

Compte-Rendu de Tests Unitaires Indice A

SOMMAIRE

<u>1. OBJET</u>	3
<u>2. PREPARATION DES TESTS UNITAIRES</u>	3
2.1 Chronologie des opérations.....	3
2.2 Moyens utilisés.....	3
2.3 Jeux d'essais.....	
<u>3. REALISATION DES TESTS UNITAIRES</u>	4
3.1 Principes mis en oeuvre.....	4
3.2 Vérification des exigences de développement	4
3.3 Résultats des tests unitaires.....	4
otho.....	5
materiel.....	5
allocation_memoire.....	6
ecran_presentation.....	6
menu_principal.....	7
jouer.....	7
initialise_ecran_jeu.....	8
jeu_normal.....	8
jeu_2joueur.....	9
ordinateur_vs_ordinateur.....	11
initialise_tableau_commande.....	12
initialise_jeu.....	13
affiche_pions.....	13
compte_pions.....	14
choix_joueur.....	14
change_joueur.....	14
jouer_coup.....	15
coup_precedent.....	15
coup_suivant.....	16
memorise_jeu.....	16
conseil_coup.....	17
retourne.....	17
coups_possibles.....	18
ordinateur_debutant.....	18
ordinateur_intermediaire.....	19
evaluation_damier.....	19
alpha_beta.....	20
ordinateur_expert.....	20
statistique.....	21
statistique_affiche.....	21
initiation.....	22

affiche_coup.....	22
fin_jeu.....	22
visualiser.....	23
apropos.....	24
regles.....	24
regles_affiche_bouton.....	25
regles_affiche.....	25
sauve_partie.....	25
charge_partie.....	26
statistique_affiche_donnee.....	26
high_score.....	26
affiche_high_score.....	27
sauve_score.....	27
charge_score.....	27
ouverture.....	28
prepare_ouverture.....	28
ouverture_visualise.....	29
selection_ouverture.....	30
4. CONCLUSIONS	31
4.1 Couverture de test.....	31
4.2 Conclusion sur les tests unitaires - OTHO version 1.0a.....	31
5. MATRICE DE TRACABILITE DES EXIGENCES	32

Indice	Date de diffusion	Intitulé
A	02/06/1997	Document initial

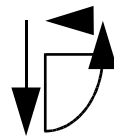
Toute modification de ce document par rapport à sa version précédente est repérée par un trait vertical dans la marge gauche.

1. OBJET :

<u>Objet :</u>	Ce document constitue le Compte-Rendu de Tests Unitaires du logiciel OTHO v 1.0a dans le cadre du projet OTHELLO.
<u>Domaine d'application :</u>	Logiciel destiné à une application type grand public.
<u>Documents de référence :</u>	Cahier des Charges indice B du 22/03/1997. Spécification Technique de Besoin indice A du 22/03/1997. Document de Conception Préliminaire indice A du 26/04/1997. Document de Conception Détaillée indice A du 02/06/1997.

2. PREPARATION DES TESTS UNITAIRES:**2.1 Chronologie des opérations :**

OPERATIONS	MACHINES
Codage des modules - Réalisation des écrans graphiques	PC Patrick / Bertrand
Compilation des composants : création des versions de travail (de la version 0.6a à la 1.0a)	PC Patrick
Tests unitaires non formalisés sur ces versions	PC Patrick
Tests unitaires formalisés sur la version 1.0a	PC Bertrand

**2.2 Moyens utilisés :**PC Patrick - Machine de codage

Configuration :

- µP : Cyrix P 166 (133 Mhz)
- RAM : 24 Mo
- Windows 95 OSR/2
- Carte graphique DIAMOND 3D 2000 (compatible VESA)
- gestionnaires de souris : mouse.com (émulé par Windows 95)

PC Bertrand - Machine hôte (servant aussi à la documentation)

Configuration :

COMPAQ DESKPRO 466

- µP : 486 DX2 66
- RAM : 12 Mo
- DOS 6.22 + Windows 3.1 Version COMPAQ
- Carte graphique Cirrus Logic (compatible VESA)
- Souris COMPAQ + gestionnaires de souris COMPAQ

Moyens humains : 1 personne (prise en compte uniquement pour les tests unitaires formalisés)

Temps prévu : 25 h (incluant la documentation) Temps réellement passé : h

2.3 Jeux d'essais :

Utilisation de fichiers othxxx.dat pour les tests de reprise de jeu, visualisation, ... (tests de gains de partie, perte de partie, égalité, différents niveaux). Ces fichiers sont créés grâce à l'utilisation d'OTHO dans le mode « enregistrement des parties ». Aucun fichier n'a été créé manuellement (le format des fichiers est binaire, donc quasi illisible). Des fichiers comme oth20.dat ont été créés grâce à OTHO en mode 2 joueurs humains (pour définir les placements des pions comme on voulait).

Utilisation du fichier ouvertur.txt en nominal. Pour les cas d'erreurs, a été réalisé un « mauvais » fichier ouvertur.txt.

Utilisation du fichier score.dat créé par OTHO.

3. REALISATION DES TESTS UNITAIRES :

3.1 Principes mis en oeuvre :

Des tests unitaires non formalisés se sont déroulés pendant toute la phase de codage sur la machine de codage. Ce document ne les prend pas en compte, et ne représente donc que la compilation des tests unitaires de chaque module décrit dans le DCD indice A (soit une couverture théorique de 100 %). Tous les modules testés ont été testés à leur dernière version (dernière date de modification).

Ces tests unitaires formalisés se sont déroulés du 25/05/97 au 02/06/1997 sur une machine hôte. La mise en forme complète de ce document a été réalisée en même temps que les tests sur la même machine (mise à jour du texte - correction d'anomalies - ...).

Pour connaître la liste des tests unitaire, il suffit de se reporter au sommaire de ce document.

Les tableaux d'essais sont calqués sur le modèle suivant :

Module :	NOM DU MODULE	DATE DE LA DERNIERE MODIFICATION
Rôle :	Rôle du module testé (en quelques lignes)	
Entrées :	Entrée(s) de la procédure testée	
Sorties :	Sortie(s) de la procédure testée	
CONTROLES :		RESULTATS :
Définition des actions à vérifier.		OK - NOK

3.2 Vérification des exigences de développement :

Ces vérifications (issues des « spécifications » du § 5 du DCD indice A) se sont soldées, pour la plupart, par des audits de code demandant :

- ① vérifications de la présence du cartouche d'en-tête pour chaque module;
- ② vérifications sur la structure du code pour quelques modules sélectionnés;
- ③ vérification de la lisibilité du code pour quelques modules sélectionnés;
- ④ vérification de l'application des règles de nommage des variables pour quelques modules sélectionnés;

Les modules sélectionnés ont été choisis en fonction de leur taille, leur situation dans le graphe d'appels, leur importance pour le fonctionnement nominal, ... Le choix s'est porté sur 1 à 2 module par fichier .c. Les résultats des audits de code sont les suivants :

Vérifications	Résultat	Commentaires
① : présence cartouche	100 %	Tous les modules ont des cartouches.
② : structure du code	100 %	Les modules vus respectent la structure du code imposée.
③ : lisibilité	OK	Les modules vus sont lisibles facilement.
④ : règles de nommage	100 %	Les règles de nommage des variables sont respectées pour les modules vus.

