Des preuves qui s’animent : de la prouesse à la méthode

Frédéric Gourdeau
fredg@mat.ulaval.ca

et

Guillaume Paré
guilpare@globetrotter.qc.ca

Université Laval, Québec, Canada

Présentation

Le but de ce cours était de permettre aux personnes possédant déjà une certaine expérience dans l'utilisation de Cabri de construire des figures dans lesquelles des preuves s'animent sur simple déplacement d'un point sur un segment. Ce point, que l'on nommera curseur dans la suite du texte, est celui à partir duquel les autres objets sont construits.

Afin d'y parvenir, et pour être fidèle au titre du cours, nous avions élaboré quelque macros-constructions (que nous nommerons ci après des outils) qui devaient servir à construire méthodiquement et relativement simplement de telles figures Cabri. L'utilisation de ces outils pendant le cours a confirmé qu'ils étaient faciles à utiliser et assez robustes pour permettre de construire des figures variées.

Dans la suite du texte, on retrouvera :

- le plan du cours et quelques indications sur son déroulement ;
- la majeure partie de son contenu ;
- et les deux documents de travail remis aux participants.

Sont aussi joints :

- la barre de menu comprenant les outils de constructions (avec leurs fichiers d'aide) ainsi que les fichiers macros des quatre outils de construction ;
- les figures Cabri présentées pendant le cours ;
- des figures Cabri réalisées par quelques-unes unes des personnes ayant pris part au cours ;
- et des figures Cabri expliquant la construction d'outils de construction.

Nous espérons que les outils et la démarche de construction seront utiles à plusieurs personne et nous encourageons donc leur utilisation. En contrepartie, nous demandons que les figures produites en utilisant ces outils soient disponibles à tous et à toutes. Nous croyons en effet qu'un partage du travail réalisé par les utilisateurs de Cabri est nécessaire pour permettre à la géométrie dynamique de profiter plus largement à toutes et à tous.

Plan du cours

Suite à une brève présentation des animateurs, des exemples de preuves sont présentés : les deux preuves du Théorème de Pythagore qui ont servi de base au cours et qui ont été réalisées avec les outils de constructions sont Pyth-A.fig et Pyth-B.fig. D'autres exemples, réalisés avant la création des outils de construction[1], sont aussi présentés : il s'agit des figures CercleA.fig et CercleB.fig qui illustrent des preuves de la formule de l'aire d'un cercle, ainsi que de la figure Rotation.fig dans laquelle l'élève est invité à réaliser la construction de la rotation étant donnés un objet et son image par cette rotation.

Par la suite, on explique le déroulement du cours, qui comporte trois séances de 75 minutes.

- Première séance : explication des éléments de base des animations et de l'utilisation des outils de constructions.

[1] Ces figures ont été construites par Guillaume Paré. Ce sont elles qui ont été le germe du travail de collaboration réalisé par la suite par les deux auteurs.

- Deuxième séance : construction d'une preuve animée (Document 1 ou 2).
- Troisième séance: suite du travail et, pour ceux et celles qui en ont le temps, début de construction d'une preuve animée personnelle.

Les éléments de base des animations

Les outils de constructions reposent sur des idées fort simples que nous expliquons maintenant.

Dans un premier temps, à partir d'un point curseur *P*, on désire créer un point *P1* ailleurs sur la feuille Cabri de telle sorte que *P1* existe lorsque *P* existe, et n'existe plus lorsque *P* n'existe plus : il faut donc créer *P1* conditionnellement à *P*. Pour ce faire (fichier Cons*P1*.fig), il suffit de créer un vecteur allant de *P* vers le point d'ancrage où l'on désire que *P1* existe, puis de faire la translation de *P* selon ce vecteur : l'image de *P* par cette translation, que l'on nommera *P1*, existera conditionnellement à *P* mais ne bougera pas lorsque l'on déplace *P*. (Il est suggéré de revoir la construction de ConsP1.fig pour mieux suivre ces explications.) Il suffit alors de sauvegarder la macro ayant pour éléments initiaux *P* et le point d'ancrage, et comme élément final le point *P1*, comme cela a été fait dans le fichier ConsP1.fig.

Cette idée d'un point conditionnel est utile notamment pour avoir un texte dans les animations. En effet, le texte explicatif peut être donné comme nom (en utilisant l'outil Nommer) du point *P1*. Lorsque *P1* n'existe pas, ce texte disparaît. On peut ainsi avoir un texte qui varie selon l'étape de l'animation à laquelle on est rendu. C'est de cette manière que le texte est créé dans les animations dont il est question ici.

Deuxièmement, on veut créer des étapes. En effet, nous avons vu comment créer un point conditionnellement à un autre point, mais comment créer des étapes? Pour ce faire, à partir d'un point curseur *P* situé sur un segment *I*, on veut créer des points qui existent successivement selon la position du point *P*. Illustrons cela par un exemple simple en créant un segment I1 et un point *P1* sur ce segment, tels que *P1* se déplace sur *I1* (ce déplacement sera utile pour la suite) lorsque *P* est sur la première moitié de *I*, et tel que *P1* n'existe plus lorsque *P* est sur la seconde moitié de I. Cette construction, que l'on peut revoir dans la construction de ConsP2.fig, est réalisée comme suit.

On crée un segment *I* et un point *P* sur ce segment. À partir de ce segment, on crée un segment *I1* parallèle à *I*, de longueur égale à la moitié de celle de *I*, et situé sous la première moitié du segment *I* (revoir la construction de ConsP2.fig pour les détails) : le point *P1* est l'intersection de *I1* et de la perpendiculaire à *I* passant par le point *P*. Le point *P1* n'existe pas lorsque *P* est sur la seconde moitié du segment *I*. Il suffit alors de sauvegarder la macro ayant pour éléments initiaux *P* et le segment *I*, et comme élément final le point *P1*.

