

1.4 油分け算プログラムの改良

問 1.3 のように、マス 1、マス 2 の容量、目標量をいろいろな値に変更して調べたいとき、その度にプログラムを書き換えるのは感心しません。値をプログラムの中で設定しないで、実行してから入力できるようにしましょう。

プログラムの実行部の初めの 3 行を書き換えます。

変更前

```
begin // 実行部
  Youryou1 := 25;
  Youryou2 := 17;
  Mokuhyou := 1;
```

変更後

```
begin // 実行部
  Write(' マス 1 の容量 マス 2 の容量 目標量 ? ');
  ReadLn(Youryou1, Youryou2, Mokuhyou);
```

1.5 解説

1.5.1 入力文

ReadLn(変数₁, ..., 変数_k)

キーボードから k 個の値を読んで、変数₁, ..., 変数_k に代入する。

入力は、“コンマ (,)” ではなく“空白” で区切る。最後に “Enter” を押す。“Enter” を押すまでは “Back space” で戻って修正できる。

2 簡単なプログラム

入出力と代入だけからなる簡単なプログラム Kantan を作りましょう。

2.1 新規プログラム

Delphi の画面の左上, “ファイル (F)” の下に白紙の右上を折り曲げたようなアイコン (小さい絵) をクリックすると, “新規作成” のダイアログが現れます。この中から “コンソールアプリケーション” を選びます。

次のような, 空のプログラムが現れます。

```
program Project2;
{$APPTYPE CONSOLE}
uses
  SysUtils;

begin
  { ここにプログラムコードを書いてください }
end.
```

[ファイル | プロジェクトに名前をつけて保存] をして, Z:\Programming1 の中に Kantan.dpr という名前で保存します。

次のように, 1 行目のプログラム名が Kantan に変わります。

```
program Kantan;
```

注 2.1 プログラム名の変更は必ず [ファイル | プロジェクトに名前を付けて保存] をして変更すること。直接書き換えてはいけません。

1 行目の後に作成者を書く習慣をつけましょう。

```
program Kantan; // 学生証番号 氏名
```

この段階で実行してみると, コンソール画面が一瞬現れるだけで, すぐ終了します。

Enter を押すまで終了しないように, 実行部に書き込みます。

```
begin
  Write('Enter を押してください');
  ReadLn;
end.
```

今度は実行するとコンソール画面が消えずに残っていて, Enter を押すと終了します。

ここまでが, 新しいプログラムを作るときにいつも行う手順です。もちろん, プログラム名はプログラムごとに異なります。

2.2 プログラム入力

次のようにプログラムを書き込んでください。

MtasuN などは変数名で 1 つの単語なので、間に空白を入れてはいけません。

終わったら [ファイル | 上書き保存] をして保存します。

```
program Kantan; // 学生証番号 氏名
{$APPTYPE CONSOLE}
uses
  SysUtils;

var
  M      : Integer; // 整数は Integer
  N      : Integer;
  MtasuN : Integer;
  MhikuN : Integer;
  MkakeruN : Integer;
  MwaruN  : Real;    // 実数 (小数部がある) は Real
  MdivN   : Integer;
  MmodN   : Integer;
  MdaiN   : Boolean; // 論理値 (真か偽) は Boolean

begin
  Write(' 2つの数 [m n] ? ');
  ReadLn(M, N);
  MtasuN := M+N;
  WriteLn(M, ' + ', N, ' = ', MtasuN);
  MhikuN := M-N;
  WriteLn(M, ' - ', N, ' = ', MhikuN);
  MkakeruN := M*N;
  WriteLn(M, ' * ', N, ' = ', MkakeruN);
  MwaruN := M/N;
  WriteLn(M, ' / ', N, ' = ', MwaruN);
  MdivN := M div N;
  WriteLn(M, ' div ', N, ' = ', MdivN);
  MmodN := M mod N;
  WriteLn(M, ' mod ', N, ' = ', MmodN);
  MdaiN := M>N;
  WriteLn(M, ' > ', N, ' = ', MdaiN);
  Write('Enter を押してください');
  ReadLn;
end.
```

