

プログラミング学習指導に関する実践研究Ⅲ

— プログラミング学習の指導と教育的効果 —

工業科 金城幸廣・工藤雄司・大平典男
茂木好和・深澤孝之・小高昌人

1. はじめに

総合学科としてのプログラミング指導は3年目を向かえ充実した内容となった。

本校紀要33・34号においてプログラミング作品紹介を主として本校工業科のプログラミング指導の報告を行った。

また、昨年度紀要35号においてはプログラミング作品概要紹介と生徒の評価や作品分析を中心として報告を行った。

今紀要においては、本校入学以来の生徒作品の紹介と関連させ、プログラミング指導と集大成としての作品紹介を通してプログラミング指導の教育的効果について報告を行いたい。

昨今の情報教育界においては、プログラミング指導はコンピューターリテラシーの中で一分野を形成している。プログラミング指導の教育的効果として以下のことが考えられる。

(1) 社会に於ける対応

社会におけるコンピュータの普及にともない、産業界に於けるパソコンの普及は益々増大している。その為にはソフト面での開発は益々必要性が増している。その観点から考えれば高等学校におけるプログラミング教育は重要な分野である。

(2) 教育の活性化

最近の教育情勢を考えると様々な観点から教育の活性化が叫ばれている。プログラミング教育を生徒のやる気向上の点からとらえると、問題解決への試行錯誤の過程できわめて有効である。

(3) 目標を持つことによる向上

研究紀要33～35号で紹介したようにプログラミングコンテストという作品制作を通して目的意識を持ち、目標達成へ向けて努力する環境を形成することは教育効果として大きい。

2. プログラミング指導の指導形態

(本校工業系における指導形態)

職業系列(機械技術科) 95年まで

1年:「情報技術基礎」

2年:「機械実習」

3年:「課題研究」

総合科学科(メカトロニクス系列)

1年次:「情報基礎」

2年次:「電子機械実習」

「プログラミング技術Ⅰ」

「プログラミング技術Ⅱ」

2.1 実践内容と方法

本研究は本校、総合科学科における「課題研究」の授業を中心としてプログラミングコンテストの作品制作を通して付随するプログラミング学習と教育的効果についての実践研究を行なったものである。

2.2 プログラミングコンテストと応募目的

最近における情報教育は益々急速になっており、さらに日進月歩のパソコンの発達やその普及において、高等学校においても情報技術教育の必要性は増大している。このような点から全国高校生プログラムコンテストは、高等学校に於けるクラブ活動や教科等で生徒が作成した作品を募り情報技術教育の普及拡大、発展を目標としている。

本校においては数年前からプログラミング学習の集大成として、全国高校生プログラムコンテストへの参加を行い、生徒のプログラミング技術向上を目標を設定した学習が可能となることを目指してプログラミング指導を行い成果を上げている。

2.3 プログラミング言語とテーマ

(1) プログラミング言語

プログラミング指導をプログラミング言語の観点からとらえると入門用としてインタプリタ形言語である、BASICが主流であると考えられる。しかし、社会における実用面からとらえるとコンパイラ形言語であるC言語等が中心に活用されている。

本校では従前の情報基礎においてBASICが行われていることに関連し、その延長としてプログラミングの技術を向上させるという観点から94年度までの作品がBASICが主流であった。その後C言語が同時に指導されることになり、プログラミングコンテスト作品制作においてもC言語を使用することになった。

(2) テーマの選定

今回のプログラミングコンテスト出展作品が「総合科学科」の「課題研究」における授業中における作品制作が主であった。従って課題研究の趣旨「これまでの学習成果を踏まえて、自分なりの興味や進路に応じた学習を深め、まとめていくという総合学科の学習の集大成」という観点から教師サイドより助言を与え今回は作品制作を生徒個人にテーマを選定させる方式とした。

3. 作品概要

以下に各テーマについて生徒の作成したプログラム作品、概要説明書、実行結果の一部、流れ図等を示す。

3. 1. 1 簡易BASIC

平成9年度(第18回)

全国高校生プログラミングコンテスト 優良賞

(作品の概要)

コンピュータ活用の一分野として多くの教育機関及び産業界等でプログラム言語、C言語は広く活用されている。本生徒は本校2年次におけるC言語の学習や社会の情勢、進路等に対応した観点からC言語における構造体の理解・応用を目標とすることで本作品をテーマとした。

3. 1. 2 プログラム概要説明書(生徒作品)

プログラム名 : 簡易BASIC

使用言語(行数) : C言語

使用機種 : PC-9821

[作成の目的]

BASICという一つの言語を作ることで、データの制御方法を理解でき、同時に簡単なアプリケーションソフトを作るための知識が養われると思い作成にあたりました。

3. 1. 3 処理の概要

入力された文字を判断して、プログラムの追加、消去を行う。また、直接制御することもできます。

[プログラムの特徴と特に工夫・努力した点]