En y réfléchissant un peu, on voit que l'on peut obtenir (en combinant les deux macros dont nous venons de décrire la construction) un point *P2* en un endroit quelconque (et fixe) de la feuille Cabri qui existe lorsque *P* est sur la première moitié de *I*, et qui n'existe pas lorsque *P* est sur la seconde moitié de *I*.

En répétant ce genre de construction pour la seconde moitié du segment *I*, on peut créer une animation à deux étapes. Remarquons que pour construire des animations à plusieurs étapes, il suffit de subdiviser le segment initial en plusieurs parties, et de faire une construction semblable correspondant à chacune de ces parties.

Les deux idées présentées permettent de comprendre la construction de deux des outils que nous avons élaborés : ***Point conditionnel*** et ***Polyconstruction***. On remarquera que l'outil ***Polyconstruction*** permet de construire des animations ayant de 2 à 10 étapes : une application successive de cet outil permet donc de faire n'importe quel nombre d'étapes, et on pourra au besoin subdiviser une étape en appliquant cette macro au segment correspondant à l'étape désirée. La construction de l'outil ***Polyconstruction*** est expliquée dans le dossier ConsPoly.fig.

Les deux autres outils élaborés (***Point de translation*** et ***Point de rotation***) permettent de construire facilement des rotations et des translations d'un point en utilisant comme paramètre un autre point situé sur un segment. Comme ces deux outils sont d'une nature plus classique (ne comportant pas d'aspect conditionnel), nous ne présentons pas ici leur construction. Cependant, principalement parce qu'il y a une difficulté technique à résoudre dans le cas de la rotation (à cause de l'outil Mesure qui ne donne pas de mesure d'angle orienté), nous présentons la construction de ces deux macros de manière détaillée dans les fichiers ConsPdeT.fig et ConsPdeR.fig. Nous recommandons de consulter ces dossiers au besoin et seulement après s'être familiarisé avec l'utilisation des outils ***Point de translation*** et ***Point de rotation*** lors de la réalisation de l'une des deux preuves du Théorème de Pythagore.

Utilisation des outils de constructions

Afin d'illustrer l'utilisation des outils de constructions, la construction de la figure Exemple1.fig est réalisée avec les participants. Tous les éléments présentés étant repris dans les documents portant sur les preuves du Théorème de Pythagore, nous ne détaillons pas ici cette partie du cours.

Construction d'une preuve animée du Théorème de Pythagore

Après avoir ouvert la barre de menu Outils.men, une nouvelle icône apparaît à la droite des icônes habituelles de Cabri. On trouvera dans le menu défilant quatre outils de constructions : ***Polyconstruction, Point conditionnel, Point de translation*** et ***Point de rotation***. Ces outils ont un fichier d'aide qui explique leur utilisation : celle-ci est d'ailleurs détaillée dans les documents qui suivent.

À l'aide de cette barre d'outils, les participants construisent l'une des deux preuves animées du Théorème de Pythagore en suivant les instructions données dans le Document 1 pour la preuve Pyth-A.fig, et celles données dans le Document 2 pour la preuve Pyth-B.fig. Ces deux documents suivent la liste des fichiers Cabri.

Conclusion

Nous croyons que l'objectif poursuivi par le cours a été atteint par la vaste majorité des participantes et des participants. Évidemment, une fois l'expertise technique acquise, il faut trouver comment bien utiliser les possibilités offertes par de telles constructions dynamiques. Nous sommes confiants que plusieurs enseignantes et enseignants y parviendront, enrichissant ainsi l'expérience et la compréhension mathématique de leurs élèves.

De plus, nous considérons que le travail réalisé par les enseignantes et les enseignants dans la préparation et la recherche de constructions dynamiques qui peuvent aider leurs élèves est enrichissant pour ceux et celles qui l'entreprennent. Il s'agit là d'une activité mathématique personnelle et créative qui permet d'aller au-delà du contenu à enseigner et qui permet aux enseignantes et aux enseignants de maintenir et même d'améliorer leur compréhension et leur expertise mathématiques.

Liste des fichiers Cabri joints

Exemples d'animation :
1- Pyth-A.fig
2- Pyth-B.fig
3- CercleA.fig
4- CercleB.fig
5- Rotation.fig
6- Exemple1.fig

Constructions de participants
1- AirePReg.fig : figure construite par Monique Morel[2]
2- Aire1.fig : figure construite par Alicia Noemí Fayó[3]
3- Aire2.fig : idem
4- Aire3.fig : idem
5- Aire4.fig : idem

Construction des macros
1- ConsP1.fig
2- ConsP2.fig
3- ConsPdeT.fig
4- ConsPdeR.fig
5- ConsPoly.fig

---

[2] Monique Morel, mmorel@cstrois-lacs.qc.ca , enseignante de mathématiques, École secondaire Vaudreuil, Commission scolaire des Trois-Lacs, Québec, Canada
[3] Alicia Noemí Fayó, aliciafayo@ciudad.com.ar, professeure de mathématique et d'informatique, Grupo de.Investigación Matemática XVIII, R. Argentina.

