

MATHÉMATIQUES - ORAL II

INTITULÉ :

Cet oral II de mathématiques dure 1 heure, préparation incluse.

Il porte sur l'ensemble du programme de mathématiques. L'utilisation d'un logiciel de calcul formel peut être demandée dans le cadre du programme d'informatique des classes préparatoires.

OBJECTIFS :

Le but d'une telle épreuve est d'abord de contrôler l'assimilation des notions au programme de mathématiques de toute la filière (première et deuxième années). Il semble que certains candidats aient « oublié » ce qui a été vu en première année, voire les connaissances de base qui font partie du programme des classes du lycée (seconde, 1^{ère} et Terminale).

C'est aussi d'examiner :

- la capacité d'initiative du candidat,
- son aisance à exposer clairement ses idées et sa réactivité dans un dialogue avec l'examineur et, pour l'exercice « calcul formel », face à un logiciel,
- son aptitude à mettre en œuvre ses connaissances pour résoudre un problème (par la réflexion... et non par la mémorisation de solutions toutes faites) ainsi que sa maîtrise des calculs nécessaires,
- sa faculté à critiquer, éventuellement, les résultats obtenus et à changer de méthode en cas de besoin.

Pour la composante « calcul formel », le candidat n'est pas jugé sur une connaissance encyclopédique du logiciel mais sur son aptitude à utiliser cet outil de manière intelligente en utilisant des fonctions de base.

ORGANISATION :

Cet oral s'est déroulé dans des conditions identiques aux sessions précédentes. Comme d'habitude, il a eu lieu au centre de Paris de Arts et Métiers ParisTech, Boulevard de l'Hôpital à Paris (13^e).

Les candidats ont deux exercices à résoudre.

Comme les autres années, ces deux exercices se sont répartis de la manière suivante :

- Un exercice « classique » portant sur le programme de mathématiques des deux années de la filière PT,
- Un exercice « calcul formel », portant sur le même programme mais exigeant l'usage du logiciel de calcul formel (Maple ou Mathematica) dans le cadre du programme d'informatique. Pour cet exercice, les candidats disposent d'un ordinateur, du logiciel adéquat et d'une liste de fonctions et de mots-clé (voir en annexe). Ils ont accès à l'aide en ligne du logiciel.

Comme annoncé, lors de cette session 2010, les candidats avaient à leur disposition les logiciels suivants :

- Maple V version 5,
- Maple 11,
- Mathematica version 7.

L'année prochaine, ces versions (aussi bien pour Maple que pour Mathematica) seront les mêmes. Les examinateurs conseillent à ceux utilisant Maple 11 de choisir la version « Classic Worksheet ».

Les exercices posés aux candidats sont classiques et ne font appel à aucune astuce particulière. Ils ne nécessitent que l'application raisonnée – et souvent directe – de théorèmes du programme. Lorsque un théorème clairement hors-programme est utilisé par un candidat, des explications complémentaires, voire une démonstration, peuvent lui être demandées : le jury déconseille donc aux candidats l'utilisation de tels théorèmes.

Il n'y a, bien sûr, aucun rapport entre l'écrit et l'oral : les examinateurs n'ont pas en leur possession les notes des écrits de mathématiques des candidats et un exercice posé lors de l'oral peut très bien concerner une partie du programme qui a déjà été sollicitée lors d'une épreuve d'écrit.

COMMENTAIRES :

Certains candidats semblent avoir oublié qu'ils sont à un oral d'un concours recrutant de futurs ingénieurs, c'est-à-dire de futurs cadres supérieurs : on attend d'eux rigueur, expression (écrite et orale) claire, autonomie, réactivité et combativité.

Une attitude passive et sans réactions aux sollicitations et aux indications de l'examineur a toujours une conséquence négative importante au niveau de la note finale.

D'autres – ou les mêmes – donnent l'impression de « jouer la montre » en passant un temps important sur la (ou les) première(s) question(s), en général simple(s), et n'ont donc pas le temps nécessaire pour aborder les questions suivantes, plus intéressantes pour tester leurs connaissances. Cette attitude est évidemment sanctionnée.

Les connaissances d'un grand nombre de candidats restent souvent fragiles. La connaissance du cours se limite souvent à l'apprentissage de formules. Même chez les bons ou très bons candidats, on constate que le fond théorique qui justifie les méthodes ou les calculs utilisés est assez fragile. Beaucoup trop d'entre eux ignorent les hypothèses précises des théorèmes utilisés.