- ・日本語でエラーを表示するので、使う人達にどこがどう違うかがわかってもらいやすくなっています。
- ・helpコマンドを作り、命令や命令の使い方がわからない場合に対応できるようになっています。
- ・一通の基本的なbasicコマンドを使えるようにしました。

[利用の効果]

純正のbasicより少ないですが、使える命令の機能は純正のbasicと一緒なので、当basicで使い方を覚えてしまえば、それがそのままbasicの知

識になります。

3. 1. 4 プログラム設計報告書

(1) 作品ができるまでの手順

- ・二人で必要な関数を分担して作り、個別にエラーのチェックをしていきそれを結合してからデータの受け渡しを確認していきしました。

(2) 苦心した点

- ・LINE文などの画面上のグラフィックス部分が機種やosに依存する部分だったので作成に苦労しました。
- ・if文が条件を判定した後にどのように実行させるか考え、作成するのに苦労しました。
- ・式の計算で()や*/など優先順位を考慮するのに苦労しました。

(3) 良くできた点

- ・helpを付けてこのプログラムで使える命令を簡単に説明したこと。
- ・if文やfor文などを実行中にifやforが出てきても処理できるようにしたこと。
- ・basic上で宣言した関数をリスト構造で管理できるようにしたこと。

(4) 作品の制作を通して学んだこと

- ・一つのプログラムを作り上げることの難しさ。
- ・C言語での基本的なプログラム作成方法が理解できた。
- ・basicの内部でどのような処理が行われているか理解できた。
- ・basicでどのようなプログラムが組めるか理解できた。

(5) 作品使用の効果例

- ・basicでの基本的なプログラム作成が行えるようになる。

変数表 I

使われた変数の中でも特に必要なものだけを説明していきます。

ほとんどの変数が、カウンタ、配列の添次、簡単な条件分岐のための変数ですが、それらの変数はすべて省略します。

・プログラムリスト

入力されたプログラムを記憶するために構造体を使っています。

構造体の内容は以下の通りです。

```
struct prog {           : 行番号を記録します。  
    int gyoku;           : 命令を記録します。
```

```

char lib[10];      : データを記憶します。
char res[100];    : プログラムリストの次の行
                  : を指します。
struct prog *next; : プログラムリストの最初の
                  : 行を常に指しています。
} *GFPRO;         : プログラムリストの最後の
                  : 行を常に指しています。
*GFPRO;          : プログラムリストの最後を
                  : 常に指しています。

```

プログラムリストのそれぞれに記憶される内容や制限についてはプログラムの書き方のページで詳しく説明します。

・変数リスト

BASIC上で宣言された変数を記憶するために構造体を使っています。

構造体の内容は以下の通りです。

```

struct hensu {      : 宣言したときの変数名を記
                  : 録します。
char  name[20];     : 宣言したときの変数名を記
                  : 録します。
char  moji[50] ;    : 変数の内容を文字型として
                  : 記憶します。
struct hensu *next; : 変数リストの次の行を指し
                  : ています。
} *GFPRO;           : 変数リストの最初の行を常
                  : に指しています。
*GFPRO;             : 変数リストの最後を常に指
                  : しています。

```

変数の内容は文字型で記憶されていますが、この内容を使う際には変数名で判断して整数型になおします。判断の仕方は\$（ドルマーク）がついているかどうかです。つまり、\$が名前に入っていたら10と打ち込んでも文字として登録されてしまい、計算には使えないということです。これはBASICでも同じことが言えます。

・LINE用の変数

```

・int xone,yone;    : 左上の座標を記憶します。
(外部変数)
・int xtwo,ytwo;    : 右下の座標を記憶します。
・int color;        : 色番号を記憶します。
・char bf[5];       : 線の書き方を指定します。

```

(四角形や塗りつぶし)

左上の座標だけ外部変数にしたのは線を書いた後、関数

が閉じる前に右下の座標を左上の座標に入れて左上の座標が省略されたときに備えるためです。つまり、左上の座標を省略すると右下の座標が入ることになります。

プログラムを書く方法

プログラムを書く方法としては作品プログラム上で書くか、`sedit`などのテキスト形式で保存されるエディタ上で書くかのどちらかです。（提出したフロッピーから起動すると、ハードディスクはドライブc:としてマウントされます。）エディタを使う場合は大抵の場合、アローキーが使えますが、書き方を忠実に守って書かなくてはロードしても実行中にエラーが多発します。また、エディタを使うと、ロードの際に入れ替えは行いますが更新は行いません。同じ行番号があると正常にロードされませんので注意して下さい。

