

Modélisation et simulation numérique
Prise en Main de Cast3M
Master IMM 1.

1 Introduction

Ce document est basée sur le manuel PRISE EN MAIN DE CASTEM 2000 PAR L'EXEMPLE de J.S. Fleuret – CEA Saclay / DRN / DMT / SEMT / LAMS – disponible en ligne. Il a pour but une rapide mais succinte prise en main du logiciel Cast3M.

1.1 Présentation de Cast3M

Cast3M ex-Castem2000 est un logiciel de calcul de structures par la méthode des éléments finis et plus généralement de résolution d'équations aux dérivées partielles par la méthode des éléments finis.

Il a été développé au Département de Mécanique et Technologie (DMT) du Commissariat à l'Energie Atomique (CEA). La principale particularité de Castem 2000, est d'être extrêmement adaptable aux multiples applications propres à chaque utilisateur.

Depuis l'été 1999, Castem2000 est gratuit pour l'enseignement et la recherche, de plus le code source est ouvert aux développeurs. Il est devenu cast3M en 2001. Pour plus de renseignement consulter le site, où vous trouverez le logiciel en téléchargement gratuit et de la documentation :

<http://www-cast3m.cea.fr>.

Le code est écrit en *esope*, un dialecte de **fortran** et met à profit la notion de programmation orientée objet dans **fortran**.

1.2 Le langage de Cast3M : Gibiane

Gibiane est le langage interprété qui permet de communiquer avec le programme. Ainsi, le principe est d'écrire un programme en langage GIBIANE à l'aide d'un éditeur de texte (n'importe lequel). Puis de lancer l'application castem sur le fichier créé. Il est recommandé d'utiliser le suffixe ".dgibi".

La syntaxe est basée sur l'utilisation de directives, d'opérateurs et de procédures qui s'appliquent à des opérandes. Dans le premier cas la syntaxe est :

DIRECTIVE OPERANDES.

Par exemple dans

TRACE MAILLAGE ;

TRACE est la directive de traçage et **MAILLAGE** est l'opérande que l'on veut visualiser. Dans un second cas, la syntaxe est :

RESULTATS = OPERATEUR OPERANDES.

Par exemple dans

LIGNE = DROITE P1 P2 ;

LIGNE sera l'objet construit en reliant P1 à P2 par une droite. DROITE est l'opérateur qui s'applique sur les opérandes P1 et P2 et le résultat s'appelle LIGNE. La procédure peut utiliser, suivant sa définition, l'une ou l'autre des syntaxes. Il convient de compléter ce paragraphe en précisant quelques règles syntaxiques de GIBIANE :

- * Le point-virgule termine une instruction.
 - * Gibiane ne fait pas a priori de distinction entre Majuscule ou en minuscule. Ainsi l'instruction "TrAce maillage ;" équivaut à "TRACE MAILLAGE ;". Il est recommandé cependant de ne pas mélanger majuscules et minuscules pour des raisons de lisibilité. Par habitude, tout est écrit en majuscules. c'est ce que nous recommandons.
 - * Une instruction doit être écrite sur moins de 9 lignes.
 - * Une ligne peut contenir plusieurs instructions (séparées par des points-virgules).
 - * Les lignes de commentaire sont précisées par un astérisque dans la première colonne.
 - * Les opérateurs et les directives sont définis par leurs 4 premiers caractères mais on peut en donner plus s'il n'y a pas de confusions possibles (ex : TRAC, TRACE, TRACER). Il est conseillé d'utiliser un mot lisible :
TRACER plutôt que TRAC ; DROITE plutôt que DROI.
- Il est alors plus facile de lire le programme et de corriger les erreurs éventuelles.
- * Une procédure se distingue au niveau utilisateur par la nécessité de l'orthographier correctement avec toutes ses lettres (< 8).
 - * L'instruction est interprétée de gauche à droite, et les opérateurs sont exécutés dès qu'ils sont lus. Ainsi $1+2*3=9$ (!) . Pour retrouver l'ordre de priorité mathématique, il convient d'ajouter des parenthèses : $1+(2*3)=7$.
 - * Seuls les 72 premiers caractères d'une ligne sont pris en compte.
 - * La longueur du nom attribué à un objet ne doit pas dépasser 8 caractères.