Ces tests permettent en grande partie de répondre à l'exigence [DEV.1] sur la maintenabilité et à l'exigence [DEV.2] sur la modularité du code logiciel d'OTHO

3.3 Résultats des tests unitaires :

Voir pages suivantes.

<u>Module :</u>	otho (programme principal)	20/03/1997
<u>Rôle :</u>	programme principal (le « main » en C) : squelette de l'application	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de l'initialisation de toutes les fonctions utilisées dans la librairie	
Vérification du branchement au module : materiel ()	
Vérification du branchement au module: allocation_memoire ()	
Vérification de l'initialisation des variables globales	
Vérification du branchement au module: ecran_presentation ()	
Vérification de l'appel du menu principal: menu_principal ()	
Vérification de la restauration du contexte graphique : fin d'utilisation de la souris, libération de la mémoire, ...	
Vérification de l'affichage de l'écran de fin de jeu :	

<u>Module :</u>	materiel	13/01/1997
<u>Rôle :</u>	Fonction permettant de vérifier si l'utilisateur d'OTHO possède bien le matériel requis pour l'utilisation du logiciel.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification du test de la carte graphique : cas nominal Vérification du test de la carte graphique : cas d'erreur	
Vérification du test de la quantité de mémoire vidéo : cas nominal Vérification du test de la quantité de mémoire vidéo : cas d'erreur	
Vérification de la détermination du type de processeur : cas nominal Vérification de la détermination du type de processeur : cas d'erreur	
Vérification de la détermination de la présence de la souris : cas nominal Vérification de la détermination de la présence de la souris : cas d'erreur (suppression du mouse.com sur le micro-ordinateur TEST)	

<u>Module :</u>	allocation_memoire	06/04/1997
<u>Rôle :</u>	allouer de la mémoire pour les éléments graphiques de l'application	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier l'allocation de mémoire : <i>cas nominal</i> - pour les chiffres graphiques (score) + chiffre graphique tampon - pour les pions de jeu + pion tampon - pour les indicateurs de choix (test de bon fonctionnement du jeu)	
Vérifier l'allocation de mémoire : <i>Cas d'erreur</i>	
Vérification du branchement au module Charge_score () (chargement en mémoire des high-scores d'OTHO)	

<u>Module :</u>	ecran_presentation	23/03/1997
<u>Rôle :</u>	Fonction permettant d'afficher l'écran de présentation (image de R. Gatliff)	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de l'apparition de l'image « Othello Match » à l'écran (cas nominal) Vérification de l'arrêt de l'application : cas d'erreur	
Vérifier l'affichage du curseur de la souris	
Vérifier la temporisation et la possibilité de passer soit en cliquant soit en appuyant sur une touche (et disparition de l'image)	

<u>Module :</u>	menu_principal	25/05/1997
<u>Rôle :</u>	Fonction permettant d'afficher et de gérer le menu principal.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Apparition de l'image du menu principal	
Vérification de l'affichage et fonctionnement nominal de la souris : - fonctionnement des click droit et gauche - positionnement	
Vérification de la prise en compte et de la gestion du clavier	
Vérification de la prise en compte (souris et clavier) des zones de validité pour l'affichage en surbrillance des options et affichage aide en ligne	
Prise en compte des choix d'option du menu principal : - Initiation : branchement au module jouer () - Jouer : branchement au module jouer () - Visualiser : branchement au module Visualiser () - A propos : branchement au module apropos () - Règles : branchement au module regles () - Ouvertures : branchement au module ouverture () - Scores : branchement au module affiche_high_score ()	

<u>Module :</u>	jouer	27/04/1997
<u>Rôle :</u>	Fonction permettant le jeu à plusieurs à Othello	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier le branchement au module initialise_ecran_jeu ()	
Vérifier la détermination du type de jeu à mettre en place : si « jeu normal » : vérifier le branchement au module jeu_normal () Si « Initiation » : vérifier le branchement au module initiation ()	

<u>Module :</u>	initialise_ecran_jeu	20/04/1997
<u>Rôle :</u>	Fonction permettant d'initialiser l'écran de jeu (mise en place des affichages, ...)	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Contrôle du chargement de l'image de fond avec effet d'affichage progressif	
Vérification de l'initialisation des objets graphiques du jeu : - chiffres indiquants les scores - pions noirs et blancs	
Affichage de la zone d'aide en ligne en bas de l'écran	

<u>Module :</u>	jeu_normal	21/05/1997
<u>Rôle :</u>	Fonction permettant de jouer en mode normal à OTHO.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Contrôle du chargement de l'image de fond « \images\otho.gif » avec effet d'affichage et vérification de l'affichage de la boîte de dialogue de choix du type de jeu	
Vérifier : « Si click sur Annuler alors retour au menu principal »	
Vérifier le branchement au module : initialise_jeu ()	
Réaliser le cas d'une partie sauvegardée (click sur reprendre une partie) alors : - Vérifier l'affichage d'une boîte de dialogue saisie du Nom de la partie - Vérifier la possibilité de rentrer le nom d'une partie au clavier - Vérifier l'affichage en aide en ligne du chargement de la partie en cours - Vérifier le branchement au module Charge_partie (nom de la partie)	
Vérifier ensuite l'initialisation des paramètres de jeu : - si jeu à deux joueurs sélectionné : vérifier le branchement au module jeu_2joueurs () - sinon vérifier le branchement au module ordinateur_vs_ordinateur ()	
ATTENTION : faire cette action pour une reprise de partie et un jeu normal	

<u>Module :</u>	jeu_2joueur	21/05/1997
<u>Rôle :</u>	Fonction permettant de joueur à deux joueurs (humain / ordinateur)	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de la possibilité de saisir des noms de joueurs humains : - cas d'un jeu humain/humain - cas OTHO Blanc - cas OTHO Noir	
Vérification de l'affichage du nom des joueurs dans leurs fenêtres respectives	
<u>Cas de reprise de partie :</u>	
Vérification de l'affichage du dernier coup de la partie reprise et de l'annonce d'une reprise de partie dans l'aide en ligne	
Vérifier le branchement au module coup_suivant ()	
<u>Cas du début de jeu normal :</u>	
Vérification de l'affichage en aide en ligne : le Joueur Noir commence Vérifier le branchement au module statistique ()	
<u>Boucle de jeu :</u>	
Vérifier le branchement au module Initialise_tableau_commande ()	
<u>C'est à l'ordinateur de jouer :</u> Si niveau initiation Vérifier le branchement au module ordinateur_debutant () Si niveau Débutant Vérifier le branchement au module ordinateur_intermediaire () Si niveau intermédiaire Vérifier le branchement au module ordinateur_expert () + profondeur Si niveau confirmé Vérifier le branchement au module ordinateur_expert () + profondeur Si niveau expert Vérifier le branchement au module ordinateur_expert () + profondeur	
Calcul des statistiques : Vérifier le branchement au module statistique ()	
Affichage pions : Vérifier le branchement au module affiche_pions ()	
Mémorise jeu courant : Vérifier le branchement au module memorise_jeu ()	
Changement de main : Vérifier le branchement au module change_joueur ()	
CONTROLES :	RESULTATS :

