

nom	retour attendu	code d'erreur	est passé	réfèrent
Constantes				
testPorteeTireurDebut	validé si le tireur a bien la portée attendue en début de partie	#001	OK	Matthieu
testPorteeTireurCours	validé si le tireur n'as pas changé de portée attendue au cours de cette même partie	#002	OK	Matthieu
testPorteePiegeurDebut	validé si le piègeur a bien la portée attendue en début de partie	#003	OK	Matthieu
testPorteePiegeurCours	validé si le piègeur n'as pas changé de portée au cours de cette même partie	#004	OK	Matthieu
testPorteeCharDebut	validé si le char a bien la portée attendue en début de partie	#005	OK	Matthieu
testPorteeCharCours	validé si le char n'as pas changé de portée au cours de cette même partie	#006	OK	Matthieu
testDeplacementTireurDebut	validé si le tireur peut se déplacer de manière attendue au début de la partie	#007	OK	Matthieu
testDeplacementTireurCours	validé si le tireur n'as pas changé de contraintes de déplacement au cours de cette même partie	#008	OK	Matthieu
testDeplacementPiegeurDebut	validé si le piègeur peut de déplace de manière attendue au début de la partie	#009	OK	Matthieu
testDeplacementPiegeurCours	validé si le piègeur n'as pas changé de contraintes de déplacement au cours de cette même partie	#010	OK	Matthieu
testDeplacementCharDebut	validé si le char peut se déplacer de manière attendue au début de la partie	#011	OK	Matthieu
testDeplacementCharCours	validé si le char n'as pas changé de contraintes de déplacement au cours de cette même partie	#012	OK	Matthieu
testEnergieInitialeTireur	validé si le tireur a l'énergie initiale attendue au début de la partie	#013	OK	Matthieu
testEnergieInitialePiegeur	validé si le piègeur a l'énergie initiale attendue au début de la partie	#014	OK	Matthieu
testEnergieInitialeChar	validé si le char a l'énergie initiale attendue au début de la partie	#015	OK	Matthieu
testBaseTireurDebut	validé si la base régénère les tireurs positionnés dessus de manière attendue au début de la partie	#016	OK	Matthieu
testBaseTireurCours	validé si la base n'as pas changé de mode de régénération pour le tireur au cours de cette même partie	#017	OK	Matthieu
testBasePiegeurDebut	validé si la base régénère les piègeurs positionnés dessus de manière attendue au début de la partie	#018	OK	Matthieu
testBasePiegeurCours	validé si la base n'as pas changé de mode de régénération pour le piègeur au cours de cette même partie	#019	OK	Matthieu
testBaseCharDebut	validé si la base régénère les chars positionnés dessus de manière attendue au début de la partie	#020	OK	Matthieu
testBaseCharCours	validé si la base n'as pas changé de mode de régénération pour le char au cours de cette même partie	#021	OK	Matthieu
testBaseTireurFullLifeDebut	validé si la base ne regénère pas plus de points d'énergie que le nombre de points de vie maximum du tireur au début de la partie	#022	OK	Matthieu
testBaseTireurFullLifeCours	validé si la base ne regénère toujours pas plus de points d'énergie que le nombre de points de vie maximum du tireur au cours de cette même partie	#023	OK	Matthieu
testBasePiegeurFullLifeDebut	validé si la base ne regénère pas plus de points d'énergie que le nombre de points de vie maximum du piègeur au début de la partie	#024	OK	Matthieu
testBasePiegeurFullLifeCours	validé si la base ne regénère toujours pas plus de points d'énergie que le nombre de points de vie maximum du piègeur au cours de cette même partie	#025	OK	Matthieu
testBaseCharFullLifeDebut	validé si la base ne regénère pas plus de points d'énergie que le nombre de points de vie maximum du char au début de la partie	#026	OK	Matthieu
testBaseCharFullLifeCours	validé si la base ne regénère toujours pas plus de points d'énergie que le nombre de points de vie maximum du char au cours de cette même partie	#027	OK	Matthieu
testMinerPiegeurDebut	validé si le nombre de points d'énergie perdue après avoir posé une mine est bien celui attendu au début de la partie	#028	OK	Matthieu
