



MAPLE ACADEMIC CONFERENCE 2012

【数学編】

活用事例

学習院大学 理学部 数学科

『数学講話 1』

プログラミングスキルの向上を考慮した Maple 操作ガイドの開発とその活用

平成24年10月15日（月）
富士ソフトビル・アキバプラザ

加藤克也
営業推進室
モデルベース開発推進事業部
サイバネットシステム株式会社

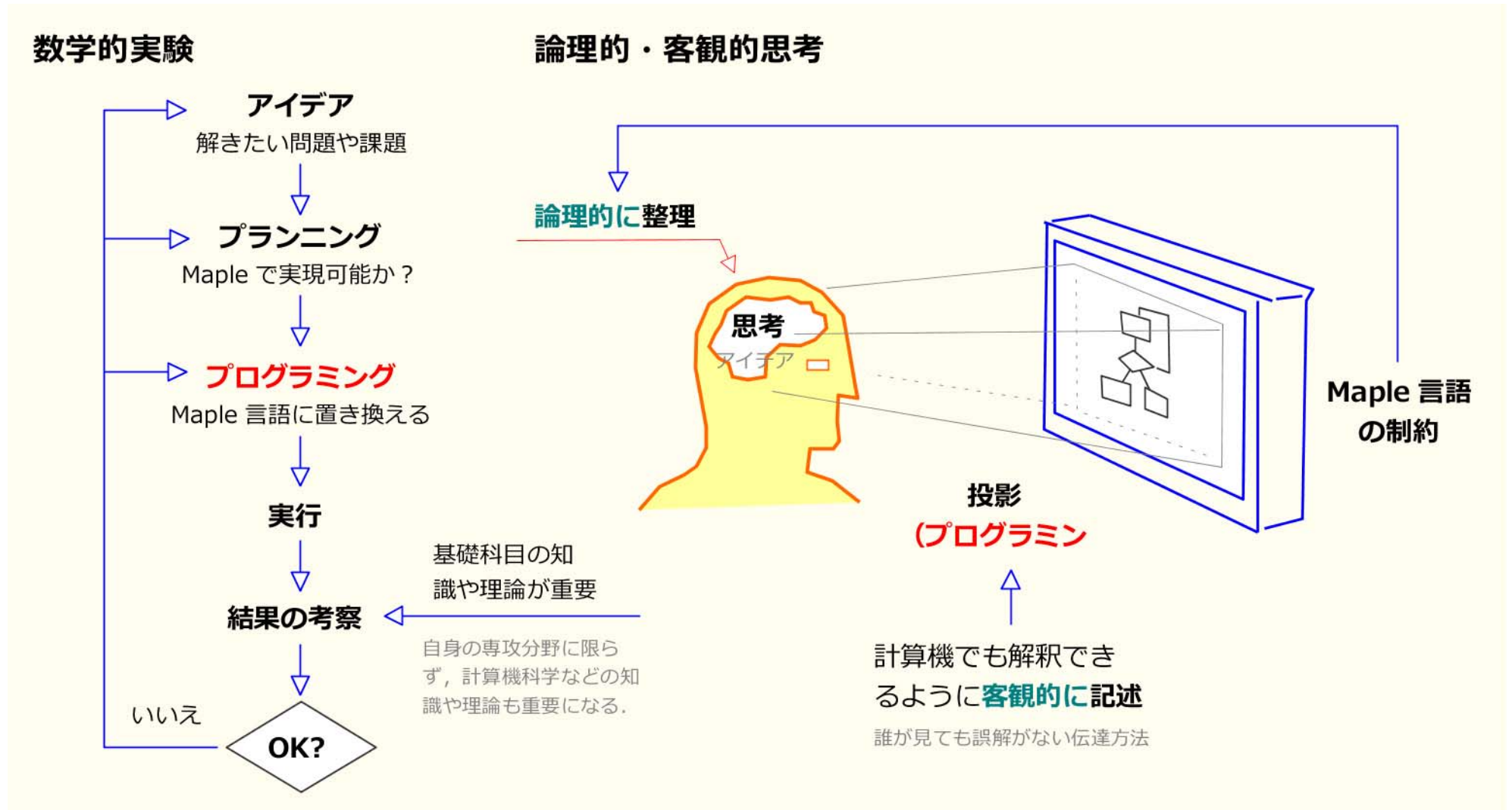
Copyright © 2012 CYBERNET SYSTEMS CO.,LTD. All rights reserved.