De nombreux candidats, pour répondre à la question posée, cherchent à « replacer » une solution vue lors d'un exercice au cours de l'année. Ils ne font donc pas d'analyse du problème et n'ont pas de réflexion sérieuse à ce propos. L'étape « Que me demande-t-on ? » est remplacée par « Est-ce que je l'ai déjà fait ? » et conduit presque toujours à un échec.

Même si une certaine technicité est indispensable, les examinateurs aimeraient surtout que les candidats comprennent ce qu'ils font et ce que signifient les notions utilisées, ce qui est loin d'être toujours le cas.

Les candidats semblent souvent déconcertés face à un exercice demandant une démonstration, situation qui se rencontre par exemple en algèbre. Ils semblent moins déroutés face à un exercice qui est l'application directe du cours et ne demandant que des calculs... même si, pour certains, le résultat est difficile à atteindre.

Les lacunes rencontrées sont toujours les mêmes. Beaucoup trop de candidats hésitent, se trompent, voire ne savent pas quoi faire quand il s'agit d'utiliser des « savoir-faire » de base. Après deux années de classes préparatoires, cela semble anormal et ne prédispose les candidats à poursuivre des études scientifiques sérieuses. On peut citer parmi ces « savoir-faire » :

- démontrer qu'une application est bijective,
- utiliser la formule du binôme,
- calculer dans ou à l'aide des nombres complexes,
- calculer un produit matriciel, faire un changement de bases,
- réduire une matrice,
- calculer une dérivée et faire l'étude d'une fonction,
- étudier la convergence d'une série numérique ou d'une intégrale impropre,
- effectuer un changement de variables dans un calcul de dérivées partielles,
- résoudre une équation différentielle linéaire,
- écrire l'équation d'une droite ou d'un cercle dans le plan,
- écrire l'équation d'une droite, d'un plan ou d'une sphère dans l'espace...

Plus généralement, aborder un exercice de géométrie est difficile pour un très grand nombre de candidats ; le programme de la filière PT contient pourtant une partie non négligeable de géométrie. Par exemple, nombreux sont ceux qui ne savent pas écrire l'équation d'une tangente ou d'une normale à une courbe plane, voire les confondent.

De même, les exercices comportant l'utilisation des nombres complexes semblent redoutables pour beaucoup... Cela est le cas aussi sur les exercices d'algèbre linéaire quand il ne s'agit pas d'un simple exercice de calcul... L'algèbre linéaire est pourtant une partie très importante du programme de mathématiques et a de nombreuses utilisations pour un ingénieur : l'une des démarches scientifiques les plus naturelles consiste à linéariser les problèmes avant de les résoudre, en général de manière numérique. Il est donc important qu'un ingénieur ait, sur ce sujet, des idées claires et des techniques maîtrisées.

Par ailleurs, la continuité ou la dérivabilité sous le signe \int pose beaucoup de problèmes aux candidats quant à la compréhension des hypothèses demandées.

Les examinateurs apprécient que les candidats utilisent un vocabulaire précis pour nommer leurs actions. D'ailleurs, on remarque souvent que la pauvreté du vocabulaire handicape le candidat pour trouver la fonction adéquate du logiciel dans l'exercice « calcul formel ». Les « fonctions » pour résoudre les équations, algébriques ou différentielles, sont ainsi souvent confondues.

Manifestement, certains candidats découvrent le jour de l'oral la liste de fonctions et de mots-clés. Elle est pourtant diffusée de manière publique (annexée à ce rapport) dans le but que les candidats se l'approprient et s'y retrouvent facilement. Découvrir une fonction de base, pour résoudre une équation par exemple, le jour de l'oral ne permet pas à un candidat d'être efficace... On rappelle que cette liste n'est ni exhaustive, ni limitative et qu'elle est fournie aux candidats pour les aider.

Les maladresses et les erreurs les plus fréquentes du point de vue « calcul formel » résultent d'une méconnaissance plus ou moins grande de :

- la notion de règle de substitution,
- la distinction entre expression et fonction,
- la définition des fonctions,
- la notion de liste (et de séquence en Maple),
- la définition et de la manipulation des vecteurs et des matrices,
- la récupération des solutions d'une équation...

Il est fortement conseillé aux candidats de nommer les résultats intermédiaires pour pouvoir les réutiliser : trop de candidats utilisent des « copier/coller » pour les réponses, pire les retapent « à la main » pour pouvoir les réutiliser. Une telle attitude est évidemment sanctionnée. Il est intéressant d'avoir une règle pour nommer : par exemple, nombreux sont les candidats qui désignent une matrice par la lettre « a » sans réaliser qu'ils ont déjà utilisé cet identificateur pour un des éléments de la matrice... Ils ont ensuite du mal à comprendre l'origine de leurs problèmes.