<p><u>C'est au joueur humain de jouer :</u></p> <p>Si niveau initiation : vérification de l'affichage des coups possibles : - Vérifier le branchement au module affiche_coup ()</p>	
<p>Prise en compte du click souris sur appui « coup précédent » : - Vérifier le branchement au module coup_precédent ()</p>	
<p>Prise en compte du click souris sur appui « coup suivant » : - Vérifier le branchement au module coup_suivant ()</p>	
<p>Prise en compte du click souris sur appui « conseil » : - Vérifier le branchement au module affiche_coup () - Vérifier le branchement au module conseil_coup ()</p>	
<p>Prise en compte du click souris sur appui « stop » : fin du jeu</p>	
<p>Prise en compte du click souris (valide) sur le damier : - Détermination de l'endroit : Vérifier le branchement au module jouer_coup () - Calcul des statistiques : Vérifier le branchement au module statistique () - Affichage des pions : Vérifier le branchement au module affiche_pions () - Mémorisation du jeu courant : Vérifier le branchement au module memorise_jeu () - Changement de main : Vérifier le branchement au module change_joueur ()</p>	
<p>Prise en compte du click souris (invalide) sur le damier : Affichage dans l'aide en ligne : coup impossible</p>	
<p><u>Tests de fin de jeu :</u></p> <p>Cas : les deux joueurs doivent passer leur tour Vérification de l'affichage à l'écran : « fin du jeu !! »</p>	
<p>Cas : le joueur courant doit passer son tour - Vérification de l'affichage à l'écran : « le joueur doit passer son tour » - Changement de main : Vérifier le branchement au module change_joueur ()</p>	
<p>Cas : Partie terminée - Congratulations au gagnant : Vérifier le branchement au module fin_jeu () - Gestion des high-scores : Vérifier le branchement au module high_scores ()</p>	
<p>Vérifier la fonction sauvegarde de la partie jouée (cas nominal) si jeu humain contre humain : Vérifier le branchement au module sauve_partie () si jeu humain contre ordinateur sauf niveau initiation : Vérifier le branchement au module sauve_partie ()</p>	
<p>Vérifier la non validité de la fonction sauvegarde de la partie pour un jeu en initiation</p>	
<p>Vérifier la non validité de la fonction sauvegarde de la partie pour un jeu où aucun coup n'a été joué</p>	

<u>Module :</u>	ordinateur_vs_ordinateur	21/05/1997
<u>Rôle :</u>	Fonction permettant à deux ordinateurs de jouer entre eux à des niveaux quelconques	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Affectation des noms OTHO Blanc et OTHO Noir dans les cases respectives.	
Vérification de l'affichage du nom des joueurs dans leurs fenêtres respectives	
<u>Cas de reprise de partie :</u>	
Vérification de l'affichage du dernier coup de la partie reprise et de l'annonce d'une reprise de partie dans l'aide en ligne	
Vérifier le branchement au module coup_suivant ()	
<u>Cas du début de jeu normal :</u>	
Vérification de l'affichage en aide en ligne : le Joueur Noir commence	
Vérifier le branchement au module statistique ()	
<u>Boucle de jeu :</u>	
Vérifier le branchement au module Initialise_tableau_commande ()	
Prise en compte du click souris sur appui « coup suivant »:	
Si niveau initiation Vérifier le branchement au module ordinateur_debutant ()	
Si niveau Débutant Vérifier le branchement au module ordinateur_intermediaire ()	
Si niveau intermédiaire Vérifier le branchement au module ordinateur_expert () + profondeur	
Si niveau confirmé Vérifier le branchement au module ordinateur_expert () + profondeur	
Si niveau expert Vérifier le branchement au module ordinateur_expert () + profondeur	
Calcul des statistiques : Vérifier le branchement au module statistique ()	
Affichage pions : Vérifier le branchement au module affiche_pions ()	
Mémorise jeu courant : Vérifier le branchement au module memorise_jeu ()	
Changement de main : Vérifier le branchement au module change_joueur ()	

CONTROLES :	RESULTATS :
Prise en compte du click souris sur appui « coup précédent » : - Vérifier le branchement au module coup_précédent ()	
Prise en compte du click souris sur appui « conseil » : - Vérifier le branchement au module affiche_coup () - Vérifier le branchement au module conseil_coup ()	
Prise en compte du click souris sur appui « stop » : fin du jeu	
<u>Tests de fin de jeu :</u> Cas : les deux joueurs doivent passer leur tour Vérification de l'affichage à l'écran : « fin du jeu !! »	
Cas : le joueur courant doit passer son tour - Vérification de l'affichage à l'écran : « le joueur doit passer son tour » - Changement de main : Vérifier le branchement au module change_joueur ()	
Cas : Partie terminée - Congratulations au gagnant : Vérifier le branchement au module fin_jeu ()	
Vérifier la fonction sauvegarde de la partie jouée (cas nominal) - Vérifier le branchement au module sauve_partie ()	
Vérifier la non validité de la fonction sauvegarde de la partie pour un jeu où aucun coup n'a été joué	

<u>Module :</u>	initialise_tableau_commande	06/04/1997
<u>Rôle :</u>	Fonction initialisation le tableau de commande du jeu (boîte à outils)	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de l'affichage des boutons « < », « > », « stop », « conseil » dans la boîte à outils	
Vérification de l'affichage de la version d'OTHO - Vérifier la cohérence	

<u>Module :</u>	initialise_jeu	19/04/1997
<u>Rôle :</u>	Fonction permettant d'initialiser le jeu	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	
CONTROLES :		RESULTATS :
Vérification de la mise à zéro de toutes les variables globales de données de statistiques pour les joueurs Noir et Blancs et initialisation du damier		
Vérification de la préparation des tableaux de bord : - Affichage de la boîte à outils - Affichage des tableaux de score joueurs BLANC et NOIR - Affichage de la zone d'aide en ligne - Affichage des pions dans les zones réservées des tableaux d'indications		
Vérifier le placement des pions en position début de jeu : Vérifier le branchement aux modules jouer_coup (4,4, BLANC) jouer_coup (4,5, NOIR) jouer_coup (5,5, BLANC) jouer_coup (5,4, NOIR)		
Affichage du nombre de pions à l'écran : vérifier le branchement au module compte_pions ()		
Affichage des pions sur le damier : Vérifier le branchement au module affiche_pions ()		

<u>Module :</u>	affiche_pions	24/03/1997
<u>Rôle :</u>	Fonction permettant d'afficher un pion sur le damier avec un effet de retournement	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	
CONTROLES :		RESULTATS :
Vérifier l'action de reconnaissance en X et Y de l'emplacement des pions :		
Vérification de l'affichage du nouveau pion et des pions retournés avec un effet de retournement		

<u>Module :</u>	compte_pions	23/01/1997
<u>Rôle :</u>	Fonction permettant de compter les nombre de pions de chaque joueur	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier l'action de comptage des pions noirs et blancs	
Vérifier la cohérence du résultat avec la réalité du damier	
Vérifier l'affichage correct grâce aux nombre graphiques des nombres de pions.	
Vérifier la réalisation de la mise à jour des statistiques (joueurs Noir et Blanc)	