▼ アウトライン

1. 講義の概要と実績					
設計項目	講義	テキスト	課題	評価方法 (自動採点)	CBT (Maple TA)
要求項目 (各設計において考慮した項目)	2. 講義とテキストの設計 (同時に課題も検討)		3. 課題と採点方法の設計 (2012年度に再設計)		4. テストの設計 (2012年度に新規導入)
Maple の基本操作を習得	段階的に基本 コンセプト・ 操作を習得	段階的に基本 コンセプト・ 操作を解説	課題で確認		CBT で確認
プログラミングスキルの向上	段階的にプログ ラミングを体験	段階的にプログ ラミングを解説	課題で確認		CBT で確認
ものづくりの中の数学 (システム設計の視点で)	時間枠を用意	関連知識を解説	課題で確認		

▼ 講義におけるプログラミングの位置づけ

- 数学的実験を Maple で実現するための**基本スキル**
- **論理的・客観的思考**を涵養するため（涵養するための）**手段**



▼ 1. 講義の概要と実績 1/4

▼ 1.1 概要

スーパーバイザー： 学習院大学理学部数学科 中島匠一教授

▼ 1.1.2 シラバス（2012年度） 1/2

数学講話 1 — 数学的実験のためのリテラシー —

担当者	単位数	配当年次		日時	時限
加藤克也	2	2	集中（第2学期）	9月10日～13日	未定

授業の目的

数式処理ツール Maple の利用を通して、コンピュータによる数学的実験のリテラシーを養います。

授業内容

1. Maple を使う前に
2. 運動を見る！
3. ワークシートを操る！
4. Maple の仲間たち
5. やっぱり微積分
6. いわゆる線形代数
7. 微分方程式に挑む！
8. プログラミング前夜
9. みんなの Maple

[補足] 高校数学（微積分や行列など）および高校物理（力と運動）に関する基礎知識があり、パソコンの操作に慣れていることが望ましい。

▼ 1. 講義の概要と実績 2/4

▼ 1.1 概要

▼ 1.1.3 シラバス (2012年度) 2/2

授業方法

各項目の説明と数式処理ツール Maple の利用を交互に繰り返しながら進めます。

成績評価の方法・基準

出席および提出課題による総合評価
課題は講義中に指示します。

履修上の注意

第1回目の授業に必ず出席のこと。

参考： 学習院大学理学部シラバス, 数学講話 1, <http://www.math.gakushuin.ac.jp/Syllabus/0543041100100.html>

▼ 1. 講義の概要と実績 3/4

▼ 1.2 実績

ハイライト：

- 9月開講の集中講義（Maple 初級者講習会）
- 2007年度よりサイバネットシステム株式会社より講師（インストラクター）を派遣
- 2009年度より担当交代。このとき現在の講義テキストを新規作成（要求3項目に基づく）
- 2010年度に「棒立ての数学」を追加
- 2011年度に提出課題の自動採点システムを設計・開発・導入
- 2012年度に CBT（Maple T.A.）を導入

	2007.9	2008.9	2009.9	2010.9	2011.9	2012.9
日数	5	5	5	4	4	4
担当者	山口哲	山口哲	加藤克也	加藤克也	加藤克也	加藤克也
TA(講義補助)	(不明)	(不明)	2	2	2	3
コンテンツ	新規作成 (PowerPoint)	-	新規作成 (Maple)	「棒立ての数学」 の追加 (Maple)	-	-
その他	-	-	-	-	自動採点の導入	CBT の導入
出席者数(登録者数)	-(-)	-(-)	33(43)	47(64)	41(55)	48(71)
学科	数学科 (その他は不明)	数学科 (その他は不明)	数学科 物理学科 化学科	数学科 物理学科	数学科 物理学科 生命科学科	数学科 物理学科 生命科学科
学年	2 (その他は不明)	2 (その他は不明)	1, 2, 3, 4	1, 2, 3, 4	1, 2, 3, 4	1, 2, 3, 4