Certains ont tendance à « empiler » tous les calculs en une seule instruction ; cette habitude leur est toujours dommageable en cas d'erreurs car ils sont alors incapables de les trouver et donc de les rectifier : il est plus efficace de procéder par étapes et de vérifier, à chaque étape, la conformité du résultat avec ce qui est attendu.

Les examinateurs remarquent que de nombreux candidats ont un manque visible de pratique du logiciel : conformément au programme, cet apprentissage doit s'échelonner tout au long de l'année.

ANALYSE DES RÉSULTATS :

1396 candidats présents, répartis en 9 jurys, ont passé cet oral.

Les résultats sont :

Moyenne	10,50
Écart-type	4,02
Note minimale	1
Note maximale	20

La répartition des notes est la suivante :

$1 \leq n \leq 4$	$4 < n \leq 6$	$6 < n \leq 8$	$8 < n \leq 10$	$10 < n \leq 12$	$12 < n \leq 14$	$14 < n \leq 16$	$16 < n \leq 20$
107	144	190	238	239	247	140	91

Malheureusement, comme tous les ans, malgré le filtre de l'écrit, certains candidats vus à l'oral n'ont pas assimilé tous les concepts et les techniques mathématiques indispensables pour un futur ingénieur.

Les meilleurs candidats (avec une note ≥ 14 pour situer le niveau, soit environ 330 candidats cette année) ont donné l'impression d'avoir assimilé le programme – tout au moins sur les parties sur lesquelles ils ont été interrogés – et d'être à l'aise avec les concepts mathématiques, les techniques de calcul et l'utilisation du logiciel de calcul formel tout en étant dynamiques et réactifs. Quant aux autres, il est à craindre que certains parmi ceux qui intégreront une école aient des difficultés dues à leurs faiblesses en mathématiques.

CONSEILS AUX FUTURS CANDIDATS :

Les conseils que l'on peut donner aux futurs candidats sont des conseils de « bon sens » que leur ont certainement déjà donné leurs enseignants. Ce sont, bien sûr, toujours les mêmes :

- Travailler de manière régulière tout au long de l'année, y compris dans l'utilisation du logiciel de calcul formel : il doit être utilisé pour illustrer les différentes parties du cours et la compétence attendue ne s'acquiert pas en quelques jours, entre l'écrit et l'oral.
- Étudier soigneusement son cours, connaître les hypothèses précises d'application des théorèmes. Un énoncé de théorème n'est pas un texte vague que l'on peut utiliser comme incantation lors d'un exercice.
- À propos de chaque chapitre, faire un petit nombre d'exercices bien choisis et ne pas se contenter d'en lire une solution, aussi parfaite soit-elle. L'apprentissage des mathématiques, comme l'utilisation d'un logiciel de calcul formel, passe obligatoirement par la pratique. Il faut souvent avoir « séché » sur une question pour en comprendre la solution.
- Ne pas faire d'impasse dans le programme, y compris celui de 1^{ère} année... Toute partie peut faire l'objet d'un exercice à l'oral.
- Lors de la résolution d'un exercice, réfléchir pour savoir quelles parties du cours sont concernées, quels théorèmes vont s'appliquer, quelles méthodes sont possibles : ne jamais se lancer sans réflexion dans un calcul.
- Apprendre à présenter ses calculs et ses résultats sur un tableau de manière ordonnée et propre : le tableau ne doit pas être un brouillon lisible seulement par son auteur.
- S'entraîner à expliquer clairement d'une voix posée et audible le fil conducteur de ses calculs ou de sa démonstration lors d'une prestation orale, et cela sans « jouer la montre ». c'est-à-dire en évitant de passer un temps important sur des questions très simples.
- S'entraîner au calcul : utiliser les nombres complexes, réduire une matrice 3×3 , calculer un développement limité ou une intégrale, résoudre une équation différentielle linéaire d'ordre 2 à coefficients constants, donner l'équation d'un plan passant par 3 points...
- S'habituer à utiliser l'aide du logiciel de calcul formel à bon escient, par exemple pour chercher la syntaxe d'une option particulière. L'utilisation de l'aide ne doit pas servir à masquer une ignorance des connaissances de base.
- Après avoir obtenu un résultat, avoir un minimum d'esprit critique pour ne pas l'accepter s'il semble absurde ou impossible. C'est une qualité importante pour un futur ingénieur.