Il est conseillé d'éviter d'attribuer à un objet le nom d'un opérateur existant, car ce dernier serait alors écrasé. Pour cela on peut conseiller de ne pas donner des noms de 4 caractères et de mettre un nombre en fin de nom (il n'y a qu'un opérateur ayant un nombre en fin de nom c'est CER3) ou de protéger les opérateurs et directives en utilisant les côtes (ex : 'DROI').

Il n'a été listé que les principales règles syntaxiques de Gibiane, pour en savoir plus sur les possibilités de ce langage, voir les manuels de référence.

1.3 Système d'unités

Castem 2000 ne dispose d'aucun système particulier d'unité de mesure. C'est à l'utilisateur de fournir les données dans un système cohérent vérifiant la loi fondamentale de la dynamique : $F = M.\gamma$.

Ainsi on peut donner deux exemples de systèmes cohérents (pour de l'acier) :

Seule la mesure des angles doit être dans tous les cas exprimée en **degrés** pour la géométrie et en **radians** pour les résultats obtenus.

Longueur	Masse	Force	Temps	Masse Volumique	Module d'Young	Contrainte ou Pression
m	kg	N	Sec	$7.85 \cdot 10^3$	2.10^{11}	Pa
mm	10kg	N	Sec	$7.85 \cdot 10^{-9}$	2.10^5	MPa

TABLE 1 – Système d'unités pour un acier

1.4 Analyse par éléments finis

Tout problème d'éléments finis peut être construit de la manière suivante :

1. Description de la géométrie, maillage. Choix du support géométrique.
2. Choix du type d'éléments finis et du modèle de comportement.
3. Donnée des caractéristiques du matériau et des caractéristiques géométriques supplémentaires.
4. Donnée des conditions aux limites.
5. Donnée du chargement.
6. Résolution du système.
7. Post-traitement des résultats.

Nous allons, dans ces T.P., reprendre ce processus en le détaillant sur des exemples simples.

1.5 Problèmes pouvant être résolus par CASTEM

CASTEM dispose d'un certain nombre d'exemples reprenant la majorité des problèmes pouvant être traités. Ces exemples sont disponibles dans le répertoire *dgibi* du user contenant *castem*. *Castem* étant à l'origine un code pour le calcul des structures, on y trouve de l'élasticité linéaire ou non-linéaire, des modèles de plaques de coques ou de poutres très variés. Mais sa souplesse lui ont permis de s'adapter à d'autres problèmes. Ainsi, on trouve également des problèmes de thermiques et de mécanique des fluides.

1.6 Environnement

Castem2000 existe pour de nombreux environnements de travail. Il existe pour pratiquement tous les UNIX (AIX, SOLARIS, HP-UX, DEC...) et en particulier pour *Linux* sur PC. Il existe également pour MS-Windows 9*/NT/XP, mais il n'y a pas à ma connaissance de version pour Mac OS.

Dans la suite, nous supposerons l'environnement KDE sous Linux familier pour l'utilisateur. Nous supposerons également qu'un éditeur de texte est à disposition (tel que *XEmacs*, *Kwrite*, etc).

Lors de l'installation sous Windows, le suffixe *.dgibi* est inscrit dans la base de registre ce qui permet de lancer l'exécution de *castem* en cliquant sur un fichier *.dgibi*.

Sous Kde, un comportement similaire est possible, il suffit de configurer l'application *konqueror* pour associer les fichiers *.dgibi* et l'exécutable *castem*.

On pourra également lancer castem à partir d'une console. C'est la méthode recommandée. Cast3M possède une documentation en ligne. On peut (et on doit) la consulter pour avoir la syntaxe d'une commande donnée. Cette documentation est au format html, on peut donc la visualiser à l'aide d'un navigateur internetk onqueror, mozilla ou firefox par exemples.