<u>Module :</u>	choix_joueur	21/02/1997
<u>Rôle :</u>	Fonction permettant de laisser choisir qui des deux joueurs commence en premier.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	TypeJeu	

CONTROLES :	RESULTATS :
Vérification de la détermination du type de jeu : - ordinateur contre ordinateur (Noir et Blanc / Blanc et Noir) - humain Noir contre ordinateur Blanc - humain Blanc contre ordinateur Noir	

<u>Module :</u>	change_joueur	18/04/1997
<u>Rôle :</u>	Fonction permettant de changer de main au cours du jeu.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification du changement de main (modification de la variable JoueurCourant)	
Vérification de l'affichage du joueur courant dans la zone de la boîte à outils et vérification du branchement au module affiche_pions () pour l'effet de retournement	

<u>Module :</u>	jouer_coup	23/03/1997
<u>Rôle :</u>	Fonction permettant de jouer un coup.	
<u>Entrées :</u>	caseX caseY : coordonnées du coup joué. Couleur : couleur des pions du joueur	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de la mise à jour du damier avec le nouveau coup (caseX , caseY)	
Vérification de l'affichage du pion aux coordonnées jouées (CaseX, CaseY)	

<u>Module :</u>	coup_précédent	19/04/1997
<u>Rôle :</u>	Fonction permettant de gérer le clic souris sur le bouton coup précédent.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Premier coup de la partie : Vérification de l’Affichage dans l’aide en ligne : « Vous ne pouvez pas revenir en arrière » et impossibilité de reculer.	
On n’est pas au premier coup de la partie : Vérification du changement de joueur	
Ré-affectation des anciennes statistiques du coup précédent (score, mobilité, évaluation, feuilles, profondeur, coups)	
Affichage du damier représentant le coup précédent : Vérifier les branchements successifs au module jouer_Coup ()	
Remise à jour des statistiques : Vérifier le branchement au module statistique ()	
Vérifier le branchement au module change_Joueur ()	

<u>Module :</u>	coup_suivant	19/04/1997
<u>Rôle :</u>	Fonction permettant de gérer le clic souris sur le bouton coup suivant.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
<u>Dernier coup de la partie :</u> Vérification de l’Affichage dans l’aide en ligne : « Vous ne pouvez pas avancer !!» et impossibilité réelle d’avancer.	
<u>On n’est pas au dernier coup de la partie :</u> Vérification du changement de joueur	
Ré-affectation des statistiques du coup suivant (score, mobilité, évaluation, feuilles, profondeur, coups)	
Affichage du damier représentant le coup suivant : Vérifier les branchements successifs au module jouer_Coup ()	
Remise à jour des statistiques : Vérifier le branchement au module statistique ()	
Vérifier le branchement au module change_Joueur ()	

<u>Module :</u>	memorise_jeu	20/04/1997
<u>Rôle :</u>	Fonction permettant de mémoriser les coups du jeu.	
<u>Entrées :</u>	JoueurCourant : couleur du joueur demandant de recalculer les statistiques	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de la copie du damier en cours en un nouveau damier	
Vérification de l’enregistrement mémoire des statistiques du damier en cours	

<u>Module :</u>	conseil_coup	21/05/1997
<u>Rôle :</u>	Fonction permettant de gérer le clic souris sur le bouton « Conseil ».	
<u>Entrées :</u>	Joueur_Courant	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification du calcul du nombre de pions (données issues des statistiques)	
Vérification de la affectation de la profondeur de recherche (niveau expert)	
Vérification de la recherche du meilleur coup avec la méthode alpha-beta et vérification du branchement au module alpha-beta ()	
Vérification de l'affichage du coup conseillé dans l'aide en ligne : « Je vous conseille la case »,X;Y	

<u>Module :</u>	retourne	02/03/1997
<u>Rôle :</u>	Fonction permettant de déterminer si le placement est autorisé et/ou permet de retourner les pions.	
<u>Entrées :</u>	Damier : damier de jeu sur lequel on travaille X, Y : coordonnées du pion placé flip : autorisation de retourner les pions Joueur_Courant : joueur pour lequel on analyse le placement	
<u>Sorties :</u>	Autorisation du choix de l'emplacement	

CONTROLES :	RESULTATS :
<u>Si la case choisie (X, Y) sur le damier est libre</u>	
Vérification du fonctionnement de l'algorithme de placement (regarde dans toutes les directions possibles si le pion placé entoure des pions adverses)	
Vérifier le marquage des pions adverses entourés (et sa cohérence par rapport à la réalité)	
Vérifier l'affectation : Autorisation_Choix = OK	
<u>Si la case choisie (X, Y) sur le damier n'est pas libre :</u>	
Vérifier l'affectation : Autorisation_Choix = NOK	

<u>Module :</u>	coups_possibles	16/04/1997
<u>Rôle :</u>	Fonction permettant de déterminer les coups possibles du joueur courant	
<u>Entrées :</u>	Damier : damier de jeu sur lequel on travaille possible : coups possibles X et Y du damier Joueur_Courant : joueur pour lequel on analyse le placement	
<u>Sorties :</u>	nombre de coups possibles.	

CONTROLES :	RESULTATS :
Vérification du fonctionnement de l'algorithme de détermination des coups possibles	
Vérifier l'affectation du nombre de coups possibles et sa cohérence par rapport au damier	

<u>Module :</u>	ordinateur_débutant	20/04/1997
<u>Rôle :</u>	Fonction permettant de déterminer le coup de l'ordinateur pour le niveau initiation.	
<u>Entrées :</u>	CouleurOrdinateur.	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier le fonctionnement de l'algorithme de priorité de jeu donnée aux 4 coins	
Vérifier le fonctionnement de l'algorithme de priorité de jeu donnée aux cases de type A (Carré central)	
Vérifier le fonctionnement de l'algorithme de priorité de jeu donnée aux cases de type B (bords du damier)	
Vérifier le fonctionnement de l'algorithme de priorité de jeu donnée aux cases de type C (cases comprises entre le carré central et le bord)	
Vérifier le fonctionnement de l'algorithme concernant les cas des autres cases : prise de la première case disponible.	
Vérification de l'affichage du coup joué : - Vérifier le branchement au module jouer_coup (X, Y, couleurOrdinateur) - Vérifier la prise en compte du coup joué dans les statistiques	
Vérifier le cas où l'ordinateur ne peut pas jouer : - vérifier l'affichage d'une fenêtre d'information : « le joueur doit passer son tour »	

<u>Module</u> :	ordinateur_intermediaire	18/04/1997
<u>Rôle</u> :	Fonction permettant de déterminer le coup de l'ordinateur pour le niveau débutant.	
<u>Entrées</u> :	CouleurOrdinateur.	
<u>Sorties</u> :	-	

CONTROLES :	RESULTATS :
Vérifier la réalisation d'un tableau priorisant les coups par rapport au damier	
Vérifier l'initialisation correcte des variables.	
Vérifier le fonctionnement de l'algorithme de réflexion de l'ordinateur pour ce niveau : vérifier la cohérence du choix du coup	
Vérifier la possibilité de retournement des pions par le branchement au module retourne ()	
Vérifier l'affichage du coup joué : Vérifier le branchement au module Jouer_coup ()	
Vérifier la prise en compte du coup joué dans les statistiques	

<u>Module</u> :	evaluation_damier	18/04/1997
<u>Rôle</u> :	Fonction permettant d'évaluer un damier afin de déterminer la situation du jeu en lui donnant une note	
<u>Entrées</u> :	Damier : damier de jeu à évaluer	
<u>Sorties</u> :	Note du damier.	