Macros
1- Point_de_rotation.mac
2- Point_de_translation.mac
3- Point_conditionnel.mac
4- Polyconstruction.mac

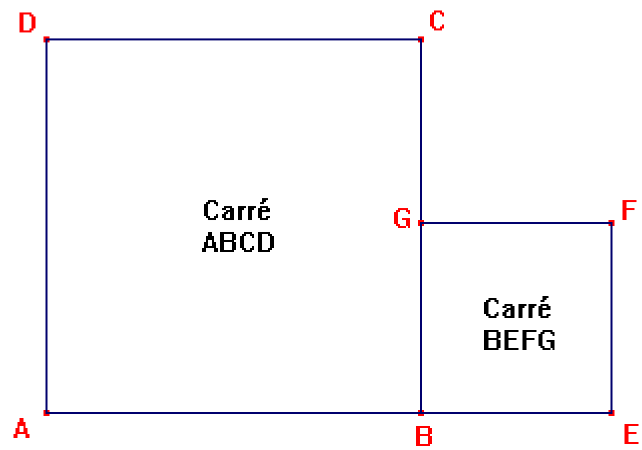
Barre d'outils
1- Outils.men

Document 1 - Première construction d'une preuve du Théorème de Pythagore

Suivez les étapes pour construire une animation de la preuve du Théorème de Pythagore semblable à celle présentée dans le fichier Pyth-A.fig

Première partie :
<u>Les carrés de départ</u>

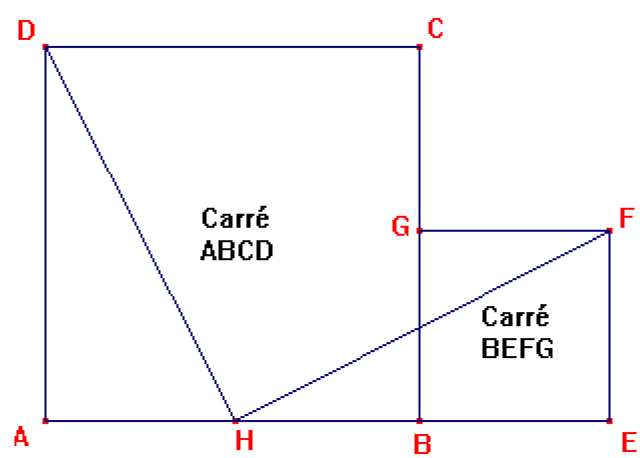
On veut tracer deux carrés de grandeurs différentes ayant un côté commun. Pour ce faire, il y a plusieurs façons de procéder. En voici une :



1. Commencer par tracer une droite dans le bas de votre feuille Cabri. Elle servira de base aux deux carrés.
2. Placer deux points sur cette droite. Ils serviront de base au carré *ABCD*. Utiliser des perpendiculaires et des cercles pour compléter le carré.
3. Procéder de la même façon pour construire le carré *BEFG*, en prenant soin d'utiliser le point *B* du premier carré comme sommet.
4. Enlever les objets inutiles, tout en gardant les points aux sommets des carrés.

Deuxième partie :
<u>Les triangles de départ</u>

Dans cette preuve du Théorème de Pythagore, on veut prouver que la somme des aires des deux carrés est égale à celle du carré construit sur l'hypoténuse du triangle rectangle (dont les côtés adjacents à l'angle droit sont congruents à *AB* et *BE*). On doit tracer deux triangles congruents dans cette figure pour faire apparaître le carré de l'hypoténuse. Vous pouvez les construire de la façon suivante :



5. À l'aide de l'outil compas, prendre la mesure d'un côté du carré *BEFG* et la reporter au point *A*. Placer un

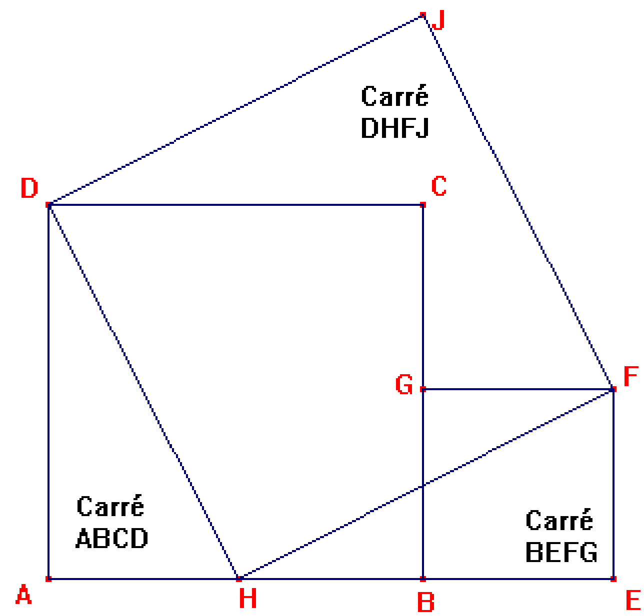
point à l'intersection de la base *AB* et du cercle nouvellement créé et le nommer *H*.

6. Tracer maintenant les segments *DH* et *HF*.

Troisième partie :
<u>Le carré de l'hypoténuse</u>

Nous voulons maintenant compléter le carré de côté *DH* (ou *HF*).

7. Tracer la perpendiculaire à *DH* passant par *D*.
8. Tracer la perpendiculaire à *HF* passant par *F*.
9. Nommer le point d'intersection de ces deux perpendiculaires *J*.
10. Vous pouvez maintenant construire le carré *DHFJ*.

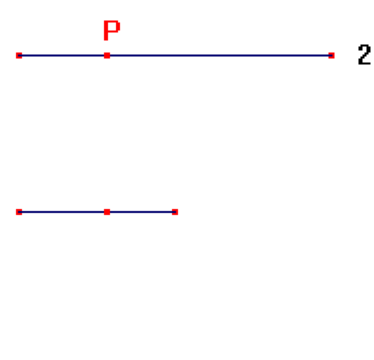


Nous avons maintenant terminé la construction de la grande majorité des objets qui nous serviront pour l'animation.

Cette preuve du Théorème de Pythagore fonctionne comme suit : il faut remplir le carré *DHFJ* à l'aide des carrés *ABCD* et *BEFG*. Pour ce faire, on remarque qu'il reste à placer les triangles *AHD* et *HEF* dans le carré *DHFJ*.