▼ 1. 講義の概要と実績 4/4

▼ 1.3 講義の進め方

- Mape で作成したテキスト（ワークシート）を PDF ファイルに変換して配布します。ただし、コマンドをテキストからコピー・ペーストできないように PDF ファイルに設定を追加します。
- 講義中に指示したコマンドを入力するための Maple ワークシートを配布します。このワークシートは課題として講義終了後に提出してもらいます（この課題を「芸術課題」あるいは「art 課題」と呼びます）。

▼ 1.4 成績評価の方法

出席	40 %	
課題	50 %	<ul style="list-style-type: none">• 芸術課題（art 課題）： 講義中に入力・実行した処理を第3者でも理解できるように整理する。• 計算課題（cal 課題）： 欠けている処理を正しく入力し全体の処理を完成させる（理科ノート方式）。
CBT	10 %	<ul style="list-style-type: none">• オンラインテスト環境 Maple T.A. を使用

▼ 1.5 テキストのダウンロード

- 講義で使用しているテキストは、学習院大学理学部数学科のウェブサイト（数学教室）からダウンロードすることができます。ダウンロードサイト内の注意事項に従ってご自由にご利用いただけます。

ダウンロードサイト： <http://www.math.gakushuin.ac.jp/Maple/>

【注意】 「棒立ての数学」や「課題」は含まれていません。

▼ 2. 講義とテキストの設計 1/5

▼ 2.1 要求

- 以下の要求項目を網羅すること

▼ 2.1.1 Maple の基本操作を習得

- 学生が Maple の基本コンセプト（数式処理など）をよく理解して、Maple の基本的な操作（立式、数式処理、数値計算、あるいは可視化）で迷わなくなること（学生が Maple をツールとして使いこなせるようになること）

▼ 2.1.2 プログラミングスキルの向上

- 学生自身が Maple を用いて数学的実験を行えるようになること
- 処理やアイデアの論理的あるいは客観的な整理方法を訓練すること

▼ 2.1.3 ものづくりの中の数学

- ものづくり、特に製品開発のシステム設計における物理モデル（常微分方程式）の導出とそのモデルをベースに行うシステムの解析や設計から、ものづくりと数学の関係を理解すること

▼ 2.1.4 その他

- 扱う問題は既知の問題であること（答えが簡単に予測できる問題を中心に取り上げることで、まずは Maple をツールとして使用することに慣れる）
- テキストは自習にも用いることができること（個々のコマンドやオプションの機能を個別に Step-by-Step で理解できるようにすること）