CONTROLES :	RESULTATS :
Vérifier l'initialisation des variables (tableaux de scores d'une case jouée par un joueur Noir et Blanc, Note)	
Vérification de la détermination de la fin de jeu (branchements au module coup_possibles () pour le joueur Noir et le joueur Blanc)	
Cas : si fin du jeu Vérification de la détermination du gagnant en balayant le damier et en comptant les pions noirs et blancs. Vérification de l'affectation de la note du damier en fonction de quelle couleur est victorieuse.	
Cas : pas en fin de jeu Vérification de la cohérence du calcul de la note du damier et de son affectation	

Module :	alpha_beta	18/04/1997
Rôle :	Fonction permettant déterminer le meilleur coup de l'ordinateur en cherchant en profondeur en garantissant d'avoir le Pième coup à venir (P=profondeur) une note égale à l'un des coups joués à la profondeur P NOTA : C'est une fonction récursive	
Entrées :	Damier Profondeur (de la recherche) Joueur X, Y : coordonnées du coup joué par le joueur Score note : note du coup qui : changement d'affectation des joueurs : utilisé pour la génération de l'arbre	
Sorties :	-	

CONTROLES :	RESULTATS :
Vérifier la détermination des possibilités de jeu des deux joueurs par les branchements au module coups_possible () et evaluation_damier ()	
Vérification du fonctionnement de l'algorithme alpha-beta - vérification des appels récursifs au module alpha_beta et des appels (non récursifs) au module evaluation_damier () - vérification du fonctionnement des procédures MIN et MAX	

Module :	ordinateur_expert	21/05/1997
Rôle :	Fonction permettant déterminer le coup de l'ordinateur pour les hauts niveaux	
Entrées :	CouleurOrdinateur couleurJoueur profondeur (de la recherche)	
Sorties :	-	

CONTROLES :	RESULTATS :
Vérification de l'affichage du curseur « montre » durant la réflexion de l'ordinateur	
Recherche du meilleur coup avec la méthode alpha-beta : vérifier le branchement au module alpha-beta	
Vérifier le branchement au module retourne ()	
Vérification de l'affichage du coup joué : vérifier le branchement au module jouer_coup	
Vérification de la prise en compte du coup joué dans les statistiques	

<u>Module :</u>	statistique	16/04/1997
<u>Rôle :</u>	Fonction permettant de calculer les statistiques des différents joueurs et de réaliser l'affichage vers les zones adéquates	
<u>Entrées :</u>	Joueur_Courant : couleur du joueur demandant de recalculer les statistiques	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de la définition du score de chaque joueur : vérifier le branchement au module compte_pions () (pour le comptage des pions)	
Vérification de la définition de la mobilité des joueurs : détermination du nombre de coups possible de chaque joueurs (joueur courant et joueur adverse) par les appels au module coups_possibles ()	
Vérification de la définition de l'évaluation du coup du joueur courant par l'appel du module evaluation_damier ()	
Vérification du nombre de feuilles analysées en recherche α - β (variable Appel)	
Vérification de l'affichage des statistiques pour les joueurs (noir et blanc) par l'appel du module Affiche_statistique () pour les deux joueurs	

<u>Module :</u>	statistique_affiche	18/05/1997
<u>Rôle :</u>	Fonction permettant d'afficher les statistiques	
<u>Entrées :</u>	Joueur_Courant : couleur du joueur demandant de recalculer les statistiques	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifications des affichages des données statistiques suivantes : - dernier coup joué - mobilité - l'évaluation du damier - la profondeur et le nombre de feuilles (uniquement si recherche α - β)	
Pour l'affichage de ces données, vérifier les branchements au module statistique_affiche_donnée ()	
Vérifier pour chaque statistique, la cohérence des informations et leur situation d'affichage.	

<u>Module :</u>	initiation	20/04/1997
<u>Rôle :</u>	Fonction permettant d'apprendre à jouer pour un joueur inexpérimenté	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification du fonctionnement de la fenêtre de demande du choix de la couleur.	
Vérification de l'affectation du type de jeu	
Initialisation du jeu : Vérifier le branchement au module initialise_jeu ()	
Vérifier le branchement au module jeu_2joueurs ()	

<u>Module :</u>	affiche_coup	20/04/1997
<u>Rôle :</u>	Fonction permettant d'afficher les coups possibles pour le joueur (niv. initiation)	
<u>Entrées :</u>	joueur : joueur pour lequel on affiche les coups etat : VRAI : coup visible - FAUX : coup non visible	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de la détermination du nombre de coups possible par l'appel au module coup_possibles ()	
Vérification de l'affichage de points noir ou blancs en fonction de la couleur du joueur et des coups possibles (tests Blanc - Noirs)	

<u>Module :</u>	fin_jeu	23/02/1997
<u>Rôle :</u>	Fin de la partie : un joueur a gagné / arrêt de la partie.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Si Match nul :	
Vérification de l'affichage dans une boîte de dialogue (« match nul »)	
Si pas match nul :	
Vérification de l'affichage dans une boîte de dialogue : « le joueur X a gagné avec x points d'écart » (à tester avec joueur Blanc vainqueur et joueur noir vainqueur)	
Vérification de la fermeture de la boîte avec le bouton OK	
Vérification du remplissage du damier avec la couleur du pion du joueur gagnant.	

Module :	visualiser	06/05/1997
Rôle :	Fonction permettant de visualiser une partie sauvegardée	
Entrées :	-	
Sorties :	-	

CONTROLES :	RESULTATS :
Vérification de l'initialisation de l'environnement de jeu : vérifier le branchement au module initialise_ecran_jeu ()	
Vérification de la mise à zéro ou de l'initialisation des variables	
Vérification du fonctionnement du système de saisie de la partie à visualiser : - Vérifier l'affichage d'une boîte de dialogue et saisie du nom de la partie - Vérifier l'affichage en aide en ligne : « chargement en cours » - Vérifier le branchement au module Charge_partie ()	
Chargement de la partie = OK	
Vérifier les actions suivantes si le chargement est OK - affichage du nom des joueurs - initialisation de début de jeu : vérifier le branchement au module statistique ()	
Boucle : tant qu'on ne quitte pas le jeu	
Vérifier le branchement au module Initialise_tableau_commande ()	
Vérifier que le bouton conseil est inactif.	
Si click souris sur le bouton « coup précédent » : vérifier le branchement au module coup_precedent ()	
Si click souris sur le bouton « coup suivant » : vérifier le branchement au module coup_suivant ()	
Si click souris sur le bouton « stop » : définition de la fin du jeu	
Jusqu'à définition de la fin du jeu	
Chargement = NOK :	
Vérifier qu'il n'y pas d'actions réalisées (retour à la procédure appelante)	

