

# 情報とコンピュータ

静岡大学工学部  
安藤和敏

2005.10.17

## 1章 プログラミング入門 決定木のコーディング

- ・耳寄りな話
- ・決定木
- ・プログラミングを始めるには
- ・データの読み取りと記憶
- ・決定木のプログラミング

## 文の意味と構造

文

```
writeln(' Great Ideas ');
```

= `writeln("");` + `Great Ideas`

構文

データ

データはコンピュータの中のメモリに記憶される。

プログラムによって操作される情報

## メモリ

番地 (メモリ内の記憶場所)

データ

0	175
1	
2	安藤
3	
4	浜松
5	
6	2004
7	

⋮

## 変数 (variable)

番地 (メモリ内の記憶場所)

データ

0	175
1	
<b>position1</b>	安藤
3	
4	浜松
5	
6	2004
7	

position1 にデータを記憶したり, position1にあるデータを取り出したりできる。

## Pascal における変数の宣言

```
var  
  変数名: 変数の型;
```

```
例:  
var  
  position1: string;
```

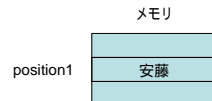
## readLn (または, readLn)

```
readLn(position1);
```

1. 入力される文字列を受け取る。(Enterキーで終わる.)
2. その文字列を position1 という名前の場所に入れる.

## プログラム例1 (ReadData)

```
program ReadData(input,output);  
var  
    position1: string;  
begin  
    readLn(position1);  
    writeLn(position1);  
end.
```



## プログラム例2 (ReadWriteDemo)

```
{ p.41 ReadWriteDemo }  
program ReadWriteDemo(input,output);  
var  
    position1: string;  
begin  
    writeLn('Please type in some data.');    readLn(position1);  
    writeLn('The data have been stored.');    writeLn('Next the data will be printed.');    writeLn(position1);  
    writeLn('This completes the run.');end.
```

## プログラム例3 (GetNameTown)

```
{ p.42 GetNameTown }  
program GetNameTown(input,output);  
var  
    PositionOfName, PositionOfTown: string;  
begin  
    writeLn('Hi, tell me your name.');    readLn(PositionOfName);  
    writeLn('What town do you live in?');    readLn(PositionOfTown);  
    writeLn('Can you tell me something?');    writeLn(PositionOfName);  
    writeLn('How do you like living in?');    writeLn(PositionOfTown);  
end.
```

## プログラム例3' (GetNameTownJ)

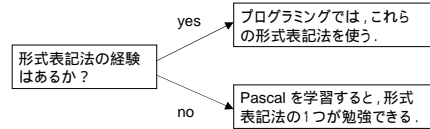
```
{ p.42 GetNameTownJ }  
program GetNameTownJ(input,output);  
var  
    PositionOfName, PositionOfTown: string;  
begin  
    writeLn('やあ, 君の名前を教えてよ. ');  
    readLn(PositionOfName);  
    writeLn('どこに住んでるの?');  
    readLn(PositionOfTown);  
    writeLn('何か話してくれる?');  
    writeLn(PositionOfName);  
    writeLn(PositionOfTown);  
    writeLn('ってどんな感じ?');  
end.
```

## 変数名

- 英字で始まって英字と数字だけを含んでいるもの.
- ただし, スペースを含んでいてはいけない.
- 名前として使ってはいけない単語 (予約語) もある. program, begin, end, など.

## LoadDemo

## 簡単な決定木



## If-then-else 構文

Pascal の if-then-else文 は例えば、以下のような書式を取る。

```
if answer = 'yes' then
  PascalコードA
else
  PascalコードB
```

「PascalコードA」と「PascalコードB」は、beginとendで挟まれた文の並びで、これらは、**複合文**と呼ばれる。

## 簡単な決定木のプログラム

```
program FirstTree(input, output);
var
  answer : string;
begin
  writeln('形式表記法の経験はあるか?');
  readln(answer);
  if answer='yes' then
    PascalコードA
  else
    PascalコードB
end.
```

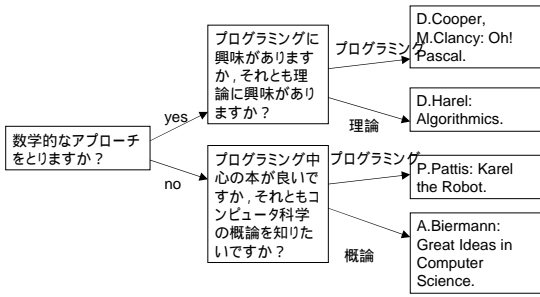
## いくつかの注意

- 質問には、yes か no で答えなければならない。
- 適当なインデント(字下げ)は、プログラムを読みやすくする。(文法的にはどうでもよい。)
- セミicolon (;) のつけ方:
  - ・ヘッダの終わりと宣言文の終わり。
  - ・(複合文中の)各文の終わり。