ANNEXE :

LISTE des OPERATEURS, FONCTIONS et MOTS-CLÉS

Ces listes sont mises à disposition des candidats sous la forme d'une feuille recto-verso pour chacun des logiciels avec d'un côté la liste où les mots sont regroupés par thème et de l'autre la liste alphabétique.

Il ne faut considérer ces listes :

- ni comme exhaustives (en cas de problème sur un exercice particulier, si une fonction indispensable était absente de la liste, l'interrogateur pourrait aider le candidat),
- ni comme exclusives (une fonction absente de ces listes n'est pas interdite : si un candidat utilise à très bon escient d'autres fonctions MAIS sait aussi répondre aux questions sur les fonctions de base, il n'y a pas de problème),
- ni comme un minimum à connaître absolument (l'examineur n'attend pas du candidat qu'il connaisse parfaitement toutes ces fonctions).

Elles doivent simplement permettre de faire les exercices proposés aux candidats.

Ces listes doivent permettre aux candidats de retrouver une fonction... ou d'en découvrir une nouvelle et lui permettre ainsi d'aller rechercher dans l'aide. L'examineur n'attend pas du candidat une connaissance encyclopédique mais une utilisation raisonnée du logiciel et une mise en pratique des connaissances de base. L'utilisation de ces listes et de l'aide en ligne ne doit pas masquer une ignorance sur ces aptitudes.

Elles sont données ici à titre indicatif.

MAPLE

Liste d'opérateurs, fonctions, mots-clés...

opérations :

+, -, *, /, ^, \$, @

opérateurs booléens (et constantes) :

=, <, >, <>, <=, >=, evalb, or, and, not, true, false

nombres réels et complexes :

evalf, Digits, Pi, I, abs, max, min, floor, evalc, conjugate, Re, Im, argument

manipulations d'expressions :

eval, lhs, rhs, " ou %, convert, combine, map, map2, nops, op, algsb, subs, remove, select

types de données :

float, integer, numeric, polynom, {}, [], seq, array, type, whattype

arithmétique :

iquo, irem, mod, ifactor

graphiques :

plot, pointplot, implicitplot, spacecurve, plot3d, bibliothèque plots, display

algèbre :

factor, expand, simplify, normal, coeff, coeffs, collect, sort, binomial, rem, quo, denom, numer, parfrac, product, sum, mul, add

analyse :

infinity, limit, asympt, series, taylor, D, diff, int, bibliothèque student, changevar

hypothèses :

assume, about, is, additionally

résolution d'équations :

solve, fsolve, allvalues, RootOf, dsolve, numeric

fonctions mathématiques :

sqrt, exp, ln, log, cos, sin, tan, arccos, arcsin, arctan, cosh, sinh, tanh, arccosh, arcsinh, arctanh

programmation :

restart, proc, local, break, RETURN, remember, :=, ->, unapply, print, do, for, while, if

algèbre linéaire :

Maple 5 et Maple 11 :

bibliothèque linalg, matrix, vector, evalm, transpose, diag, band, submatrix, augment, concat, matadd, scalarmul, &*, multiply, inverse, det, rank, kernel, linsolve, gausselim, charpoly, eigenvals, eigenvects, jordan, norm, dotprod, crossprod

Maple 11 :

bibliothèque LinearAlgebra, Matrix, Vector, << | >, < | >>, Transpose, DiagonalMatrix, BandMatrix, SubMatrix, MatrixAdd, ScalarMultiply, ., Multiply, MatrixInverse, Determinant, Rank, NullSpace, LinearSolve, GaussianElimination, CharacteristicPolynomial, Eigenvalues, Eigenvectors, JordanForm, Norm, DotProduct, CrossProduct