<u>Module :</u>	apropos	22/03/1997
<u>Rôle :</u>	Affichage des informations concernant les concepteurs du jeu OTHO (c'est nous !!).	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification à l'écran du chargement de l'image « \images\apropos.gif »	
Vérification : aucun curseur souris à l'écran pendant l'attente de 2 secondes	
Vérification de la disparition de l'image sur les événements : - appui clavier - click souris - fin de la temporisation	

<u>Module :</u>	regles	18/05/1997
<u>Rôle :</u>	Fonction permettant de gérer les écrans de règles du jeu.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de l'affichage de la page courante des règles du jeu : - vérifier le branchement au module regle_affiche (Page_Courante) - vérifier l'affichage (et la cohérence) du n° de la page courante - vérifier l'affichage des boutons de sélection des pages par le branchement au module regle_affiche_bouton ()	
Vérifier que le bouton « suivant » est inactif en dernière page :	
Vérifier que le bouton « précédent » est inactif en première page	
Vérifier la bonne gestion des pages en fonction des clicks souris sur les boutons « suivant » et « précédent »	

<u>Module :</u>	regles_affiche_bouton	02/05/1997
<u>Rôle :</u>	Fonction permettant d'afficher les boutons pour les écrans « règles du jeu »	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier l'affichage des boutons « Précédent » - « Sortir » et « Suivant » dans les écrans des règles du jeu (tous)	
Vérifier la gestion de type appuyé / relâché en cliquant sur ces boutons.	

<u>Module :</u>	regles_affiche	02/05/1997
<u>Rôle :</u>	Fonction permettant d'afficher l'écran courant des règles du jeu.	
<u>Entrées :</u>	PAGE : Page courante à afficher	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier que le chargement de l'image demandée est correct.	

<u>Module :</u>	sauve_partie	21/05/1997
<u>Rôle :</u>	Fonction permettant de sauvegarder une partie	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
<u>Vérifier le cas où tous les numéros de fichiers sont pris :</u> Affichage sous la forme d'une boîte de dialogue du message : « dépassement de la capacité » et sortie de la boîte par un bouton OK	
<u>Vérifier le cas où il reste des numéros de fichiers libres :</u> - vérification de la création d'un fichier de sauvegarde - vérification de l'affichage sous forme de boîte de dialogue (partie enregistrée sous le nom othxxxx.dat) et sortie de la boîte par un bouton OK	
<u>Hors fonctionnement :</u> vérifier la prise en compte du fichier dans le répertoire data du disque dur.	

<u>Module :</u>	charge_partie	21/05/1997
<u>Rôle :</u>	Fonction permettant de charger une partie	
<u>Entrées :</u>	Nom du fichier contenant les données de la partie	
<u>Sorties :</u>	Résultat du chargement (OK - NOK)	

CONTROLES :	RESULTATS :
<p><u>Ouverture du fichier (\\data\Nom du fichier.dat) possible :</u></p> <p>Vérifier le montage en mémoire des éléments enregistrés : [type de partie - nom des joueurs - indices du jeu - tous les coups de la partie]</p> <p>Vérifier leur cohérence par rapport à leur enregistrement</p>	
<p><u>Ouverture du fichier (\\data\Nom du fichier.dat) impossible :</u></p> <p>Vérifier l'affichage à l'écran d'une boîte de dialogue : « Impossible d'ouvrir le fichier » et sortie de la boîte par click sur bouton OK</p> <p>Vérifier l'affectation du résultat du chargement : NOK</p>	

<u>Module :</u>	statistique_affiche_donnée	18/05/1997
<u>Rôle :</u>	Fonction de bas niveau permettant d'afficher en fonction de son entrée une donnée de statistique en cours de jeu sur l'écran (prise en compte des coordonnées à réaliser)	
<u>Entrées :</u>	Donnée à afficher.	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier l'affichage des données au bon emplacement sur l'écran.	

<u>Module :</u>	high_score	21/05/1997
<u>Rôle :</u>	Fonction permettant de gérer les high-scores	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier le fonctionnement de l'algorithme de détermination du gagnant	
<p><u>Joueur Humain vainqueur :</u></p> <p>Vérifier qu'il y a bien comparaison de son score avec les autres scores des highscores Si le score est suffisant :</p> <ul style="list-style-type: none"> - vérifier qu'il y a bien insertion du score dans les meilleurs scores et décalage des autres scores - vérifier le branchement au module sauve_score () - vérifier le branchement au module affiche_high_score () <p>Si le score est insuffisant : vérifier qu'aucun traitement n'est effectué</p>	
<p><u>Joueur Humain perdant :</u></p> <p>Vérifier qu'aucun traitement dans ce module n'est effectué</p>	

<u>Module :</u>	affiche_high_score	18/05/1997
<u>Rôle :</u>	Fonction permettant d'afficher le tableau complet des high-scores	
<u>Entrées :</u>	NomEcran : permet de savoir si l'affichage du tableau est commandée de l'écran de jeu ou du menu principal.	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier le chargement de l'image « \images\otho.gif » et son affichage sur l'écran.	
Vérifier l'affichage immédiat du tableau des high-scores avec, pour chaque ligne : le nom du joueur, le score des noirs et le score des blancs et la date (affichage réalisé pour chaque ligne par le branchement au module Statistique_affiche_données ()).	
Vérification de la gestion d'un bouton OK sur le tableau des meilleurs scores :	

<u>Module :</u>	sauve_score	18/05/1997
<u>Rôle :</u>	Fonction permettant de sauvegarder sur un fichier les scores du jeu	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
<u>Cas de non-existence du fichier « data\scores.dat »</u>	
Vérifier qu'il y a création du fichier scores.dat dans le répertoire data, puis écriture dans le fichier de la ligne correspondante au high-score réalisé	
S'il y a un problème quelconque à la création: noter l'affichage d'un message d'erreur	
<u>Cas d'existence du fichier « data\scores.dat »</u>	
Vérifier qu'il y a écriture dans le fichier de la ligne correspondante au high-score réalisé	

<u>Module :</u>	charge_score	18/05/1997
<u>Rôle :</u>	Fonction permettant de monter en mémoire le fichier High-score	
<u>Entrées :</u>	-	
<u>Sorties :</u>	résultat du chargement (OK - NOK)	

CONTROLES :	RESULTATS :
<u>Ouverture du fichier (data\scores.dat) possible :</u>	
Vérifier le montage en mémoire de tous les éléments enregistrés et leur cohérence par rapport aux valeurs réellement enregistrées auparavant. Vérifier aussi l'affectation du résultat de chargement : OK	
<u>Ouverture du fichier (data\scores.dat) impossible :</u>	
Vérifier l'affichage à l'écran d'une boîte de dialogue : « Impossible d'ouvrir le fichier » et sortie de la boîte par click sur bouton OK	
Vérifier l'affectation du résultat du chargement : NOK	