Quatrième partie :
<u>Le début de l'animation</u>

On doit maintenant construire les étapes de l'animation. Nous prévoyons deux étapes.



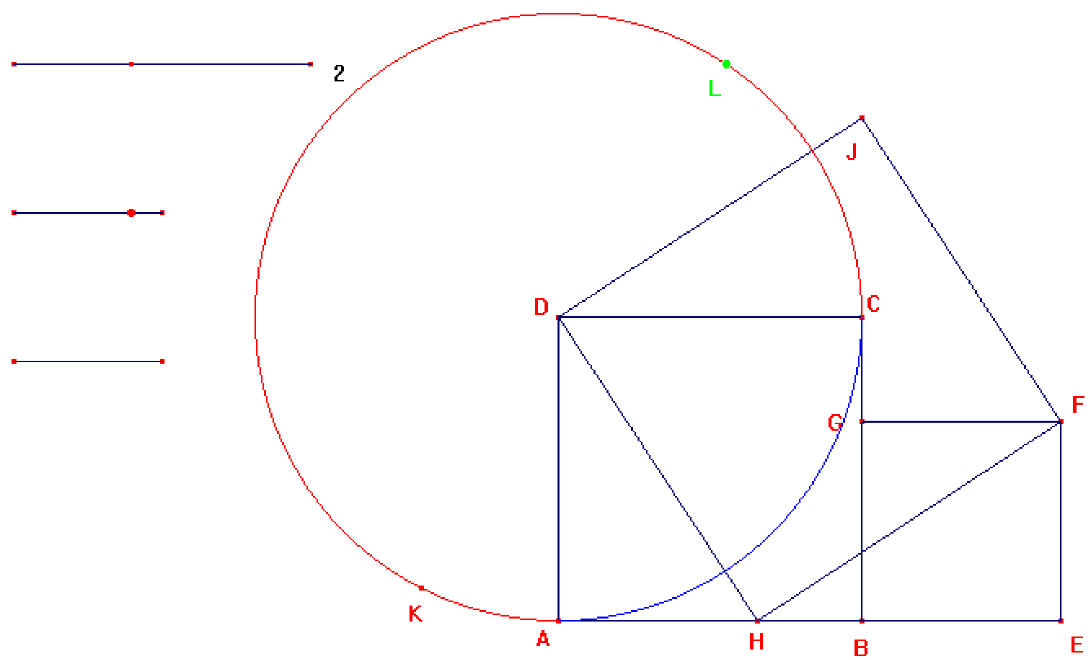
11. Inscrire le nombre 2 sur la feuille Cabri.
12. Tracer le segment qui servira à l'animation et y placer un point *P*.
13. On construit les étapes avec la macro ***Polyconstruction*** (éléments initiaux : *P*, son segment et le nombre 2)

On obtient alors deux segments sur lesquels existera successivement un point lorsque le point *P* se déplacera. Ces points seront *P1* et *P2*. Il n'est pas nécessaire de les nommer, mais ces noms allégeront les présentes explications.

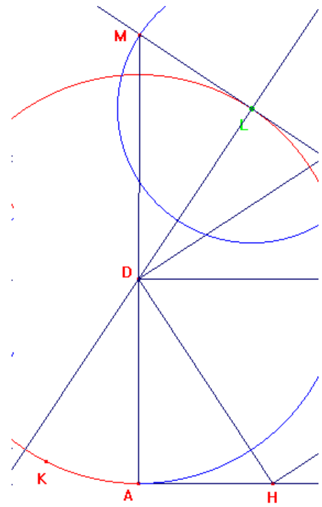
Cinquième partie :
<u>La rotation du triangle *AHD*</u>

Pour compléter le carré *DHFJ*, nous allons commencer par faire une rotation du triangle *AHD* autour du sommet *D*. Noter que la rotation choisie sera d'orientation négative (puisque de sens horaire) et ce afin d'éviter que le triangle ne masque la figure lors de sa rotation. Comme il s'agit là de notre première étape, assurez-vous que le point *P* est placé de manière à ce que le point *P1* soit visible sur le premier segment.

14. La macro ***Point de rotation*** fonctionne avec un arc de cercle. Nous allons donc le tracer. Tracer un cercle de centre *D* et de rayon *DA*.
15. Placer, sur ce cercle le point *K* qui servira à définir notre arc de cercle. Tracer par la suite l'arc de cercle *AKC*.
16. Utiliser la macro ***Point de rotation*** pour construire la rotation du point *A* (éléments initiaux : *P1*, son segment et l'arc *AKC*). Nommer le nouveau point *L*.



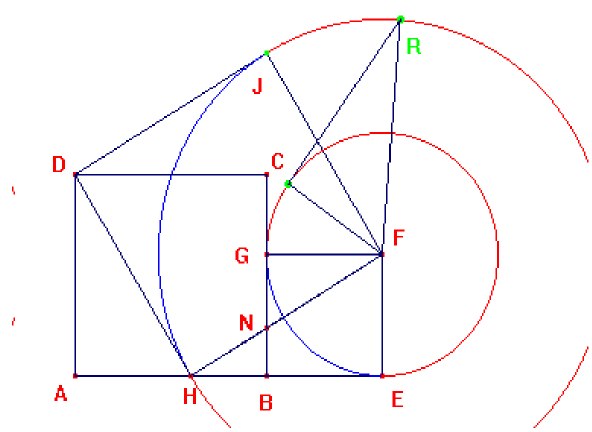
17. Tracer la droite *DL*.
18. Tracer la perpendiculaire à cette droite passant par *L*.
19. Avec le compas, reporter au point *L* la distance *AH*.
20. À l'intersection du cercle tracé par le compas et la perpendiculaire à *DL*, placer le point *M*.
21. Tracer le triangle *DLM*.