▼ 2. 講義とテキストの設計 2/5

▼ 2.2 設計

• 以下のような講義，テキスト，（計算）課題の関係と構成をもとにして，講義，テキスト，課題を設計します。

	主な講義内容	テキスト		（計算）課題
1 日 目	<ul style="list-style-type: none"> 講義の進め方，課題の提出方法，評価の方法など 数式処理と数値計算 基本的な入力方法やコマンドの実行方法 	1. Maple を使う前に	Maple の基本コンセプトや基本操作を理解します。	
	<ul style="list-style-type: none"> ワークシートの操作方法（セクションやショートカットなど） 	2. ワークシートを操る！	ワークシートの基本的な操作方法を習得します	
	<ul style="list-style-type: none"> 投げ上げ運動の可視化（立式→計算→可視化） 	3. 運動を見る！	基本的な数式処理、数値計算、可視化の流れを理解する	A. 放物線運動の可視化
2 日 目	<ul style="list-style-type: none"> 数学定数や複素数（複素平面）など 多項式，方程式，関数，文字列，リスト，ブーリアン，型など 	4. Maple の仲間たち	様々な Maple オブジェクトの定義方法を習得する	
	<ul style="list-style-type: none"> 数列，極限，級数展開，微分，積分など 	5. やっぱり微積分	微積分に関する基礎的操作を習得する	B. テーラー展開による線形近似の可視化
3 日 目	<ul style="list-style-type: none"> ベクトルや行列の計算，固有値の計算，あるいは連立1次方程式の解法 	6. いわゆる線形代数	線形代数に関する基礎的操作を習得する	
	<ul style="list-style-type: none"> 常微分方程式の立式と解法，初期値問題の解法，あるいは結果の可視化など 	7. 微分方程式に挑む！	微分方程式の定義方法とその解法を習得する	C. 微分方程式の導出と求解（物理モデリング・シミュレーション）
	<ul style="list-style-type: none"> 棒立ての物理モデリング（制御系設計も含める） 	棒立ての数学	（資料の配布はしない）	（B と C を基礎知識とする）
4 日 目	<ul style="list-style-type: none"> 処理の再利用（プロシージャ） グローバル変数とローカル変数 制御構文（条件分岐・繰り返し）やフローチャートなど 	8. プログラミング前夜	Maple 言語の基礎を習得する	D. 連立方程式の解法（線形代数・プログラミング）
	<ul style="list-style-type: none"> 計算アプリケーション（GUI 環境）の作成 	9. みんなの Maple	Maple の GUI コンポーネントの基礎を習得する	

▼ 2. 講義とテキストの設計 3/5

▼ 2.2 設計

▼ 2.2.1 プログラミングスキル習得の流れ

目的： プロシージャ（ユーザ定義関数）を作成できるようになること

1. 処理の整理

個別に実行していた複数の処理をひとつの実行グループに整理する（意味のある処理をひとつの実行グループに整理する）

2. パラメータの抽出

その処理から変更する式，値，あるいはオプションをパラメータとして抽出する（最終的に各パラメータはプロシージャの引数となる）

3. 処理の再利用

問題を変えて 1 と 2 を繰り返す。はじめに与える入力（式や値）によって出力が変わることを確認する。

4. プログラミング

プログラミング技法を少しずつ取り入れる。ローカル・グローバル変数，引数，制御構文，プロシージャなどの機能を一つひとつ確認する。

5. プロシージャ化

ここまで学習した知識を応用して，微分方程式の求解（立式から可視化まで）をプロシージャにまとめる。ここで，デバッグを体験する。

▼ 2. 講義とテキストの設計 4/5

▼ 2.3 実現

テキスト一覧

1. [Maple を使う前に](#)
 2. [ワークシートを操る！](#)
 3. [運動を見る！](#)
 4. [Maple の仲間たち](#)
 5. [やっぱり微積分](#)
 6. [いわゆる線形代数](#)
 7. [微分方程式に挑む！](#)
 8. [プログラミング前夜](#)
 9. [みんなの Maple](#)
- [棒立ての数学](#)

▼ 2. 講義とテキストの設計 5/5

▼ 2.4 運用

- その日の課題をこなすのに必要な処理を中心に、上記テキストに含まれる例題を（自身のワークシートに）書き写して実行します。

	Mapleの基本操作を習得	プログラミングスキルの向上	ものづくりの中の数学
1 日 目	1. Mapleを使う前に		
	2. ワークシートを操る！	(ワークシート内の構造化 = 処理のグループ化)	
	3. 運動を見る！	(立式から数式処理・数値計算を経て可視化までの処理を整理)	
2 日 目	4. Mapleの仲間たち		(複素数のプロット・2次方程式の求解)
	5. やっぱり微積分		(テーラー展開を用いた三角関数の線形近似) (指数関数の性質)
3 日 目	6. いわゆる線形代数		(固有値の計算)
	7. 微分方程式に挑む！		(常微分方程式の求解と計算結果の可視化)
4 日 目	8. プログラミング前夜	(行列データの繰り返し処理) (制御構文・プロシージャ化)	棒立ての数学 (解説のみ・動画あり) (振動系の安定性)

繰り返し

(動画)