<u>Module :</u>	ouverture	19/05/1997
<u>Rôle :</u>	Fonction permettant d'accéder à la bibliothèque des ouvertures.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérifier le chargement de l'image « \images\otho.gif » et son affichage sur l'écran.	
Vérifier l'affichage immédiat de la boîte de dialogue de choix de l'ouverture et de la boîte d'aide en ligne (vide)	
« préparation » de la bibliothèque d'ouvertures : OK	
Vérifier l'affichage en aide en ligne : « Faites votre choix » Vérifier le branchement au module selection_ouverture () Vérifier l'affichage en aide en ligne: du nom de l'ouverture et vérifier le branchement au module ouverture_visualise ()	
« préparation » de la bibliothèque d'ouvertures : NOK	
Vérifier qu'il n'y a pas d'actions réalisées (c'est un cas d'erreur)	

<u>Module :</u>	prepare_ouverture	19/05/1997
<u>Rôle :</u>	Fonction permettant de vérifier l'existence des fichiers de données et de créer éventuellement la bibliothèque.	
<u>Entrées :</u>	-	
<u>Sorties :</u>	-	

CONTROLES :	RESULTATS :
Vérification de l'existence du fichier « \data\ouvertur.dat »	
<u>Le fichier n'existe pas :</u>	
Vérifier la création à partir du fichier « data\ouvertur.txt »	
Si le fichier n'est pas trouvé sur disque : vérifier l'affichage en aide en ligne : « le fichier des ouvertures n'existe pas !! »	
Si le fichier est trouvé sur le disque : Vérifier sur disque la transformation du fichier .txt en .dat	
Tester le cas d'un problème de transformation : (bricolage du fichier) Vérifier l'affichage en aide en ligne : « il y a un léger problème ... » Vérifier qu'il n'y a pas de plantage	
<u>Le fichier existe :</u>	
Vérifier qu'aucune action n'est réalisée	

Module :	ouverture_visualise	20/05/1997
Rôle :	Fonction permettant de visualiser l'ouverture sélectionnée.	
Entrées :	selection : indice dans la table ouverture	
Sorties :	-	

CONTROLES :	RESULTATS :
Vérifier l'initialisation des coups de l'ouverture à jouer avec l'apparition de l'affichage en aide en ligne « l'ouverture comporte », nombreCoups » « coups. ».	
Vérifier le branchement aux modules initialise_ecran_jeu () et initialise_Jeu ()	
Affichage des statistiques de début de jeu : Vérifier le branchement au module statistique ()	
Boucle de jeu	
Vérifier le branchement au module Initialise_tableau_commande ()	
Vérifier l'inactivité du bouton « conseil ».	
Si click souris sur le bouton « coup précédent » : Vérifier le branchement au module coup_precédent () Vérifier l'affichage en aide en ligne : « Coup courant de l'ouverture », « nombrecoup/totalcoup » et sa cohérence ou l'affichage en aide en ligne « Vous êtes au début de l'ouverture !! »	
Si click souris sur le bouton « coup suivant » : Vérification de coup possible (pas arrivé à la fin des coups de l'ouverture) Si oui : Vérifier la lecture cohérente du coup à jouer Vérifier le branchement au module retourne () Vérifier le branchement au module jouer_coup () Vérifier le calcul des statistiques pour le coup réalisé Vérifier la possibilité de passage au coup suivant Vérifier le branchement au module statistique () Vérifier le branchement au module affiche_pions () Vérifier le branchement au module memorise_jeu () Vérifier le branchement au module change_joueur () Vérifier l'affichage en aide en ligne du coup courant de l'ouverture Si non : Vérifier l'affichage en aide en ligne : « dernier coup de l'ouverture atteint !! »	

CONTROLES :	RESULTATS :
--------------------	--------------------

Si click souris sur le bouton « stop » : définition de la fin du jeu (sortie de la boucle)	
Affichage des coups possibles : Vérifier le branchement au module affiche_coup ()	
Fin de la boucle de jeu	

<u>Module :</u>	selection_ouverture	19/05/1997
<u>Rôle :</u>	Gestion de la sélection de l'ouverture avec la souris	
<u>Entrées :</u>	-	
<u>Sorties :</u>	NomOuverture	

CONTROLES :	RESULTATS :
Vérifier que la préparation des pages d'ouvertures est nominale	
<u>tant que pas clic sur Annuler ou OK</u>	
Vérifier l'affichage de la page courante si une nouvelle page a été demandée Si on est en début de consultation, vérifier que c'est la première page qui est affichée.	
Si Click sur bouton >> : Vérifier le changement de page (page suivante)	
Si Click sur bouton << : Vérifier le changement de page (page précédente)	
Si Click sur bouton + : Si on est à la dernière ligne de la page : vérifier qu'il n'y pas de changement d'ouverture sinon vérifier la sélection de la ligne suivante et passage en surbrillance du nom de l'ouverture	
Si Click sur bouton - : Si on est à la première ligne de la page : vérifier qu'il n'y pas de changement d'ouverture sinon vérifier la sélection de la ligne précédente et passage en surbrillance du nom de l'ouverture	
Si Click sur bouton OK : Vérifier la prise en compte de la sélection	
<u>Fin tant que</u>	