Sixième partie :
<u>La rotation du triangle *FHE*</u>

Pour la seconde étape de notre animation, nous allons déplacer le triangle *FHE* dans le haut du carré *DHFJ*. Avant de commencer, déplacer le point *P* pour que le point *P2* soit bien visible sur le second segment.

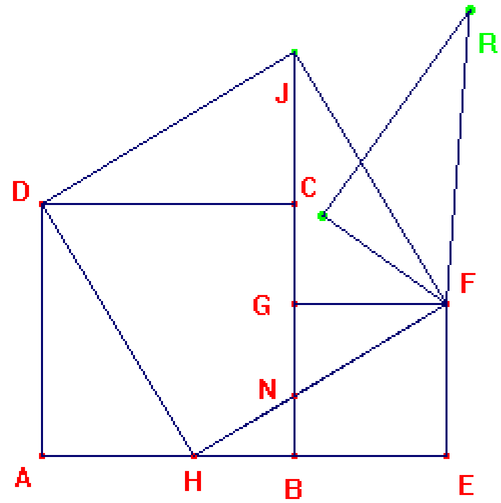
22. Nommer *N* le point d'intersection entre le segment *GB* et le segment *FH*.
23. Tracer le cercle de centre *F* et de rayon *FH*. Tracer par la suite l'arc *HJ* qui servira à la rotation (arc d'orientation positive de *H* vers *J*).
24. Utiliser la macro ***Point de rotation*** pour construire la rotation du point *H* (éléments initiaux : *P2*, son segment et l'arc *HJ*). Nommer le nouveau point *R*.
25. Construire un triangle congruent au triangle *HEF* à partir du point *R,* ce qui peut être fait en ré-utilisant la macro ***Point de rotation*** avec le point *E* ou autrement. Soit *K* le troisième sommet de ce triangle, les deux autres étant *R* et *F*.



Septième partie :
<u>Le triangle *AHD*</u>

Il serait bon que le triangle *DCJ* (celui obtenu après la rotation) reste visible dans le carré *DHFJ* pendant cette seconde étape.

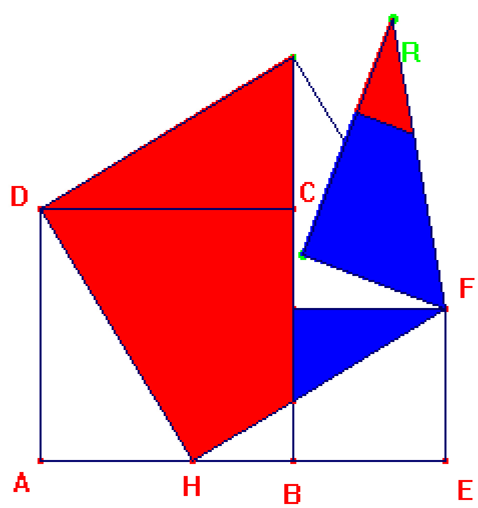
26. Utiliser la macro ***Point conditionnel*** (éléments initiaux : *P2* et *J*) pour placer un point sur le point *J* qui n'existera que lors de cette étape.
27. Construire le triangle *DCJ* en utilisant le nouveau point qui s'est superposé à *J*.



Huitième partie :
<u>Les couleurs</u>

On peut maintenant remplir chacune des figures pour qu'elles soient plus visibles. Pour bien identifier chacun des carrés de départ, nous allons remplir les polygones qui composent le carré *ABCD* en rouge et ceux du carré *BEFG* en bleu.

28. À l'aide de la macro ***Point conditionnel***, faire, à l'étape 1 et dans son emplacement d'origine, le triangle *NHB* en rouge (éléments initiaux : *P1* et *B*, par exemple). Faire également le polygone *FNBE* en bleu de la même façon.
29. Colorer le triangle *DLM* de la première étape en rouge.
30. Dans la seconde étape, avec l'outil compas, reporter la mesure *HB* au sommet *R*. Tracer par la suite une perpendiculaire à *KR* (voir l'étape 25 pour le point *K*) au point d'intersection du cercle et de *KR*. Vous avez maintenant les deux points qui sont les images des points *B* et *N* pendant la rotation (nommés *B'* et *N'*). Créer et colorer, le triangle *RB'N'* en rouge et le polygone *FN'B'E'* en bleu.
31. Colorer le triangle *DJC* de la septième partie des explications en rouge.
32. Finalement, créer et colorer le polygone *DHNC* en rouge et le triangle *NFG* en bleu, sans toutefois utiliser de macro (nous voulons que ces polygones restent toujours visibles).



Neuvième partie :
<u>Le texte explicatif</u>

Pour placer un texte qui expliquera ce théorème étape par étape :

33. Placer un point où vous voulez que le texte commence (point d'ancrage).
34. Pour chacune des étapes, utiliser la macro ***Point conditionnel*** (éléments initiaux : point curseur de l'étape et point d'ancrage) pour qu'un nouveau point par étape apparaisse.
35. Cacher le point d'ancrage et donner comme nom à chaque point conditionnel l'explication de l'étape à laquelle il est associé.

Dixième partie :
<u>La présentation</u>

Il ne vous reste plus qu'à cacher tout ce qui est maintenant inutile. (Par exemple, les deux étapes, leur point et les points de chaque côté du segment du point P.) Ajouter un titre à la figure et enregistrer. Si votre figure doit être utilisée par des élèves, nous vous conseillons de cacher tous vos points.

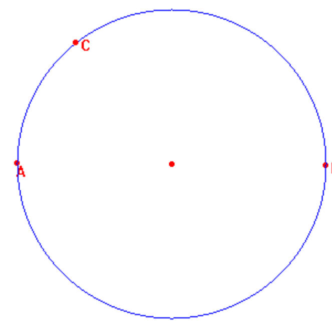
La figure créée ne fonctionne que si le carré $ABCD$ est plus grand que le carré $BEFG$, ce que vous pouvez vérifier en déplaçant le point B vers la gauche. Pour remédier à la situation, on peut déplacer B pour que le carré $ABCD$ soit plus petit que le carré $BEFG$ et reprendre la construction.