▼ 3. 課題と採点方法の設計 1/3

▼ 3.1 要求

- その日の講義内容の復習になること
- 情報を整理する訓練になること
- プログラミングスキルの向上に繋がること
- 問題にあわせてコマンドを正しく使用できるかどうかを確認できること

▼ 3.2 設計

▼ 3.2.1 2種類の課題

芸術課題 (art 課題)

- 可読性が高いかどうか (美しいかどうか)
 - 「講義」セクション
 - > 講義中に指示したサンプルを入力, 実行する
 - 「課題 1」セクション
 - > テキスト内の例題 (サンプル) を入力, 実行する
- 備考
 - できるだけわかりやすく整理する
 - 必要に応じて文章やコメントなどを追加する
 - 結果の正確さは“あまり”対象にしない.

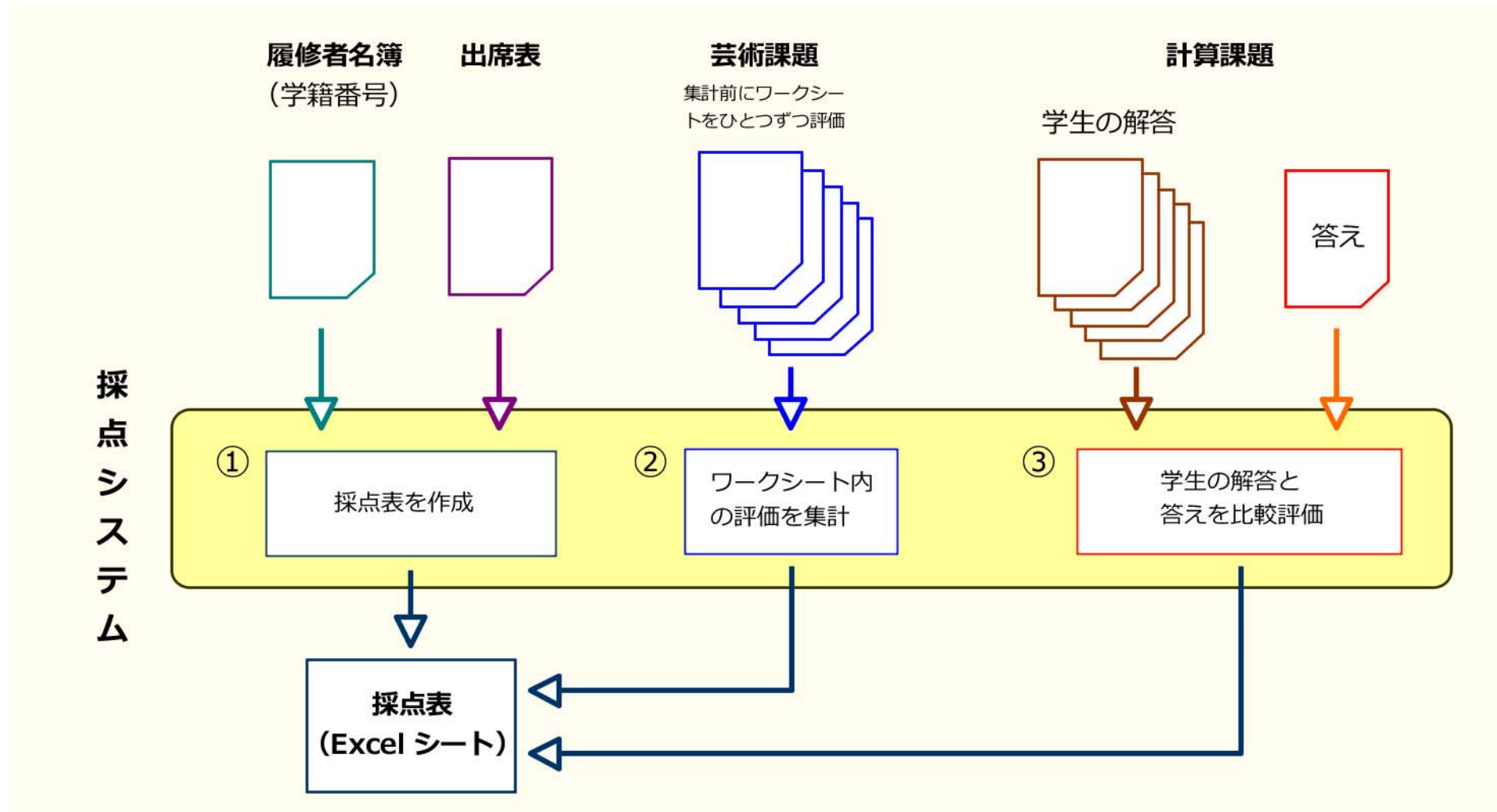
計算課題 (cal 課題)