testMinerPiegeurCours	validé si le nombre de points d'énergie perdue après avoir posé une mine n'as pas changé au cours de cette même partie	#029	OK	Matthieu
testMinerTireurDebut	validé si le tireur n'as pas la possibilité de poser une mine au début de la partie	#030	OK	Matthieu
testMinerTireurCours	validé si le tireur n'as toujours pas la possibilité de poser une mine au cours de cette même partie	#031	OK	Matthieu
testMinerCharDebut	validé si le char n'as pas la possibilité de poser une mine au début de la partie	#032	OK	Matthieu
testMinerCharCours	validé si le char n'as toujours pas la possibilité de poser une mine au cours de cette même partie	#033	OK	Matthieu
testTirerTireurDebut	validé si le nombre de points d'énergie perdue après un tir du tireur est bien celui attendu au début de la partie	#034	OK	Matthieu
testTirerTireurCours	validé si le nombre de points d'énergie perdue après un tir du tireur n'as pas changé au cours de cette même partie	#035	OK	Matthieu
testTirerCharDebut	validé si le nombre de points d'énergie perdue après un tir du char est bien celui attendu au début de la partie	#036	OK	Matthieu
testTirerCharCours	validé si le nombre de points d'énergie perdue après un tir du char n'as pas changé au cours de cette même partie	#037	OK	Matthieu
testTirerPiegeurDebut	validé si le piègeur n'as pas la possibilité de tirer au début de la partie	#038	OK	Matthieu
testTirerPiegeurCours	validé si le piègeur n'as toujours pas la possibilité de tirer au cours de cette même partie	#039	OK	Matthieu
testAvancerTireurDebut	validé si le nombre de points d'énergie perdu après que le tireur ait avancé est bien celui attendu au début de la partie	#040	OK	Matthieu
testAvancerTireurCours	validé si le nombre de points d'énergie perdu après que le tireur ait avancé n'as pas changé au cours de cette même partie	#041	OK	Matthieu
testAvancePiegeurDebut	validé si le nombre de points d'énergie perdu après que le piègeur ait avancé est bien celui attendu au début de la partie	#042	OK	Matthieu
testAvancePiegeurCours	validé si le nombre de points d'énergie perdu après que le piègeur ait avancé n'as pas changé au cours de cette même partie	#043	OK	Matthieu
testAvanceCharDebut	validé si le nombre de points d'énergie perdu après que le char ait avancé est bien celui attendu au début de la partie	#044	OK	Matthieu
testAvanceCharCours	validé si le nombre de points d'énergie perdu après que le char ait avancé n'as pas changé au cours de cette même partie	#045	OK	Matthieu

testDegatsTireurDebut	validé si le nombre de points d'énergie perdu par le tireur après avoir subi un tir ou marché sur une mine est bien celui attend au début de la partie	#046	OK	Matthieu
testDegatsTireurCours	validé si le nombre de points d'énergie perdu par le tireur après avoir subi un tir ou marché sur une mine n'as pas changé au cours de cette même partie	#047	OK	Matthieu
testDegatsPiegeurDebut	validé si le nombre de points d'énergie perdu par le piègeur après avoir subi un tir ou marché sur une mine est bien celui attend au début de la partie	#048	OK	Matthieu
testDegatsPiegeurCours	validé si le nombre de points d'énergie perdu par le piègeur après avoir subi un tir ou marché sur une mine n'as pas changé au cours de cette même partie	#049	OK	Matthieu
testDegatsCharDebut	validé si le nombre de points d'énergie perdu par le char après avoir subi un tir ou marché sur une mine est bien celui attend au début de la partie	#050	OK	Matthieu
testDegatsCharCours	validé si le nombre de points d'énergie perdu par le char après avoir subi un tir ou marché sur une mine n'as pas changé au cours de cette même partie	#051	OK	Matthieu
Plateau				
testDimensionsDebut	validé si le plateau est bien de la dimension entrée en paramètres au début de la partie	#052	OK	Matthieu
testDimensionsCours	validé si le plateau fait toujours la même dimension au cours de cette même partie	#053	OK	Matthieu
testObstacles	validé si le plateau contient bien le bon pourcentage d'obstacles entré en paramètre au début de la partie	#054	OK	Matthieu