Document 2 – Deuxième construction d'une preuve du Théorème de Pythagore

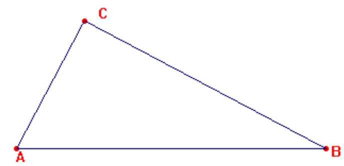
Suivez les étapes pour construire une animation de la preuve du Théorème de Pythagore semblable à celle présentée dans le fichier Pyth-B.fig

1. Première partie : les éléments de départ

On crée les éléments de la construction qui doivent toujours être présents. Dans le cas qui nous occupe, il s'agit d'un triangle rectangle *ABC*, d'hypoténuse *AB*. Pour construire un tel triangle, il y a plusieurs possibilités. On peut construire deux points *A* et *B*, puis un cercle centré au point milieu de *AB* et passant par *A* (et *B*). Il suffit alors de choisir un point quelconque *C* sur le cercle.

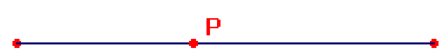


Le triangle *ABC* peut alors être construit et on cache par la suite le cercle et son centre.



2. Début de l'animation.

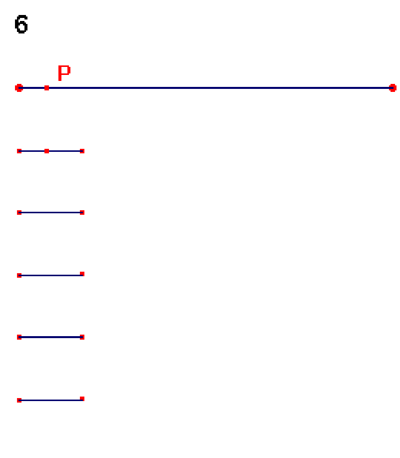
On crée un segment et un point *P* sur le segment : *P* est le point qui contrôlera l'animation.



3. Nombre d'étapes de la construction.

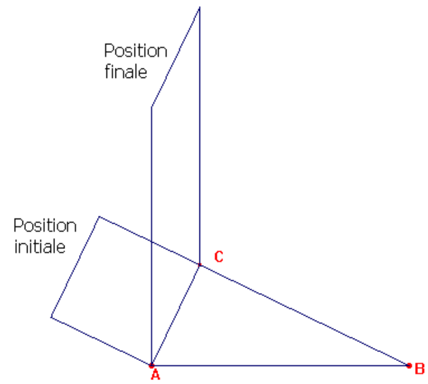
On doit maintenant commencer les étapes de la construction. Nous prévoyons 6 étapes à l'animation. On doit entrer le nombre 6 à l'écran puis utiliser la macro ***Polyconstruction*** appliquée au segment de départ, au point *P* et au nombre 6. On obtient alors six segments sur lesquels existera en succession un point : cela nous permet de créer une animation à 6 étapes.

On positionne le point *P* de telle sorte qu'il y ait un point mobile *P1* sur le premier segment. Il n'est pas nécessaire de nommer le point *P1* : nous ne le faisons ici que pour clarifier les explications.



4. **Première étape de l'animation** : un carré construit sur le côté AC du triangle doit se transformer en un parallélogramme et glisser… pour atteindre la position finale dans la figure.

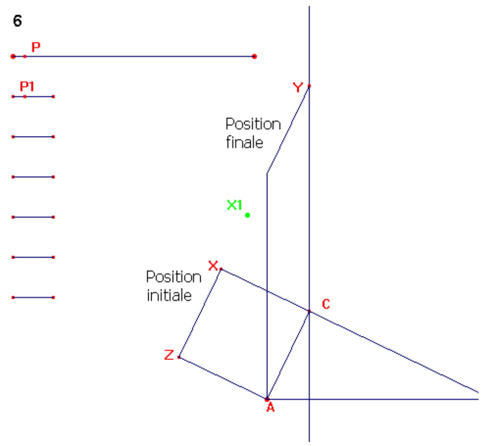
   Pour ce faire, on doit en premier lieu construire un carré de côté AC et la perpendiculaire à AB par le point C.



5. Considérons les points X et Y sur la figure. Le point Y est le point d'intersection de la perpendiculaire à AB par le point C et de la droite ZX.

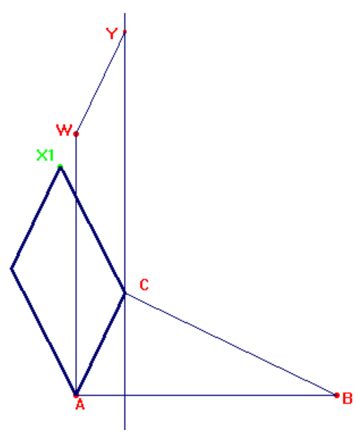
   Un point $X1$, sommet du parallélogramme, devra se déplacer de X à Y : on crée donc un point $X1$ avec la macro ***Point de translation*** (éléments initiaux $P1$, segment auquel appartient $P1$, X et Y).

   Explication : le déplacement de P entraîne celui de $P1$, qui entraîne celui de $X1$. Lorsque $P1$ n'existe pas (par exemple, à la deuxième étape de l'animation), $X1$ n'existe pas.



6. On crée un parallélogramme dont trois des sommets sont A, C et $X1$ (ce que l'on peut faire en deux coups de compas, ou autrement). Remarquons que lorsque $P1$ n'existe pas, le parallélogramme n'existe pas.