Une copie de cette documentation peut être consultée à la page

<http://www-cast3m.cea.fr/cast3m/listnotices.do>

2 Génération de Maillage

L'objet du maillage est de discrétiser géométriquement le domaine d'analyse de manière à pouvoir ultérieurement associer une formulation éléments finis au support géométrique. Concrètement cette discrétisation s'effectue par la création d'objets de type maillage (points, lignes, surfaces, volumes) à l'aide des opérateurs géométriques. La technique à suivre est presque toujours la même :

1. construction des points
2. construction des lignes à partir des points
3. construction des surfaces à partir des lignes
4. construction des volumes à partir des surfaces

Cependant les objets de type **maillage** constituent le support géométrique des éléments finis qui seront définis ultérieurement. Le type des supports géométrique doit donc être en accord avec les éléments qui seront utilisés : Par exemple, si les éléments de type poutre ou barre ou encore coque axisymétrique, les supports géométriques devront être des **SEG2** (segments à 2 noeuds).

2.1 Maillage Simple

Dans un premier exemple, nous allons mailler un cube de côté 10 m (l'unité en mètre est ici indiquée pour fixer les idées, rappelons qu'il n'y a pas de système d'unité sur Castem).

```
*
* DEBUT DU FICHIER MAILLAGE-CUBE.DGIBI
*
TITR ' PREMIER MAILLAGE' ;
OPTION DIMENSION 3 ELEMENT CUB8 ;
*
* POINTS
*
P1 = 0. 0. 0. ;
P2 = 10. 0. 0. ;
P3 = 10. 10. 0. ;
P4 = 0. 10. 0. ;
*
* LIGNES
*
P1P2 = P1 DROITE 10 P2 ;
P2P3 = P2 DROITE 10 P3 ;
P3P4 = P3 DROITE 10 P4 ;
P4P1 = P4 DROITE 10 P1 ;
*
* SURFACE
*
SURF1 = DALLER P1P2 P2P3 P3P4 P4P1 ;
*
* VOLUME
*
VOL1 = SURF1 VOLUME 10 TRANSLATION (0. 0. 10.) ;
*
* TRACAGE
*
TRACER SURF1 QUALIFICATION ;
TRACER CACHER VOL1 ;
*
*FIN DU FICHIER
*FIN ;
```

2.2 Analyse de la solution

TITR 'PREMIER MAILLAGE' ; Donne un nom à l'étude.

OPTION DIMENSION 3 ELEMENT CUB8 ; On aurait également pu écrire

OPTI DIME 3 ELEM CUB8 ;

La directive OPTI(ON) permet de déclarer les principaux paramètres du programme, par exemple la dimension de l'espace, le type d'éléments géométriques utilisé.... Ici on travaille en trois dimensions d'où le choix de DIME(NSION) 3, et un choix d'éléments géométriques appropriés à notre problème semble être les éléments CUB8 (élément cubique à 8 noeuds). Le choix de la dimension et des éléments géométriques est très lié à celui des éléments finis et donc au type d'étude.

En fait ELEM permet de définir les classes d'éléments autorisés en donnant l'élément le plus complexe. Ainsi CUB8 autorise les éléments POI1, SEG2, TRI3, QUA4, TET4, PYR5 et PRI6. La définition CUB8 en début de fichier évite de préciser à chaque étape (courbes, surfaces, volumes) le nouveau type d'éléments.

P1 = 0. 0. 0. ;

P2 = 10. 0. 0. ;

P3 = 10. 10. 0. ;

P4 = 0. 10. 0. ;

Le point est construit en associant à son nom, ses coordonnées (et éventuellement une densité définie en quatrième coordonnée).