testBasesDebut	validé si les bases sont situées aux extrémités d'une diagonale au début de la partie	#055	OK	Matthieu
testBasesCours	validé si les bases sont situées sur les même cases qu'au début de cette même partie	#056	OK	Matthieu
testSortiImpossibleDebut	validé si la sortie de plateau par chaque côté est impossible au début de la partie	#057	OK	Matthieu
testSortiImpossibleTireur	validé si un tireur ne peut pas sortir du plateau en cours de partie	#058	OK	Matthieu
testSortiImpossiblePiegeur	validé si un piègeur ne peut pas sortir du plateau en cours de partie	#059	OK	Matthieu
testSortiImpossibleChar	validé si un char ne peut pas sortir du plateau en cours de partie	#060	OK	Matthieu
testToucherBaseTireurDebut	validé si le tireur ne peut se faire toucher dans sa base en début de partie	#061	OK	Matthieu
testToucherBasePiegeurDebut	validé si le piegeur ne peut se faire toucher dans sa base en début de partie	#062	OK	Matthieu
testToucherBaseCharDebut	validé si le char ne peut se faire toucher dans sa base en début de partie	#063	OK	Matthieu
testToucherBaseTireurCours	validé si le Tireur réfugié dans sa base ne peut pas se faire toucher en cours de partie	#064	OK	Matthieu
testToucherBasePiegeurCours	validé si le piegeur réfugié dans sa base ne peut pas se faire toucher en cours de partie	#065	OK	Matthieu
testToucherBaseCharCours	validé si le char réfugié dans sa base ne peut pas se faire toucher en cours de partie	#066	OK	Matthieu
testVisibiliteBaseTireurDebut	validé si le tireur n'est pas visible dans sa base en début de partie	#067	OK	Matthieu
testVisibiliteBasePiegeurDebut	validé si le piegeur n'est pas visible dans sa base en début de partie	#068	OK	Matthieu
testVisibiliteBaseCharDebut	validé si le char n'est pas visible dans sa base en début de partie	#069	OK	Matthieu
testVisibiliteBaseTireurCours	validé si le tireur réfugié dans sa base n'est pas visible en cours de partie	#070	OK	Matthieu
testVisibiliteBasePiegeurCours	validé si le piegeur réfugié dans sa base n'est pas visible en cours de partie	#071	OK	Matthieu
testVisibiliteBaseCharCours	validé si le char réfugié dans sa base n'est pas visible en cours de partie	#072	OK	Matthieu
testAccessDeniedTireurDebut	validé si le tireur n'as pas possibilité d'accéder à la base ennemie en début de partie	#073	OK	Matthieu
testAccessDeniedPiegeurDebut	validé si le piegeur n'as pas possibilité d'accéder à la base ennemie en début de partie	#074	OK	Matthieu
testAccessDeniedCharDebut	validé si le char n'as pas possibilité d'accéder à la base ennemie en début de partie	#075	OK	Matthieu
testAccessDeniedTireurCours	validé si le tireur n'as toujours pas possibilité d'accéder à la base ennemie en cours de partie	#076	OK	Matthieu
testAccessDeniedPiegeurCours	validé si le piegeur n'as toujours pas possibilité d'accéder à la base ennemie en cours de partie	#077	OK	Matthieu
testAccessDeniedCharCours	validé si le char n'as toujours pas possibilité d'accéder à la base ennemie en cours de partie	#078	OK	Matthieu
testCharPasseSurMineDebut	validé si le char qui passe sur une mine dans son déplacement prend des dégats en début de partie	#079	OK	Matthieu
testCharPasseSurMineCours	validé si le char qui passe sur une mine dans son déplacement prend des dégats en cours de partie	#080	OK	Matthieu
testVisibiliteMinesDebut	validé si on voit les mines de son equipe en début de partie	#081	OK	Matthieu
testVisibiliteMinesCours	validé si on voit toujours les mines de son équipe en cours de partie	#082	OK	Matthieu
testNonVisibiliteMinesDebut	validé si on ne voit pas les mines de l'équipe adverse en début de partie	#083	OK	Matthieu
testNonVisibiliteMinesCours	validé si on ne voit toujours pas les mines de l'équipe adverse en cours de partie	#084	OK	Matthieu
testDispartitionMineDebut	validé sur une mine disparaît bien si on passe dessus en début de partie	#085	OK	Matthieu
