
論理思考とプログラミング

第1回－1時限目

イントロダクション

Logical Thinking



今日の授業の流れ

Logical Thinking

① 1時限目

● 講義

- 「論理思考とプログラミング」についての解説
- お絵かきプログラム開発演習についての解説

● 演習

- お絵かきプログラム開発演習

② 2時限目

● 講義

- 2時限目の演習範囲の解説

● 演習

- 付録B Macで使うには(「ことだま on Squeak」のインストール)
- Project1 車を描いてみよう
- Project2 車を動かしてみよう



配布資料

Logical Thinking

- ① 第1回 演習チェックシート
 - 演習で利用します
 - 使い方は後ほど説明します
- ② お絵かきプログラム開発演習セット
 - 全部で7枚あります
 - 演習で利用します
 - 受け取ったら、一枚ずつ切り離してください

Logical Thinking



1時限目の目標

- 「論理思考とプログラミング」の授業の概要を理解する
- プログラミングの全体像を知り、一通りの体験をする
- 同じクラスの人と仲良くなる

Logical Thinking

論理思考とプログラミング

Logical Thinking

- 単位数
 - 4単位
- 履修条件
 - 履修申告時までには情報技術認定試験に合格していること
- 対象者
 - 論理思考能力を身につけたい学部生
- 人数制限
 - 1クラスは「特別教室に座れる人数」の約30名
 - 事前にWebの希望クラス登録システム (<http://ipl.sfc.keio.ac.jp>) で、希望クラスを登録すること

授業の目標

Logical Thinking

- ① 論理思考を身につける
 - 問題から解くべき課題を設定して、合理的な方法で課題を解決し、それによって問題が解決できたかを検証できるようになる
- ② プログラムが作れるようになる
 - 自分でアルゴリズム(仕事の手順)を考案し、それを他人やコンピュータに伝わるように正確に、完全に記述できるようになる

Logical Thinking

授業の大まかな流れ

Logical Thinking

- 前半の6回分:「ことだま on Squeak」編
 - プログラム言語の知識が不要な「ことだま on Squeak」を用いる
 - プログラムの基本的な概念(制御構造, 変数)を習得し, 基本的なアルゴリズム(検索・整列)を自分で構築した上で, それを応用してみる
- 後半の7回分:「Java」編
 - 広く使われている実用プログラミング言語「Java」を用いる
 - 前半の内容をJavaを使いながら復習する
 - プログラムの部品化等, 前半で学習できなかった事柄を学習する
 - 個人ミニプロジェクトを実施する

Logical Thinking

授業スケジュール

Logical Thinking

- 前半:「ことだま on Squeak」編
 - 第1回 イントロダクション, 初めてのプログラミング
 - 第2回 制御構造を理解する
 - 第3回 記憶装置(変数)を理解する
 - 第4回 記憶装置(変数)を使いこなす
 - 第5回 アルゴリズム概念の理解と構築
 - 第6回 アルゴリズムの発明と構築
- 後半:「Java」編
 - 第7回 Javaプログラミング入門
 - 第8回 Javaによる制御構造の表現
 - 第9回 オブジェクトの概念習得とゲーム作成
 - 第10回 プログラムの部品化①
 - 第11回 プログラムの部品化②
 - 第12回 Javaの入れ物を用いたデータ構造とアルゴリズム構築
 - 第13回 プロジェクト成果発表とまとめ

評価方法

Logical Thinking

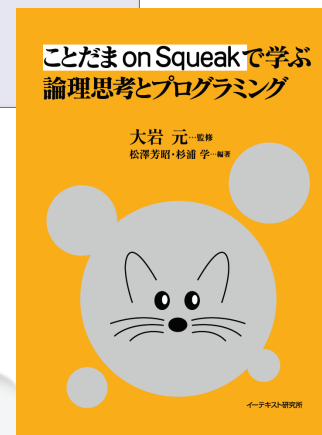
- 3つの提出物で評価
 - その1: 中間レポート
 - 授業前半で学んだことをまとめる
 - 第6回の授業で告知予定
 - その2: 最終プロジェクト
 - Javaを利用して、個人でプログラム作品を制作する
 - 第13回の授業で発表してもらう予定
 - その3: 最終レポート
 - 授業後半(全体)で学んだことをまとめる
 - 第13回の授業で告知予定
- 試験はなし
- クラスによって通常課題や出席点が加算される場合あり

