erreurAutorise	Affichage
afficherContinuer	afficher "appuyez sur une touche pour continuer"	aucune	Affichage

6. Tests réalisés et traces d'exécution

Les résultats obtenus peuvent être consignés sur le tableau des jeux d'essais :

	Tests programme XXX			résultats attendus		affichage écran		Résultats exécution		affichage écran
	données	élément1	élément2	élément3	résultat_1	résultat_2		résultats OBTENUS	Remarques : problèmes rencontrés - Idées de solution	
3										
4										
5										
6										
7										
8										
9										

Les copies d'écran peuvent être ajoutées ici, ou, si elles sont trop nombreuses, être regroupées dans une annexe spécifique, en fin de document.

Copies d'écran :

fond clair /
caractères foncés

Capture_écran_1 : Comportement lié au scénario XX

7. Remarques

Informations que les étudiants souhaitent communiquer aux enseignants au sujet de cette SAé :

- Choix réalisés
- Difficultés
- Temps total passé hors séances prévues à l'emploi du temps (somme des temps passés par chaque membre du groupe)

8. Code C++

1. Choix d'organisation des fichiers composant le code source

Quelques mots pour expliquer les choix d'organisation du code dans le/les fichiers rendus.

Liste des fichiers et contenu (en intention = quels types de sous-programmes il contient ; et/ou extension = liste des sous-programmes qu'il contient).

2. Code source

Les fichiers .cpp et .h ne sont pas inclus dans le présent document, mais sont joints au dossier avec un paragraphe en-tête certifiant l'originalité du code produit.

9. Annexe 1 – maquettes d'écran prévues dans les spécifications

maquette_écran_compatible

maquette_écran_incompatible

10. Annexe 2 – Traces d'exécution - copies d'écran

Des copies d'écran

11. Annexe 3 – Précisions

Calcule coin Gauche Indice Zone :

indice	/3	arrondirHaut	-1*3
1	0,3333333333	1	0
2	0,6666666667	1	0
3	1	1	0
4	1,333333333	2	3
5	1,666666667	2	3
6	2	2	3
7	2,333333333	3	6
8	2,666666667	3	6
9	3	3	6