P1P2 = p1 DROITE 10 p2 ;

P2P3 = p2 DROITE 10 p3 ;

P3P4 = p3 DROITE 10 p4 ;

P4P1 = p4 DROITE 10 p1 ;

L'opérateur DROI(TE) permet de construire une ligne droite à partir de ses deux points extrêmes et de la règle permettant de calculer les coordonnées des points internes. Ces lignes créées sont orientées et automatiquement subdivisées en un certain nombre de segments que l'on pourra spécifier. Ici on a spécifié 10 segments entre deux points. Les opérateurs CERC, CER3, PARA, CUBP, CUBT, COURBE, QUELCONQUE et INTERSECTION permettent également de tracer des courbes telles que des arcs de cercle, de paraboles, des lignes brisées... On se reportera à la notice d'utilisation de CASTEM 2000 pour plus de précision.

SURF1 = DALLER P1P2 P2P3 P3P4 P4P1 ;

L'opérateur DALL(ER) permet de construire une surface délimitée par 4 côtés ayant deux à deux le même nombre de points et formant une ligne fermée. Il a été étendu à des cas où les cotés se faisant face n'ont pas le même nombre de points. Le résultat obtenu n'est alors pas

toujours satisfaisant, il faut alors utiliser l'opérateur `SURF(ACE)`. On peut aussi définir une surface entre deux segments avec l'opérateur `REGL(ER)`, ou à partir de la translation ou de la rotation d'une ligne quelconque avec les opérateurs `TRAN(LATION)` et `ROTA(TION)`.

```
VOL1=SURF1 VOLUME 10 TRANLATION (0. 0. 10.) ;
```

L'opérateur `VOLU(ME)` permet de construire des volumes. Ceci peut se faire :

- par translation de surface suivant un vecteur avec l'option `TRAN(LATION)` comme dans notre exemple.

- par translation parallèlement à une génératrice avec l'option `GENE(RATRICE)` :

```
VOL2 = SURF1 VOLU 10 GENE LIG1 ;
```

- par rotation de surface entre deux vecteurs :

```
VOL3 = SURF1 VOLU 10 ROTA SURF2 ;
```

Le nombre 10 signale que l'on met 10 éléments dans la direction donnée. Le second et seul autre opérateur permettant de créer des volumes est l'opérateur `PAVE(R)` qui maille l'intérieur d'une surface parallélépipédique. Mais uniquement dans ce cas où les six faces ont été générées par `DALL(ER)`.

```
TRACER SURF1 QUALIFICATION ;
```

La directive `TRAC(ER)` trace un objet de type maillage (ici la surface de base). L'option `QUAL(IFICATION)` permet d'afficher les noms des objets visualisés à l'écran.

Voici quelques exercices sur le maillage.

2.3 Maillage d'un cube

Récupérer le fichier `maillage-cube.dgibi` correspondant à l'exemple décrit ci-dessus à partir de la page

<http://www.meca.unicaen.fr/castem/index.html>

et lancer `castem2000` en double-clicquant sur le fichier. Vous venez d'exécuter votre premier programme `castem2000`. A la fin de l'exécution, fermer l'application en cliquant sur la croix en haut à droite.

1. A l'aide d'un éditeur de texte, modifier le fichier `maillage-cube.dgibi` de façon à construire un parallélépipède de base carré de côté 5m et de hauteur 20m. Essayer d'avoir un maillage régulier (i.e. vraiment cubique!). Sauvegarder le fichier sous un autre nom.
2. On veut remplacer la commande `DALL(ER)` par `SURF(ACE)`. Commencez par consulter la documentation concernant ces commandes.
 - (a) Définir à l'aide de la commande `ET` un contour fermé carré.
 - (b) Puis définir la surface carré à l'aide de la commande `SURF(ACE)` avec l'option `'PLAN(E)'`.
 - (c) Essayer les autres options juste pour voir.

2.4 Maillage d'un cylindre

A l'aide des instructions `CERC(LE)`, `SURF(ACE)` et `VOLU(ME)` nous allons construire un cylindre de base circulaire de centre $(0, 0, 0)$ de rayon $2m$ et hauteur $10m$. Pour obtenir la syntaxe de ces commandes, consulter la documentation en ligne.