testDispartitionMineCours	validé sur une mine disparaît bien si on passe dessus en cours de partie	#086	OK	Matthieu
testdimensioncarte				
testdimensioncarte	validé si la carte créée fait bien les dimensions que l'on a mises en paramètre			
Attaques				
testposemine	validé si la mine est posée	#087	OK	Benjamin

testposemineendroitindiquer	validé si la mine est posée à l'endroit valide indiqué	#088	OK	Benjamin
testposemineobstacle	validé si la mine n'est pas posée sur l'obstacle ciblé (robot ruine décor,...)	#089	OK	Benjamin
testposeminetroploin	validé si la mine n'est pas posée en dehors de la zone ou le piègeur peut la poser	#090	OK	Benjamin
testposetropdemine	validé si plusieurs mines ne peuvent pas être posées par un même piègeur durant le même tour	#091	OK	Benjamin
testposemineyenadejaune	échoue si une mine peut être posé a un endroit où se trouve déjà une mine	#092	OK	Benjamin
testminestockapurien	validé si le piègeur ne peut pas poser de mine quand son stock est à 0.	#093	OK	Benjamin
testminestockposey	validé si une mine est retirée du stock du piègeur quand il l'a posée	#094	OK	Benjamin
testminestocktenatrop	validé si le nombre de mines qu'à un piègeur dans son stock ne dépasse pas le maximum possible.	#095	OK	Benjamin
testminestockrecharge	validé si le stock de mine du piègeur a été remis au max dans sa base.	#096	OK	Benjamin
testminexplose	validé si la mine explose quand un robot est sur la case où elle est posée	#097	OK	Benjamin
testminexplosedamage	validé si la mine explose et fait des dommages au robot qui l'a enclenché	#098	OK	Benjamin
testminexplosedommagetroploin	validé si la mine qui explose ne fait pas de dommage a d'autres robots que celui qui l'a déclenchée	#099	OK	Benjamin
testminexplosedisparait	validé si une mine qui a explosé disparaît de la carte	#100	OK	Benjamin
testminexplosedisparitionalerteenlevement	validé si une ou des mines ne disparaissent pas sans avoir été enclenchées	#101	OK	Benjamin
testirebonnedirection	validé si le tir est dirigé dans la direction indiquée	#102	OK	Benjamin
testirebonneporté	validé si le tire de dépasse pas sa portée	#103	OK	Benjamin
testirepertenergetireure	validé si le tireur perd le bon montant d'énergie quand il tire.	#104	OK	Benjamin
testirepertenergiecible	validé si la cible qui reçoit un tir perd le bon montant d'énergie	#105	OK	Benjamin
testtirepertenergiepourrien	validé si, après un tir, les robots non impliqués ne perdent pas d'énergie	#106	OK	Benjamin
testireobstacle	validé si un tir ne peut pas toucher un obstacle(décor....)	#107	OK	Benjamin
testireobstaclestop	validé si un tir qui rencontre un obstacle ne peut pas être lancé	#108	OK	Benjamin
testiretroploin	validé si le tir ne continue pas après avoir dépassé sa portée	#109	OK	Benjamin
testirhorsdeporté	validé si le tir n'est pas possible si la cible n'est pas dans la portée du tireur	#110	OK	Benjamin
testirecible	validé si le tir s'effectue sur une cible identifiée(que le joueur peut voir, n'étant donc pas dans sa base)	#111	OK	Benjamin
testcibleTK	validé si la cible fait bien partie de l'équipe du tireur	#112	OK	Benjamin
testcibleEnemie	validé si la cible fait partie de l'équipe adverse	#113	OK	Benjamin
testcibleDecorIndes	validé si la cible est bien un obstacle indestructible	#114	OK	Benjamin
testireSupressionObstacleIndes	échoue si un tire peut supprimer un obstacle indestructible	#115	OK	Benjamin
testireSupressionObstacleDestruc	validé si un peut supprimé un obstacle destructible	#116	OK	Benjamin
testireSupressionRobot	validé si un tire supprime le robot ciblé si il n'a plus d'énergie	#117	OK	Benjamin
testireSupressionAbusif	échoue si un objet(robot ,obstacle) est supprimé alors qu'il na pas reçu de tire	#118	OK	Benjamin
testireSupressionDeMasse	échoue si un tire peut supprimer plusieurs robots	#119	OK	Benjamin
Déplacement :				
testdeplacementchar	validé si le char se déplace	#120	OK	Benjamin
testDeplacementCharBonneDir	validé si le char se déplace dans la direction (celle que l'on donne en paramètre)	#121	OK	Benjamin