- 入力 (計算結果) が正確かどうか
 - 「課題 2」
 - > 講義内容に関連した課題を指示された手順に従って整理, 実行する
- [備考]
 - 採点は自動的に行われますので, 結果の正確さだけが対象になる

▼ 3. 課題と採点方法の設計 2/3

▼ 3.2 設計

▼ 3.2.2 採点のプロセス



▼ 3. 課題と採点方法の設計 3/3

▼ 3.3 実現

▼ 3.3.1 芸術課題（art 課題）

- [芸術課題](#)は1日目～4日目まで同じフォーマットを使用

▼ 3.3.2 計算課題（cal 課題）

- 講義内容に応じて、4つの課題を作成.

1日目: [放物線運動の軌跡](#) (答え)

2日目: [級数展開の可視化](#) (答え)

3日目: [微分方程式の求解](#) (答え)

4日目: [連立方程式の解法](#) (答え)

▼ 3.3.3 採点用ワークシート

- 芸術課題, 計算課題を[採点するためのワークシート](#)を作成

▼ 3.4 運用

▼ 3.4.1 芸術課題（art 課題）

- **講義開始時**に共有ドライブを介して**学生に配布**.
- テキスト内の処理（コマンド）をワークシートにタイプして実行.

▼ 3.4.2 計算課題（cal 課題）

- **講義終了後**に共有ドライブを介して**学生に配布後**、**課題の説明**.

▼ 4. テストの設計

▼ 4.1 要求

- 講義で説明した基本的な数式処理、可視化、あるいはプログラミングの手順を正しく理解しているかどうかをテストする.
- 入力モードはテキスト入力 (例えば、`> eq := x^2 + p*x + q = 0;`) とする.
- Maplesoft 社に設置された [Maple T.A. サーバ](#) を (一種のアプリケーションサービスとして) 使用する.

▼ Maple T.A. について

- T.A. = Testing and Assessment (Testing, Evaluation, Grading Software)
- オンライン型テスト・採点環境
- 問題の作成や設定はウェブ上ですべて可能
- 学生の習熟度に応じて問題設定が可能 (結果を判定して問題の条件分岐が可能)
- ランダムに出題の順番や問題内の値などを設定可能
- 採点評価 (統計処理) の自動化
- 採点表を Excel 形式にして出力可能

▼ 4.2 設計

- Maple T.A. をベースにしたテスト問題を設計する.
- 以下のような項目を確認する.
 - プロット, データの定義, 初等関数の定義, 行列の定義と計算, 微分方程式の求解, 繰り返し処理など

▼ 4.3 実現

- 『Maple 検定 3 級』として、[Maple T.A.](#) (Maplesoft 社のサーバ) に問題を 10 問作成.

▼ 4.4 運用

- 講義最終日 (4 日目) に 30 分程度で実施.

▼ 5. おわりに

▼ まとめ

- Maple の基本操作習得を主目的としながらプログラミングスキルの向上にも配慮した講義、テキスト、及び課題を設計・開発
- 講義内容の範囲内において「ものづくりと数学」というテーマを導入
- 講義をサポートするツールとして、自動採点システム（2011年度から）、オンラインテスト環境（2012年度から）を導入

▼ CYBERNET のミッション

ここから始まる、次のイノベーション

CYBERNET × Maple で ひとづくり

CYBERNET は時代が求める理工系人材教育を
先進的なコンピュータ技術で強かにサポートします。