1. Pour commencer, construire 4 quarts de cercle avec la commande `CERC(LE)`.
2. Réunir les quatre morceaux précédent avec la commande `ET`.
3. Utiliser la commande `SURF(ACE)` avec l'option '`PLAN(E)`' pour construire un disque.
4. Achever la construction du cylindre à l'aide de la commande `VOLU(ME)`.
5. Construire un cylindre creux de hauteur $10m$ et dont la base est une couronne délimité par deux cercles de centre $(0, 0, 0)$ et de rayons respectifs $2m$ et $2.5m$. Indication : tracer les deux cercles comme précédemment, les réunir à l'aide de la commande `ET` puis définir la couronne à l'aide de la commande `SURF(ACE)`. Essayer d'obtenir un maillage régulier.

2.5 Maillage d'une plaque trouée

1. En dimension 2, créer le maillage d'une plaque carré de côté $5m$ troué d'un disque de rayon $1m$. Indication construire le contour du carré puis du cercle. Les réunir, puis utiliser ensuite la commande `SURF(ACE)`.
2. Même chose en dimension 3.

3 Premier calcul mécanique linéaire

3.1 Objectif du calcul

Le but de ce calcul est de voir sur un exemple très simple l'enchaînement des étapes nécessaires à un calcul par éléments finis par `CASTEM 2000`. Pour cela on veut calculer la déformée d'une poutre encastree à une extrémité et subissant une force fléchissante à l'autre extrémité.

Les données du problème sont :

$F = 10N$: force appliquée

$L = 1m$: longueur

$\rho = 7800kg/m^3$: masse volumique

$S = 1,392.10 - 4m^2$: section

$I_{yy} = 2,673.10 - 10m^4$: inertie suivant y

$I_{zz} = 2,673.10 - 10m^4$: inertie suivant z

$IT = 1,0.1010$: inertie de torsion

$E = 210000MPa$: module d'Young

Le résultat recherché est la flèche à l'extrémité.

3.2 Proposition de solution

```
*
* DEBUT DU FICHIER POUTRE.DGIBI
*
TITR 'PREMIER CALCUL MECANIQUE';
OPTI DIME 3 ELEM SEG2;
*
* GEOMETRIE
*
P1 = 0. 0. 0. ;
P2 = 1. 0. 0. ;
POUTRE1 = P1 DROI 10 P2;
*
* CHOIX DU COMPORTEMENT DU MATERIAU ET DU MODELE EF
*
M1 = MODE POUTRE1 MECANIQUE ELASTIQUE ISOTROPE POUT;
*
* DONNEES MATERIAU ET ELEMENT
*
CARMAT1 = MATE M1 YOUN 2.1E11 NU 0.3 RHO 7.8E3;
CAREF1=CARA M1 SECT 1.392E-4 INRY 2.673E-10 INRZ 2.673E-10 TORS 1.E10;
CARTOT1 = CARMAT1 ET CAREF1;
*
* CONDITIONS AUX LIMITES
*
COND1 = BLOQ DEPL ROTA P1;
*
* CHARGEMENT
*
FOR1 = FORC (0. 0. -10.) P2;
*
* RESOLUTION
*
RIGI1 = RIGI M1 CARTOT1;
RIGICL1 = RIGI1 ET COND1;
DEPL1 = RESO RIGICL1 FOR1;
*
* POST-TRAITEMENT
*
F1 = EXTR DEPL1 UZ P2;
F1 = 1.E3*F1;
```

```
MESS 'FLECHE EN P2' F1 'mm';
OEIL1 = 0. -1000. 0.;
DEFO = DEFO POUTRE1 DEPL1 0. BLAN;
DEF1 = DEFO POUTRE1 DEPL1 ROUG;
TRAC OEIL1 (DEFO ET DEF1);
```

```
*
```

```
* FIN DU FICHIER
```

```
*
```