   On peut cacher les éléments qui ne nous sont plus utiles; il est souvent préférable d'attendre avant de cacher des éléments car certains peuvent être utiles dans la suite de la construction. Ici par exemple, le point W (sur la figure ci-contre) est utile : il s'agit du point d'intersection de la droite ZX avec la perpendiculaire à AB au point A.

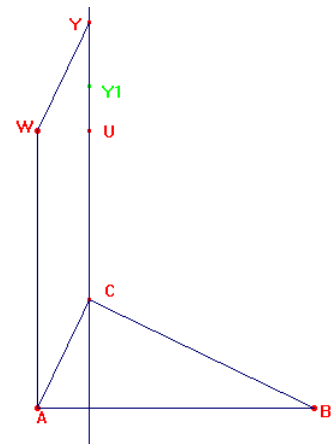


7. **Deuxième étape de l'animation.**

   On doit faire glisser deux sommets du parallélogramme (*Y* et *C*) pour obtenir un rectangle dont la base est sur *AB*.

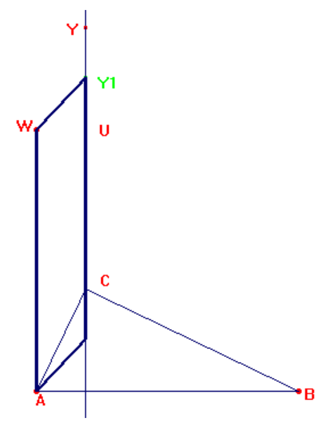
   On déplace *P* pour qu'un point *P2* apparaisse sur le deuxième segment. On crée le point *U* (*UW* est parallèle à *AB*) et on crée le point *Y1* qui se déplace de *Y* vers *U* avec la macro ***Point de translation*** (éléments initiaux *P2*, segment auquel appartient *P2*, *Y* et *U*).

   Le déplacement de *P* entraîne celui de *P2*, qui entraîne celui de *Y1*. Lorsque *P2* n'existe pas, *Y1* n'existe pas.



8. On crée le parallélogramme dont trois des sommets sont *A*, *W* et *Y1*.

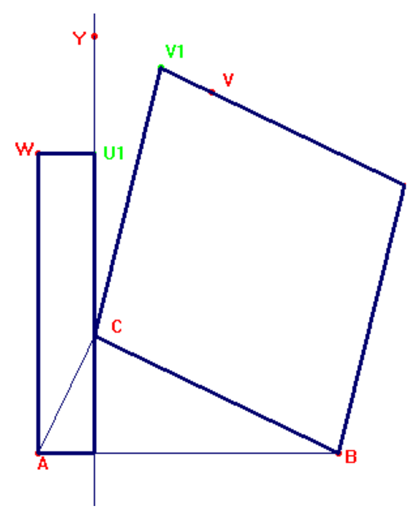
   La deuxième étape est terminée : on cache ce qui ne sera plus utile.



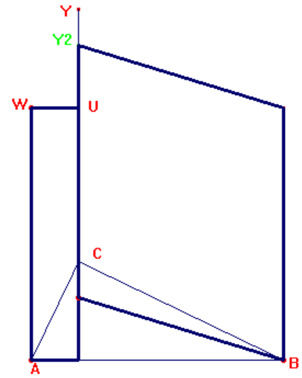
9. **Troisième étape de l'animation.**

   On déplace le point *P* pour qu'un point *P3* apparaisse sur le troisième segment. On applique la même démarche de construction que lors de la première étape pour un carré construit sur le côté *BC*, obtenant un parallélogramme dont le sommet *V1* se déplace de *V* à *Y* (où *V* est l'un des sommets du carré construits sur *BC*).

   Il est intéressant que le rectangle dont trois des sommets sont *W*, *U* et *A* demeure présent lors de cette étape. On doit donc le construire conditionnellement à l'existence de *P3*. Pour ce faire, appliquer la macro ***Point conditionnel*** à *P3* et *U* pour qu'un nouveau point *U1* existe et soit superposé à *U*. On construit alors le rectangle désiré en prenant bien soin d'utiliser *U1* comme l'un de ses sommets (voir figure).

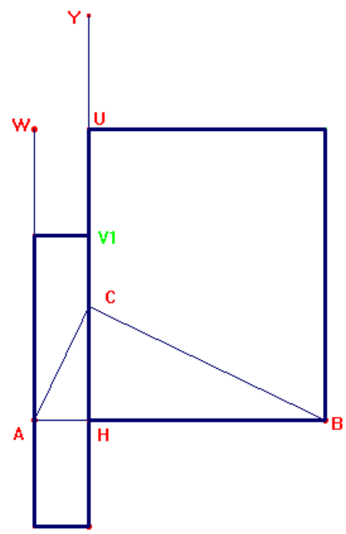


10. Quatrième étape. On déplace le point *P* pour qu'un point *P4* apparaisse sur le quatrième segment. On applique la même démarche de construction que lors de la deuxième étape, pour le parallélogramme approprié. Sur la figure, le point *Y*2 se déplace de *Y* vers *U*.



Il est intéressant que le rectangle dont trois des sommets sont W, *U* et *A* soit présent lors de cette étape également. On doit donc le construire conditionnellement à l'existence de *P4*.

11. Cinquième étape. On déplace le point *P* pour qu'un point *P5* apparaisse sur le cinquième segment.

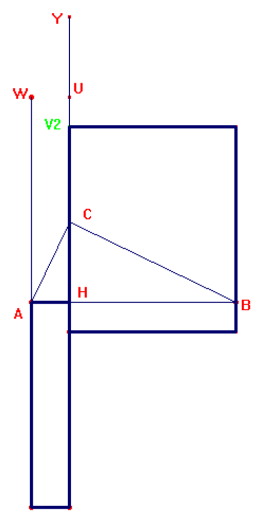


Considérons les points *U* et *H* sur la figure. Le point V1 devra se déplacer de *U* jusqu'en *H* : on crée donc le point *V1* avec la macro ***Point de translation*** (éléments initiaux P5, son segment, *U* et *H*).