教材

Logical Thinking

前半:「ことだま on Squeak」編

- プログラミング環境
 - ことだま on Squeak
 - 授業ページよりダウンロード可能
- テキスト
 - ことだま on Squeakで学ぶ論理思考とプログラミング
 - 生協で購入可能
 - 忘れた場合は授業ページからPDFが閲覧できる



後半:「Java」編

- プログラミング環境
 - Java
- テキスト
 - タートルとJavaで学ぶ論理思考とプログラミング
 - Webで公開予定

Logical Thinking

授業のWebページ

Logical Thinking

① <http://lms.crew.sfc.keio.ac.jp>

- 2010年度 秋学期の「論理思考とプログラミング（共通）」から閲覧可能（授業のトップページをBookmarkしておくとう便利）
- 教材や配布資料が閲覧できる
- 授業で使うソフトウェアがダウンロードできる

The screenshot shows the LMS interface for the course '論理思考とプログラミング(共通)'. The page title is '2010年度 秋学期 論理思考とプログラミング'. There is a navigation menu on the left with 'コース' (Courses) and a list of semesters from 2007 to 2010. The main content area displays the course title and a button labeled '論理思考とプログラミング'. Below the main content, there are links for '共通授業シラバス' (Common Course Syllabus) and '曜日・担当者・連絡先・教室一覧' (Day, Instructor, Contact, and Classroom List). The top right corner shows the user is logged in as 'ゲストユーザ' (Guest User).

1～6回までの授業スタイル

Logical Thinking

- テキスト(ことだま on Squeak で学ぶ論理思考とプログラミング)に沿って授業を進める
 - 授業ページのPDF版のテキストを閲覧するか、テキストを購入し、演習を実施する
- 講義(10分程度)、演習(80分程度)を2回繰り返す
 - 講義
 - テキストの会話部分で紹介されている例題の解説
 - 指定問題(テキスト中で必ずやらなければならない問題)と発展問題(時間があまったら取り組む問題)の提示
 - 演習
 - 各自でテキストを読み進めながら演習を行う、随時講師やSA・TAに質問する
 - テキストを読みながら、例題を真似して作ってみる
 - 指定問題や発展問題に解答する
 - 問題ができたらチェックシートに第三者(講師・TA・SA等)にサインをもらって、授業後に提出
 - 宿題
 - 指定問題がその日の授業時間に終わらなかった場合、授業に欠席した場合は、次の週までに自宅や学校で取り組むことを宿題とする
 - 授業時間内に全ての指定問題をこなせば宿題はなし

お絵かきプログラム開発演習

Logical Thinking



演習の目的

Logical Thinking

- プログラミングの全体像を知り，一通りの体験をする
- 同じクラスの人と仲良くなる

Logical Thinking

演習の準備

Logical Thinking

○ 準備

● グループを組む

- 5～7人のグループに分かれてください(原則5人)
- グループで円形に紙をまわせるように、席を若干グループの中央に向けてください

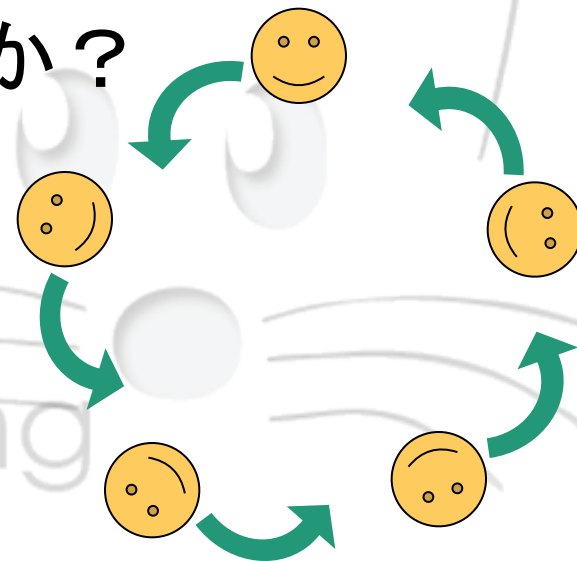
● 教材の確認

- 配布したお絵かき演習セット(7枚)
- 持っていない人・足りないものがある人は手をあげてください

練習

Logical Thinking

- 紙の回し方を確認しましょう
 - 説明書に名前を書いてください
 - 右隣の人に説明書を回してください
 - 自分の紙が戻ってくるまで繰り返しましょう
- 紙の回し方は確認できましたか？