FIN ; Vous pouvez récupérer ce fichier intitulé “poutre.dgibi” à la page

<http://www.meca.unicaen.fr/castem/index.html>

3.3 Analyse de la solution

1. Début de l'étude :

```
TITR 'PREMIER CALCUL MECANIQUE';
```

Donne un nom à l'étude.

```
OPTI DIME 3 ELEM SEG2;
```

Le problème mécanique nous amène à choisir des éléments finis de type poutre. La documentation (présentée en annexe 3) nous informe que les éléments géométriques correspondant à l'élément POUT sont des SEG2 et que la dimension requise pour ce type d'étude est 3.

2. Définition de la géométrie :

```
P1 = 0. 0. 0.;
```

```
P2 = 1. 0. 0.;
```

```
POUTRE1 = P1 DROI 10 P2;
```

La géométrie est décrite en plaçant les points extrémités P1 et P2, et en traçant une ligne droite entre P1 et P2 contenant 10 segments.

3. Choix du modèle de comportement et du type d'Eléments Finis :

```
M1=MODE POUTRE1 MECANIQUE ELASTIQUE ISOTROPE POUT;
```

L'opérateur MODE(LE) (cf annexe 3) sert à définir un type de comportement et une formulation E.F. qui seront affectés à un objet de type maillage. L'objet créé (ici M1) est de type MMODEL. On a affecté sur la géométrie POUTRE1 des éléments finis de type POUT et une loi de comportement mécanique élastique isotrope.

4. Entrée des caractéristiques des matériaux et de celles nécessaires au modèle E.F.

```
CARMAT1=MATE M1 YOUN 2.E11 NU 0.3 RHO 7.8E3;
```

MATE(RIAU) sert à définir les propriétés physiques du matériau (Module d'Young, coefficient de Poisson...) pour un modèle donné. L'objet créé est de type champ par élément (MCHAML) à plusieurs composantes : YOUN, NU, RHO, ALPH,...

```
CAREF1=CARA M1 SECT..... TORS 1.E10;
```

Certains éléments nécessitent la donnée de caractéristiques supplémentaires qui ne peuvent se déduire de la géométrie. Dans le cas des poutres (POUT) il faut préciser SECT, INRY, INRZ, TORS. Dans le cas des barres (BARR) il suffit de préciser la section. Dans le cas des coques (COQ2, COQ3, COQ4, DKT, COQ6, COQ8), il faut préciser l'épaisseur

(EPAI) et l'excentricité (EXCE).

CARTOT1=CARMAT1 ET CAREF1 ;

On prend en compte dans CARTOT1 l'ensemble des caractéristiques définies. On peut préciser ici que tout cela aurait pu s'écrire en une seule ligne :

CARTOT1=MATE M1 YOUN.....TORS 1.E10 ;

Cependant, il nous a semblé pédagogique de séparer les caractéristiques suivant leurs origines. D'autre part, il ne faut pas oublier qu'une ligne d'instructions ne doit comprendre que 72 caractères au maximum.

5. Conditions aux limites :

COND1=BLOQ DEPL ROTA P1 ;

Les conditions aux limites sont traitées dans CASTEM 2000 par la méthode des multiplicateurs de Lagrange. Elles sont écrites sous la forme : $[C]\{u\} = \{q\}$ L'utilisateur doit alors construire deux objets :

- (a) La rigidité $[C]$ à adjoindre à la rigidité du système libre grâce à l'opérateur BLOQ (UER)
- (b) Le vecteur $\{q\}$ à adjoindre au vecteur des forces nodales grâce à l'opérateur DEPI (MPOSE). Par défaut, ce vecteur est mis à zéro.

Dans le cas qui nous intéresse, les déplacements sont fixés nuls il n'y a donc pas besoin de créer le vecteur $\{q\}$. Signalons quelques autres opérateurs utilisables dans des cas particuliers :

SYMT permet de spécifier des conditions aux limites de symétrie par rapport à un plan ou une droite.