2.3 実行

実行して 2 つの数として、1000 7 を入れるとどのように出力されるでしょうか。実行前に考えて、実行結果と比べてください。

M+N, M-N, M*N は思った通りだったでしょう。

M/N は複雑な出力ですね。これは $1.428571 \dots \times 10^2$ すなわち 142.8571... という意味です。

M div N, M mod N はそれぞれ M を N で割った商と余りです。この 2 つの演算はよく使うので早くおぼえてください。

M>N の値は TRUE(真)または FALSE(偽)です。このように、値は数値とは限りません。Aburawakezan には、値が文字列のもの (string) がありましたね。

2.4 出力の桁指定

出力の右辺の値を右端が揃うように、桁数を指定しましょう。

```
WriteLn(M, ' + ', N, ' = ', M+suN:6);
```

のように、Write 文の最後のデータの後に :6 をつけます。7 箇所もあるので、置換機能を使うと楽です。

変更したい一番上にある行にカーソルを移動して [検索 | 置換] をします。置換ダイアログが現れるので、検索文字列に) 置換文字列に :6) を入れて、「すべて置換」をします。)が見つかるたびに、「置き換えますか?」と訊かれるので、Yes か No を答えます。

実行 整数の値は右端が揃ってきれいですが、実数の M/N がはみ出ています。

次のように変更します。

```
WriteLn(M, ' / ', N, ' = ', M+suN:10:3);
```

実数の場合は、: 全体の桁数:小数部の桁数 のように指定します。:10:3 とすると、全体で 10 桁、小数部が 3 桁なので整数部が 6 桁になります (小数点が 1 桁あります)。

2.5 問題

次のプログラムを作りなさい。

2.5.1 等差数列の和 TousaSuuretu

入力 等差数列の、初項 Syokou、公差 Kousa、項数 Kousuu (すべて Integer)
出力 その和 Wa (Integer)

実行例

```
等差数列の和を求めます
初項 ? 1
公差 ? 2
項数 ? 100
和 = 10000
Enter を押してください
```

ヒント 等差数列の公式

$$\text{末項} = \text{初項} + \text{公差} (\text{項数} - 1)$$

$$\text{和} = \frac{\text{項数} (\text{初項} + \text{末項})}{2}$$

2.5.2 球の表面積と体積 Kyuu

入力 球の半径 Hankei (Real)

出力 その表面積 Menseki と体積 Taiseki (いずれも Real)

実行例

```
級の表面積と体積を求めます
半径 ? 50
表面積 = 31415.927
体積    = 523598.776
Enter を押してください
```

ヒント 球の公式

$$\begin{aligned} \text{表面積} &= 4 \cdot \text{円周率} \cdot \text{半径}^2 \\ \text{体積} &= \frac{\text{表面積} \cdot \text{半径}}{3} \end{aligned}$$

ヒント 円周率 π は Pi という定数に定義されています。

```
WriteLn(Pi:8:6)           3.141593
```

ヒント Pascal には n 乗を計算する演算子や関数はありません。ただし、2 乗を計算する関数 Sqr があります。

```
WriteLn(Sqr(10.5):8:2)    110.25
```

2.5.3 対称か Taisyou3keta

入力 3桁の自然数 Moto (Integer)

出力1 各桁の数字を逆順にした3桁の数 Gyaku (Integer)

出力2 入力した数が左右対称かどうか Taisyou (Boolean)

実行例

```
3桁の数が対称かどうか調べます
元の数 ? 475
逆にした数 = 574
対称である = FALSE
Enter を押してください
```

```
3桁の数が対称かどうか調べます
元の数 ? 626
逆にした数 = 626
対称である = TRUE
Enter を押してください
```

ヒント まず元の数 Moto の、百の位 Hyaku, 十の位 Juu, 一の位 Ichi を div と mod を使って求めます。百の位と一の位は簡単ですね。

Ichi, Juu, Hyaku を組み合わせて、逆の数 Gyaku を求めます。

2つの数 Moto と Gyaku が等しいときが対称です。