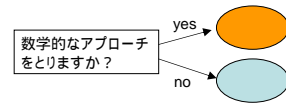
## セミicolonのつけ方

```
program FirstTree(input, output);
var
  answer : string;
{M} begin
  {M1} writeln('形式表記法の経験はあるか?'); {endM1}
  {M2} readln(answer); {endM2}
  {M3} if answer='はい' then
    {N} begin
    {N1} writeln('プログラミングではこれらの形式表記法を');
    {N2} writeln('実際に使う。');
    {N} end
  else
    begin
    writeln('Pascalを勉強すると、');
    writeln('表記法の1つが学習できる。');
    end; {endM3}
{M} end.
```

## 図書推薦の決定木



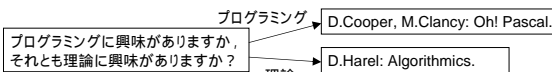
## BookAdviceのコード(1)



```

program BookAdvice(input, output);
var
  answer1, answer2, answer3 : string;
begin
  writeln('数学的アプローチをとりますか?');
  readLn(answer1);
  if answer1='yes' then
    yesの分岐を扱うコードをここに入れる
  else
    noの分岐を扱うコードをここに入れる
end.
  
```

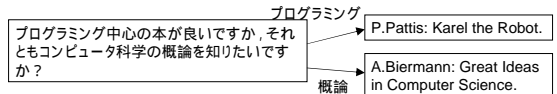
## BookAdviceのコード(2)



```

begin
  writeln('プログラミングに興味がありますか、');
  writeln('それとも理論に興味がありますか?');
  readLn(answer2);
  if answer2='programming' then
    begin
      writeln('D. Cooper and M. Clancy著');
      writeln('\"Oh, Pascal!\"を薦めます。');
    end
  else
    begin
      writeln('D. Harel著');
      writeln('\"Algorithmics\"を薦めます。');
    end;
end
  
```

## BookAdviceのコード(3)



```

begin
  writeln('プログラミング中心の本が良いですか、それ');
  writeln('ともコンピュータ科学の概念を知りたいです');
  readLn(answer3);
  if answer3='programming' then
    begin
      writeln('R. Pattis著');
      writeln('\"Karel the Robot\"がお薦めです。');
    end
  else
    begin
      writeln('A. Biermann著');
      writeln('\"Great Ideas in Computer Science\"がお薦めです。');
    end;
end;
  
```

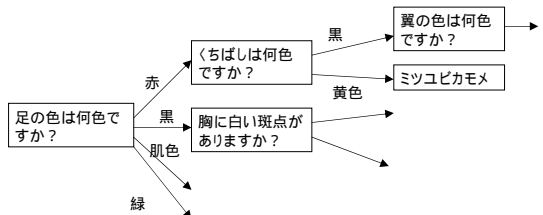
## If-then 構文

Pascal の if-then構文 は、以下のような書式を取る。

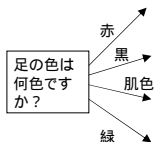
If 条件 then  
複合文;

条件が満たされれば、それに続く複合文が実行され、条件が満たされなければ、この複合文は実行されずに、次の文の実行に移る。

## カモメの分類の決定木



## SeagullClassのコード(1)



```
program SeagullClass(input, output);
var
  answer1: string;
begin
  writeln('足の色は何色ですか?');
  readln(answer1);
  if answer1='red' then
    '赤,の場合のコード;
  if answer1='black' then
    '黒,の場合のコード;
  if answer1='flesh' then
    '肌色,の場合のコード;
  if answer1='green' then
    '緑,の場合のコード;
end.
```

## SeagullClassのコード(2)



```
begin
  writeln('くちばしは何色ですか ');
  readln(answer2);
  if answer2 = 'black' then
    begin
      くちばしが黒い場合のコード
    end;
  if answer2 = 'yellow' then
    begin
      writeln('ミツユビカモメです. ');
    end;
end;
```

## 第1章のまとめ

- 決定木概念と決定木を用いた情報処理の方法.
- 決定木のプログラミングに必要な概念:
  - 変数
  - Pascalの入出力 (readLn, WriteLn)
  - If-then-else 構文
  - If-then 構文