On construit le rectangle congruent au rectangle *UHAW* et dont l'un des sommets est *V1* (voir figure). Il y a plusieurs manières de réaliser cette construction.

Il est intéressant que le rectangle dont trois des sommets sont *U*, *H* et *B* soit présent lors de cette étape également. On doit donc le construire conditionnellement à l'existence de *P5*.

12. Sixième étape. On déplace le point *P* pour qu'un point *P6* apparaisse sur le sixième segment.



On répète la construction faite à l'étape précédente, en faisant déplacer le rectangle de droite conditionnellement à *P6*. Le point *V2* se déplace de *U* vers *H*.

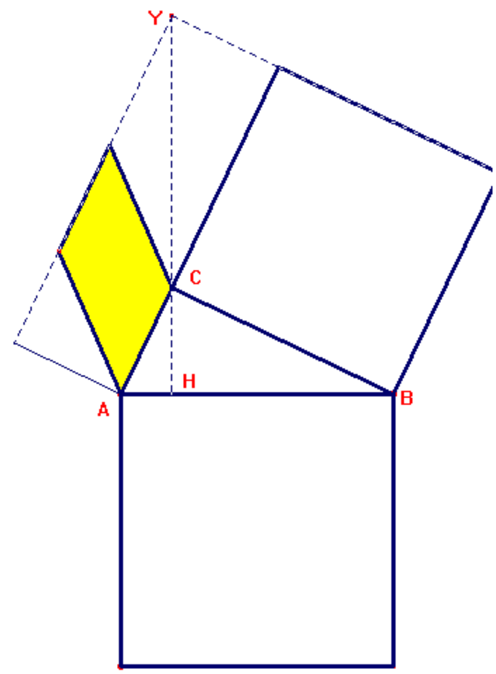
Il est intéressant que le rectangle obtenu à la fin de l'étape précédente soit présent lors de cette étape également. On doit donc le construire conditionnellement à l'existence de *P6*.

13. Les couleurs et les éléments fixes. On peut maintenant ajouter les éléments que l'on souhaite avoir en tout temps : ici, les carrés construits sur les trois côtés du triangle initial.

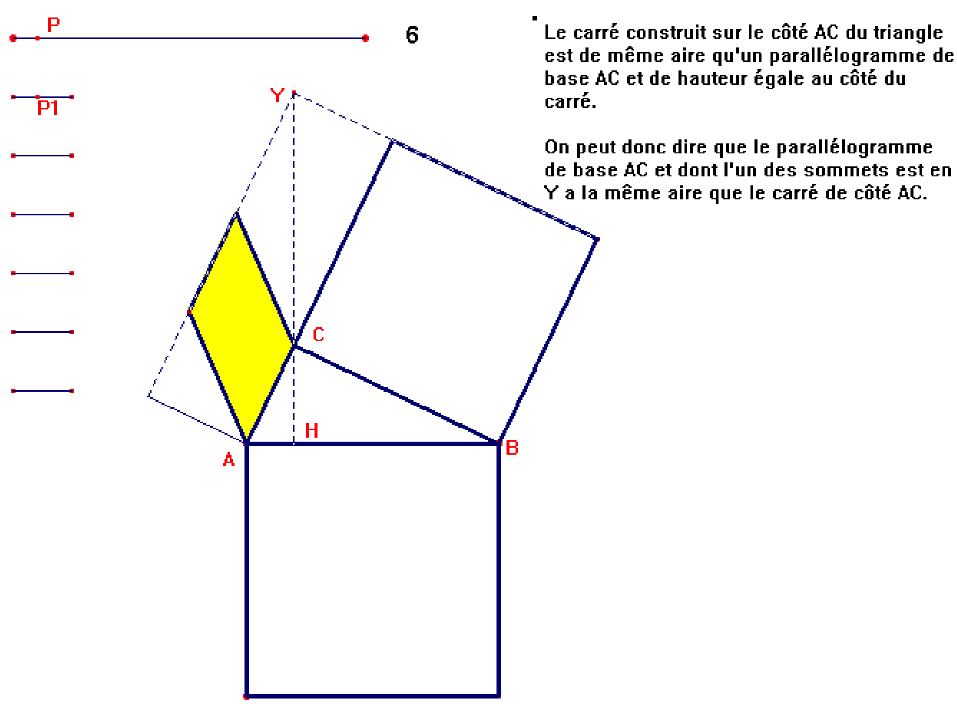
On ajoute aussi les couleurs de remplissage. Il est important de le faire à chaque étape en prenant soin de déplacer le point *P* pour couvrir toutes les étapes.

Note

Il n'est pas toujours une bonne idée d'utiliser des couleurs de remplissage puisque celles-ci peuvent masquer des objets.



14. Le texte explicatif. On peut composer un texte différent pour chacune des six étapes de la construction. Il suffit de placer un point d'ancrage *Q*, lieu où apparaîtra le texte. On crée par la suite un point conditionnel superposé à *Q* pour chacune des étapes (pour l'étape 1, on applique la macro ***Point conditionnel*** à *P1* et *Q*) et on nomme de nos explications ce point conditionnel. On complète le travail en cachant le point d'ancrage.



15. La présentation et la robustesse. Il ne reste plus qu'à cacher tout ce qui est maintenant inutile dont les 6 segments ayant servi à la construction. Si le fichier doit être manipulé par des élèves, il est utile de le rendre aussi robuste que possible. Pour ce faire, on peut punaiser des points; cacher des points; ou encore créer des points superposés à nos points initiaux, cacher les points initiaux et nommer les nouveaux points.