RELA permet d'imposer une combinaison linéaire quelconque des déplacements en différents points.

VISAVIS permet d'extraire les noeuds en vis à vis de 2 géométries.

6. Conditions de chargement :

FOR1=FORC (0. 0. -10.) P2 ;

Définir le chargement revient à définir un champ par point correspondant au vecteur du second membre de $[K]\{u\} = \{f\}$. Ici on applique une force de $-10N$ suivant l'axe z au point P2. Ceci s'effectue avec l'opérateur FORC (E). On aurait pu également écrire

FOR1=FORC FZ -10. P2 ;

La syntaxe de l'opérateur MOME(NT) est du même type. L'opérateur pression, sera quant à lui étudié ultérieurement.

7. Résolution : L'ensemble des données étant défini, on peut constituer le système $[K]\{u\} = \{f\}$ et le résoudre.

RIGI1=RIGI M1 CARTOT1 ;

L'opérateur RIGI (DITE) permet de construire la matrice de rigidité à partir du modèle et des caractéristiques relatives au modèle.

RIGICL1=RIGI1 ET COND1 ;

Comme nous l'avons dit dans le paragraphe conditions aux limites, il convient de prendre en compte la matrice des blocages au sein de la matrice rigidité.

DEPL1=RESO RIGICL1 FOR1 ;

L'opérateur `RESO(UT)` résout le système $[K]\{u\} = \{f\}$. Ces déplacements, solutions du problème, sont stockés dans `DEPL1`.

8. Post traitement :

```
F1=EXTR DEPL1 UZ P2 ;
```

L'opérateur `EXTR(AIRE)` permet d'extraire une composante d'un ensemble de valeurs. Ici on recherche le déplacement en `UZ` du point `P2` au sein du vecteur solution en déplacement `DEPL1`.

```
F1 = 1.E3 * F1 ;
```

La résolution par `RESO` donne des résultats en mètres avec notre choix d'unité. Pour l'avoir en *mm* on le multiplie donc par 1000.

```
MESS 'FLECHE EN P2' F1 'mm' ;
```

La directive `MESS(AGE)` permet d'afficher un message sur l'unité de sortie.

```
OEIL1 = 0. -1000. 0. ;
```

On définit ici le point à partir duquel sera vue la structure lors des visualisations. On peut ainsi définir de multiples points de vue.

```
DEFO = DEFO POUTRE1 DEPL1 0. BLAN ;
```

```
DEF1 = DEFO POUTRE1 DEPL1 ROUG ;
```

```
TRAC OEIL1 (DEFO ET DEF1) ;
```

L'opérateur `DEFO(RME)` construit la déformée d'une structure à partir de la géométrie initiale et d'un champs de déplacement. On peut également préciser un certain nombre d'options comme la couleur (ici `BLAN(C)` et `ROUG(E)`), ou le facteur d'amplification pour rendre les phénomènes plus visibles. Ici on utilise un facteur multiplicatif de 0 sur `DEFO`. Ceci permet de visualiser la structure non déformée en même temps que la déformée finale `DEF1`. Cet artifice est nécessaire car la directive `TRAC(ER)` ne peut être appliquée qu'à des objets de même type. On ne pourrait donc pas avoir : `TRAC OEIL1 (POUTRE1 ET DEF1)` car l'un est de type maillage et l'autre déformée. On remarque également l'utilisation du point `OEIL1` pour préciser le point de vue selon lequel on doit effectuer le traçage des déformés.

9. fin du programme :

```
FIN ;
```

La directive `FIN` permet de quitter `CASTEM 2000`.

3.4 Exercices

1. Modifier le fichier `poutre.dgibi` disponible en ligne afin d'étudier une poutre droite reposant sur deux appuis simples aux extrémités et soumise à une force ponctuelle au milieu.
2. Même chose avec cette fois-ci une densité linéique constante appliquée le long de la poutre (voir la commande `PRES(SION)`).