testDeplacementCharMauvaiseDir	échoue si le char se déplace dans un direction autre que celle indiquer	#122	OK	Benjamin
testDeplacementCharinvalide	échoue si le char se déplace en diagonal	#123	OK	Benjamin
testDeplacementcharHorizontal	validé si le char se déplace de façon horizontal	#124	OK	Benjamin
testDeplacementcharVertical	validé si le char se déplace de façon vertical	#125	OK	Benjamin
testDeplacementcharLignedroite	validé si le char se déplace en ligne droite	#126	OK	Benjamin
testdeplacementcharZigzag	échoue si le char ne se déplace pas en ligne droite	#127	OK	Benjamin
testdéplacementcharmax	validé si le char se déplace au maximum de sa zone de déplacement autorisé	#128	OK	Benjamin
testDeplacementcharTropLoin	validé si le char se déplace plus loin que sa zone de déplacement autorisé	#129	OK	Benjamin
testDeplacementcharRencontreOb	validé si le char rencontre un obstacle lorsqu'il se déplace	#130	OK	Benjamin
testDeplacementcharRencontreObStop	validé si le char qui rencontre un obstacle autre que les mines arrête sont déplacement	#131	OK	Benjamin

testDeplacementCharMassif	échoue si un char ou plusieurs char non sélectionner se déplace quand le char sélectionner se déplace	#132	OK	Benjamin
testDeplacementcharsurMine	validé si un char se déplace sur une case ou se trouve une mine quelle soi allier ou ennemie	#133	OK	Benjamin
testDeplacementcharRencontreObMineContinu	validé si un char rencontre un obstacle de type mines et continu sa route sil peut encore se déplacer	#134	OK	Benjamin
testdeplacementCharRencontreLimite	validé si le char rencontre la limite du terrain quand il se déplace	#135	OK	Benjamin
testdeplacementcharRencontreLimiteStop	validé si le char rencontre la limite du terrain arrête son déplacement	#136	OK	Benjamin
TestdeplacementcharTombedansLeVide	échoue si le char rencontre la limite du terrain continu sont chemin	#137	OK	Benjamin
testDeplacementcharColision	validé si lors d'un déplacement le char rencontre un robot ou un char allié ou adverse	#138	OK	Benjamin
testDeplacementcharColisionStop	validé si lors d'un déplacement le char rencontre un robot ou un char allié ou adverse s'arrête	#139	OK	Benjamin
testDeplacmentcharneseplacepas	validé si le char ne se déplace pas( il reste sur sa position)	#140	OK	Benjamin
testDeplacementTireur	validé si le tireur se déplace	#141	OK	Benjamin
testDeplacementTireurBonneDir	validé si le tireur se déplace dans la bonne direction(celle que l'on ordonne)	#142	OK	Benjamin
testDeplacementTireurMouvaiseDir	échoue si le tireur se déplace dans une direction que l'on a pas demander	#143	OK	Benjamin
testDeplacementYTireurHorizontal	validé si le tireur se déplace en horizontal	#144	OK	Benjamin
testDeplacementTireurVertical	validé si le tireur se déplace en Vertical	#145	OK	Benjamin
testDeplacementTireurDiagonal	validé si le tireur se déplace en diagonal	#146	OK	Benjamin
testDeplacementTireurMax	validé si le tireur se déplace dans sa zone maximal de déplacement	#147	OK	Benjamin
testDeplacementTireurMassif	échoue si un ou plusieurs tireur se déplace alors que le tireur sélectionner se déplace	#148	OK	Benjamin
testDeplacementTireurBougepas	validé si le tireur ne se déplace pas	#149	OK	Benjamin
testDeplacementTireurTropLoin	échoue si le tireur se déplace plus loin que sa zone de déplacement autorisé	#150	OK	Benjamin
testDeplacementTireurRencontreOb	validé si le tireur rencontre un obstacle dans sa zone de déplacement	#151	OK	Benjamin
testDeplacementTireurRencontreObYSNP	validé si le tireur ne peux pas se déplacé a un endroit où il y a un obstacle sauf mine	#152	OK	Benjamin
testDeplacementTireurRencontreObFantome	échoue si le tireur se déplace a un endroit où il y a un obstacle autre que une mine	#153	OK	Benjamin
testDeplacementTireurRencontreMine	validé si le tireur se déplace a un endroit où il y a une mine	#154	OK	Benjamin
testDeplacementTireurRencontreLimite	validé si le tireur la limite de la carte se trouve dans la zone de déplacement du tireur	#155	OK	Benjamin