MAPLE

Liste d'opérateurs, fonctions, mots-clés... En * : seulement valable en Maple 11

^	coeff	implicitplot	polynom
*	coeffs	infinity	print
-	collect	int	proc
+	combine	integer	product
=	concat	inverse	quo
<	conjugate	iquo	rank
>	convert	irem	Rank *
,	cos	is	Re
;	cosh	jordan	rem
"	crossprod	JordanForm *	remove
%	CrossProduct *	kernel	restart
/	D	lhs	RETURN
:	denom	limit	rhs
&*	det	linalg	RootOf
.	Determinant *	LinearAlgebra *	Scalarmul
()	diag	LinearSolve *	ScalarMultiply *
->	DiagonalMatrix *	linsolve	select
{ }	diff	ln	seq
[]	Digits	local	series
<>	display	log	simplify
' '	do	map	sin
:=	dotprod	map2	sinh
< >	DotProduct *	matadd	solve
about	dsolve	matrix	sort
abs	eigenvals	Matrix *	spacecurve
add	Eigenvalues *	MatrixAdd *	sum
additionally	eigenvects	MatrixInverse *	sqrt
algsubs	Eigenvectors *	max	student
allvalues	eval	min	submatrix
and	evalb	mod	SubMatrix *
arcsin	evalc	mul	subs
arcosh	evalf	multiply	tan
arcsinh	evalm	Multiply *	tanh
arctan	exp	nops	taylor
arctanh	expand	norm	transpose
array	factor	Norm *	Transpose *
asympt	false	normal	true
assume	float	NULL	type
augment	floor	NullSpace *	unapply
band	for	numer	vector
BandMatrix *	fsolve	op	Vector *
binomial	gausselim	or	whattype
break	GaussianEliminat	parfrac	while
changevar	ion *	Pi	with
charpoly	I	plot	
Characteristic	if	Plot3d	
Polynomial *	ifactor	plots	
	Im	pointplot	

MATHEMATICA

Liste d'opérateurs, fonctions, mots-clés...

opérations :

+, -, *, /, ^,
(), %, i,
Evaluate

opérateurs booléens (et constantes) :

==, !=,
<, >, <=, >=,
Or, ||, And, &&,
Not, !,
True, False

nombres réels et complexes :

N,
Pi, I, E,
Abs, Max, Min,
Floor,
Conjugate,
Re, Im, Arg, ComplexExpand

manipulations d'expressions :

{}, Table, Array,
Length, Part, [[]],
AppendTo, Select,
Map,
ReplaceAll, /.,
FullForm,
Simplify,
FullSimplify,
Remove, Sort

arithmétique :

Quotient, Mod,
FactorInteger

algèbre :

Factor, Expand, Coefficient,
CoefficientList,
Collect,
PolynomialQuotient,
PolynomialRemainder,
Binomial,
Denominator,
Numerator,
Apart, Together,
Product, Sum

analyse :

Infinity,
Limit, Series,
Normal,
D,
Integrate,
NIntegrate

résolution d'équations :

Solve, NSolve,
FindRoot,
DSolve, NDSolve

fonctions mathématiques :

Sqrt, Exp, Log,
Cos, Sin, Tan,
ArcCos, ArcSin,
ArcTan,
Cosh, Sinh, Tanh,
ArcCosh, ArcSinh,
ArcTanh

graphiques :

Plot, ListPlot,
ParametricPlot,
Plot3D,
Show

algèbre linéaire :

Transpose, IdentityMatrix,
DiagonalMatrix,
MatrixForm,
Inverse, Det,
MatrixPower,
NullSpace,
RowReduce,
LinearSolve,
CharacteristicPolynomial
Eigenvalues,
Eigenvectors,
Eigensystem,
JordanDecomposition

géométrie :

Dot, Cross

programmation :

Module,
Break, Return,
=, :=,
[], Blank, _,
#, &,
Print,
Do, For, While,
If

MATHEMATICA

Liste d'opérateurs, fonctions, mots-clés...

^	Collect	Mod
*	ComplexExpand	Module
-	Conjugate	N
+	Cos	NDSolve
=	Cosh	NIntegrate
<	Cross	Normal
>	D	Not
.	Denominator	NSolve
,	Det	NullSpace
;	DiagonalMatrix	Numerator
%	Do	Or
/	Dot	ParametricPlot
!=	DSolve	Part
()	Eigensystem	Pi
->	Eigenvalues	Plot
==	Eigenvectors	Plot3D
<=	Evaluate	PolynomialQuotient
>=	Exp	PolynomialRemainder
{ }	Expand	Print
[]	Factor	Product
/.	FactorInteger	Quotient
:=	False	Re
[[]]	FindRoot	Remove
!	Floor	ReplaceAll
	For	Return
&&	FullForm	RowReduce
Abs	FullSimplify	Select
And	I	Series
Apart	IdentityMatrix	Show
AppendTo	If	Simplify
ArcCos	Im	Sin
ArcCosh	Infinity	Sinh
ArcSin	Integrate	Solve
ArcSinh	Inverse	Sort
ArcTan	JordanDecomposition	Sqrt
ArcTanh	Length	Sum
Arg	Limit	Table
Array	LinearSolve	Tan
Binomial	ListPlot	Tanh
Break	Log	Together
CharacteristicPolynomial	Map	Transpose
Coefficient	MatrixForm	True
CoefficientList	MatrixPower	While
	Max	
	Min	