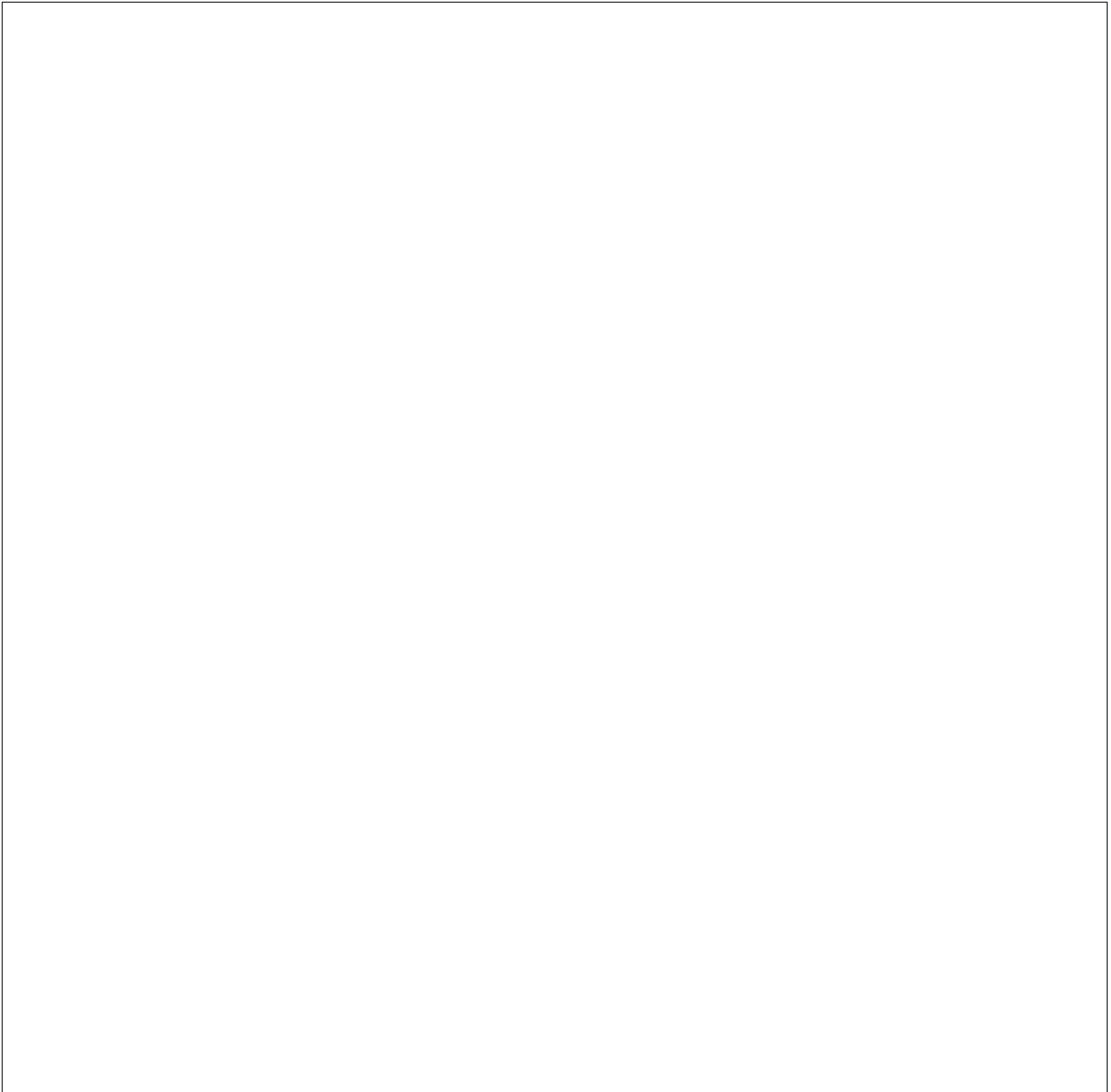
4. CONCLUSIONS

4.1 Couverture de test :

Tous les modules de haut niveau du logiciel OTHO sont testés.

En ce qui concerne les modules de bas niveau, leur « testabilité » est prise en compte dans les tests des modules de haut niveau. En effet, ce sont pour la plupart des modules de gestion d'affichage : il suffit juste de voir la cohérence des données et leur positionnement sur l'écran, ce qui est pris en compte dans le libellé des tests à réaliser.

4.2 Conclusions sur les tests unitaires - OTHO version 1.0a:



5. MATRICE DE TRACABILITE DES EXIGENCES :

Cette matrice est le rappel de la matrice figurant dans le DCD Indice A. Elle n'a subi aucun changement.

SERVICE (CC)	EXIGENCE (STB)	FONCTIONS (DCP)	MODULES (DCD)
INITIALISER	[INI.1]	[ACC.1] [ACC.2]	[materiel] [allocation_memoire] [ecran_presentation]
	[INI.2]	[MEN.1] [MEN.2] [MEN.3] [REG.1] [REG.2] [REG.3] [PRO.1] [PRO.2] [PRO.3]	[menu_principal] [menu_principal] [menu_principal] [regles] [regle_affiche_bouton] [regles] [regle_affiche] [apropos] [apropos] [apropos]
	[INI.3]	[PAR.1] [PAR.2] [PAR.3]	[jouer] [initialise_ecran_jeu] [choix_joueur] [jeu_normal] [initiation] [jeu_normal]
JOUER UN COUP0	[JOU.1]	[GRI.1] [GRI.2]	[jeu_2joueur] [statistique] [initialise_tableau_commande]
	[JOU.2]	[PLA.1] [PLA.2]	[jeu_2joueur] [ordinateur_debutant] [ordinateur_intermediaire] [ordinateur_expert] [ordinateur_debutant] [evaluation_damier] [alpha_beta] [coup_possible] [change_joueur]
	[JOU.3]	[PLA.3]	
	*	[PLA.4]	[jeu_2joueur]
		[PLA.5] [PLA.6] [PLA.7]	[jeu_2joueur] [conseil_coup] [jeu_2joueur]
	[JOU.4]	[VER.1]	[jeu_2joueur] [jouer_coup] [retourne] [retourne]
	[JOU.5]	[VER.2] [COU.1] [COU.2] [COU.3] [COU.4]	[jouer_coup] [retourne] [retourne] [statistique] [statistique_affiche]
	[JOU.6]	[COU.1] [COU.2] [COU.3] [COU.4]	[jouer_coup] [retourne] [retourne] [statistique] [statistique_affiche]
[JOU.7]	[FIN.1]	[fin_jeu]	
GERER LES SCORES	[SCO.1]	[CAL.1] [CAL.2]	[compte_pions] [statistique]
	[SCO.2]	[VSC.1] [VSC.2] [PPE.1] [PPE.2] [PGA.1] [PGA.2] [PGH.1] [PGH.2]	[fin_jeu] [high_scores] [fin_jeu] [high_scores] [fin_jeu] [high_scores] [affiche_ecran_score] [fin_jeu] [high_scores] [sauve_score]
	[SCO.3]	[ATA.1]	[affiche_ecran_score]

SERVICE (CC)	EXIGENCE (STB)	FONCTIONS (DCP)	MODULES (DCD)
GERER LA BIBLIOTHEQUE	[BIB.1]	[CON.1] [LAN.1] [LAN.2] [SAU.1] [SAU.2]	[jeu_2joueur] [sauve_partie] [sauve_partie]
	[BIB.2]	[CHO.1] [CHO.2] [CHO.3] [CHO.4] [CHO.5] [CHO.6] [CHO.7] [VIS.1] [VIS.2] [VIS.3]	[visualiser] [initialise_ecran_jeu] [visualiser] [charge_partie] [charge_partie] [visualiser] [charge_partie] [ouverture] [prepare_ouverture] [selection_ouverture] [prepare_ouverture] [visualiser] [initialise_ecran_jeu] [initialise_tableau_commande] [ouverture_visualise] [coup_precedent] [coup_suivant] [fin_jeu]
	[BIB.3]	[REP.1] [REP.2] [REP.3] [REP.4]	[jeu_normal] [charge_partie] [charge_partie] [charge_partie] [jeu_2joueur]
Exigences de développement	[DEV.1]	<i>Non développé dans le DCP</i>	<i>developpé dans le § 5.1</i>
	[DEV.2]	<i>Non développé dans le DCP</i>	<i>du fait de l'utilisation de Turbo C</i>
	[DEV.3]	<i>Non développé dans le DCP</i>	<i>developpé dand le § 5.1</i>

Tous les modules ont été testés, mais certains ne sont pas présents dans cette matrice du fait de rajouts de fonctionnalités comme le module suivant :

- **ordinateur_vs_ordinateur** : jeu ordinateur contre ordinateur (niveaux différents);

ou du fait que ce soit des fonctions de plus bas niveau (non apparentes sur la matrice car ces fonctions sont appelées par jeu_2joueur, par exemple)

- **initialise_jeu**
- **affiche_pions**
- **change_joueur**
- **memorise_jeu**
- **affiche_coup**
- **statistique_affiche_donnees**

Il est à noter que les exigences de développement [DEV.1 à 3] ont été en partie vérifiées lors des tests unitaires (voir § 3.2)