testDeplacementTireurRencontreLimiteYSNP	validé si le tireur ne peut pas se déplacé en dehors de la limite	#156	OK	Benjamin
testDeplacementTireurRencontreLimiteEtTombe	échoue si le tireur peux se déplacé en dehors de la limite de la carte	#157	OK	Benjamin
testDeplacementTireurColision	validé si le tireur a un robot dans sa zone de déplacement et que c'est indiqué	#158	OK	Benjamin
testDeplacementTireurColisionPassePas	validé si le tireur ne peux se déplacé a un endroit où il y a un robot adverse dans sa zone de déplacement	#159	OK	Benjamin
testDeplacementPiegeur	validé si le piéteur se déplace	#160	OK	Benjamin
testDeplacementPiegeurBonneDir	validé si le piéteur se déplace dans la bonne direction(celle que l'on ordonne)	#161	OK	Benjamin
testDeplacementPiegeurMouvaiseDir	échoue si le piéteur se déplace dans une direction que l'on a pas demander	#162	OK	Benjamin
testDeplacementPiegeurHorizontal	validé si le piéteur se déplace en horizontal	#163	OK	Benjamin
testDeplacementPiegeurVertical	validé si le piéteur se déplace en Vertical	#164	OK	Benjamin
testDeplacementPiegeurDiagonal	validé si le piéteur se déplace en diagonal	#165	OK	Benjamin
testDeplacementPiegeurMax	validé si le piéteur se déplace dans sa zone maximal de déplacement	#166	OK	Benjamin
testDeplacementTireurMassif	échoue si un ou plusieurs piéteurs se déplace alors que le piéteur sélectionner se déplace	#167	OK	Benjamin
testDeplacementTireurBougepas	validé si le piéteur ne se déplace pas sans ordre de déplacement	#168	OK	Benjamin
testDeplacementTireurTropLoin	échoue si le piéteur se déplace plus loin que sa zone de déplacement autorisé	#169	OK	Benjamin
testDeplacementPiegeurRencontreOb	validé si le piéteur rencontre un obstacle dans sa zone de déplacement	#170	OK	Benjamin
testDeplacementPiegeurRencontreObYSNP	validé si le piéteur ne peux pas se déplacé a un endroit où il y a un obstacle sauf mine	#171	OK	Benjamin
testDeplacementPiegeurRencontreObFantome	échoue si le piéteur se déplace a un endroit où il y a un obstacle autre que une mine	#172	OK	Benjamin
testDeplacementPiegeurRencontreMine	validé si le piéteur se déplace a un endroit où il y a une mine	#173	OK	Benjamin
testDeplacementPiegeurRencontreLimite	validé si le piéteur la limite de la carte se trouve dans la zone de déplacement du tireur	#174	OK	Benjamin
testDeplacementPeigeurRencontreLimiteYSNP	validé si le piéteur ne peut pas se déplacé en dehors de la limite	#175	OK	Benjamin
testDeplacementPiegeurRencontreLimiteEtTombe	échoue si le piéteur peux se déplacé en dehors de la limite de la carte	#176	OK	Benjamin
testDeplacementPiegeurColision	validé si le piéteur a un robot dans sa zone de déplacement	#177	OK	Benjamin
testDeplacementPiegeurColisionPassePas	validé si le piéteur ne peux se déplacé a un endroit où il y a un robot adverse dans sa zone de déplacement	#178	OK	Benjamin

Graphique				
testAffichageChar	validé si un char s'affiche bien	#179	OK	Benjamin
testAffichagePiéreur	validé si un piéreur s'affiche bien	#180	OK	Benjamin
testAffichageTireur	validé si un tireur s'affiche bien	#181	OK	Benjamin
testAffichageMine	validé si une mine s'affiche bien	#182	OK	Benjamin
testAffichageBase	validé si une base s'affiche Bien	#183	OK	Benjamin
testAffichageObstacle	validé si un obstacle s'affiche bien	#184	OK	Benjamin
testAffichagePlateau	validé si le plateau s'affiche Bien	#185	OK	Benjamin
testImageChar	Valide si l'image du char correspond bien a un char	#186	OK	Benjamin
testImagePiéreur	Valide si l'image du piéreur correspond bien a un piéreur	#187	OK	Benjamin
testImageTirreur	Valide si l'image du tireur correspond bien a un tireur	#188	OK	Benjamin
testImageMine	Valide si l'image d'une mine correspond bien a une mine	#189	OK	Benjamin
testImageBase	Valide si l'image d'une base correspond bien a une base	#190	OK	Benjamin
testImageObstacle	Valide si l'image d'un obstacle correspond bien a un obstacle	#191	OK	Benjamin