Logical Thinking

お絵かきプログラム開発演習

【演習スライドへ】

Logical Thinking



ソフトウェア開発の問題

Logical Thinking

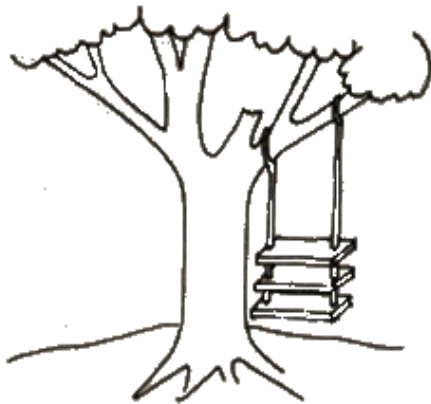
- このような要求から始まったプロジェクトはどのようなでしょうか？

ある小さな女の子の要求
「3通りの遊び方ができる
ブランコが欲しいの」



ソフトウェア開発の問題

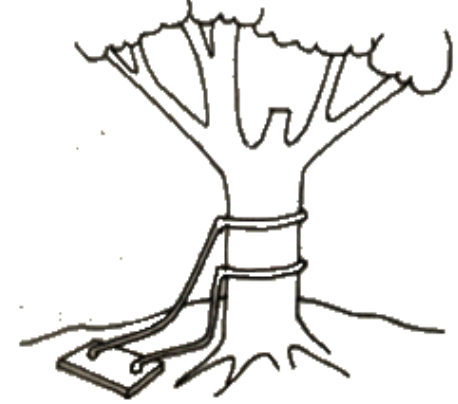
Logical Thinking



プロジェクト依頼者の考えていたこと



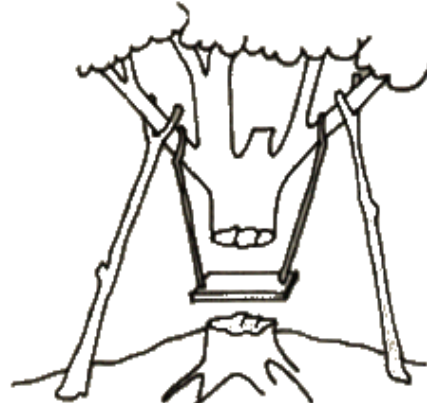
プロジェクト要求書に書かれていたこと



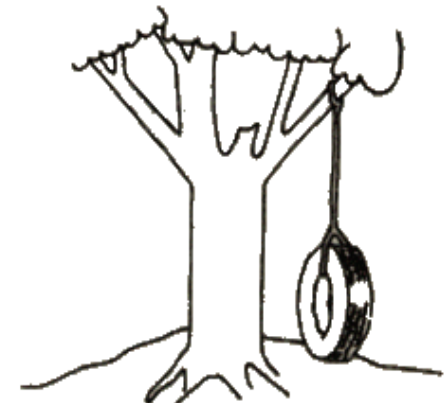
システム分析者が設計したもの



プログラマが作ったプログラム



利用者側に導入されたプログラム

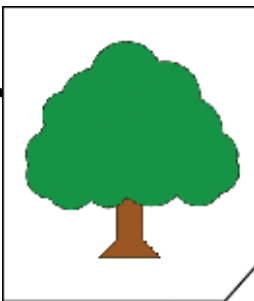


利用者が要求していたもの

まとめ:プログラミングの全体像

🕒 プログラミングの全体像を知り、一通りの体験をした

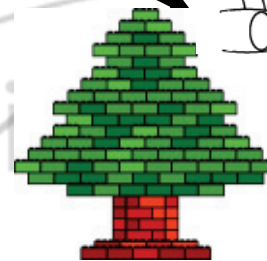
要求分析 — アイディアを明確にする



設計 — 実現する方法をきめる

あれをこうして、
次にこれをして、
最後にあれを
こうする。

実装 — プログラムを書く
(前半はSqueak 後半はJava)



テスト — プログラムが意
図通り出来上がったか確
かめる(必要があれば前
の工程に戻る)