プログラムの書き方

プログラムを書く際は必ず次ぎに示すように3つの文に分けて下さい。（分けるとはスペースを一文子入れることです。）

- 1 行番号 1～30000 までです。
 - 2 命令 `print` や`for` などです。
 - 3 データ `print` で使う"～"や`for`で使う`a=0 to step 1` などです。
- `for`や`if`などは文節ごとに区切って下さい。
`if a=0 then print "TUTIYA"`
`for a=0 to 10 step 1`
 英字のみで80～90文字、日本語を使うならその半分くらいに押さえて下さい。

以下の書き方はしないで下さい。

- ・スペースを一カ所に2個以上付ける。
`for a=0 to 10 step 1`
- ・必要以上に区切る
`for a=0 to 30 s tep 1`
- ・全く区切らない
`fora=0to30step1`

直接命令（`cls` や`list`など）の場合は、2と3の二つに分けて下さい。

※以下省略

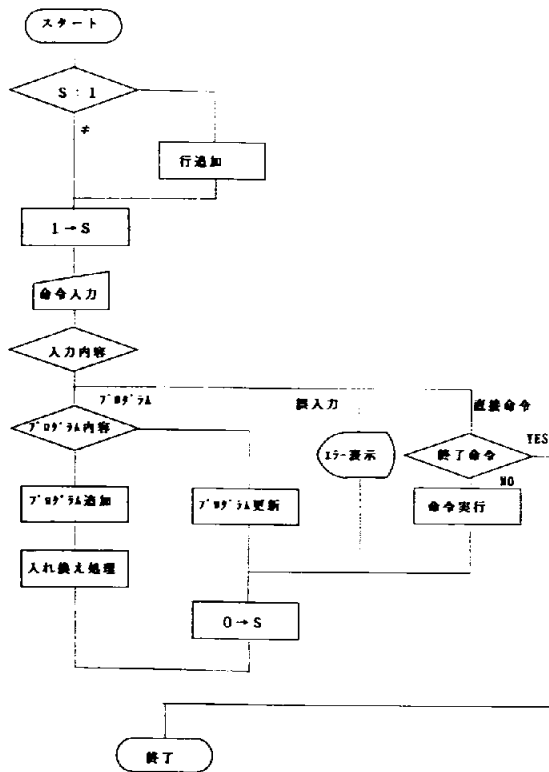


図1 1 流れ図

3. 2 実行結果

メニューを選択して下さい。
上下の矢印キーで選択、リターンキーで決定します。

CLS	画面の消去
LIST	プログラムの表示
DELETE	プログラムの削除
RENUM	行番号の整理
NEW	プログラムの削除
SAVE, LOAD	ファイルの読み込みと書き込み
COLOR	文字の色を変える
PRINT	データの出力
INPUT	データの入力
LET	代入計算
FOR	ループ命令
IF	分岐命令
GOTO	飛び越し命令
GOSUB	サブルーチンへの飛び越し命令
LINE	線を引く命令

HELPの終了

黄色で書かれた命令は直接命令です。入力された直後に実行されます。

赤で書かれた命令は行番号とともに書く命令紫で書かれた命令は両方で使えます。

図1 2 1 実行画面1 1

RUN と入力すれば実行できます。
SEdit と入力すればSEdit にいけます。

OK

```
10 FOR I=1 TO 10 STEP 1
```

```
20 T=T+1
```

```
30 NEXT T
```

```
40 PRINT T
```

```
50 END
```

RUN

エラー 行番号=30

FOR とNEXTで宣言した変数名が一致しません

```
30 NEXT I
```

LIST

```
10 FOR I=1 TO 10 STEP 1
```

```
20 T=T+1
```

```
30 NEXT T
```

```
40 PRINT T
```

```
50 END
```

RUN

55

図1 2 2 実行画面1 2

SYSTEMまたはQUITで終了します。

HELPと入力すると簡単な説明が受けられます

RUN と入力すれば実行できます。

SEdit と入力すればSEdit へ行けます。

OK

LOAD"PRO

OK

LIST 100-200

```
100 PRINT "バックをカラフルにしますか
```

```
110 PRINT "YorN"
```

```
120 INPUT A$
```

```
130 IF A$="Y" THEN GOSUB *BACK
```

```
130 IF A$="y" THEN GOSUB *BACK
```

```
150 END
```

```
160 *BACK
```

```
170 CLS 3
```

```
180 X1=0
```

```
190 X2=80
```

```
200 Y1=0
```

図1 2 3 実行画面1 3

3. 1. 4 授賞の要因

C言語はプログラマの為のプログラム言語だと言われるように多くの特長の為に広く産業界において活用されている。C言語学習の応用として、すでに学習したBASIC言語が使用できるソフトを作成した作品であり、C言語実用化の学習として賞賛できる。

3. 2. 1 4サイクルエンジンシミュレーション

平成9年度(第18回)

全国高校生プログラミングコンテスト 優良賞

(作品の概要)

工業教科、「原動機」の中にエンジンの構造、機能がある。そこでパソコンで4サイクルエンジンをシミュレーションすることによりプログラミング学習の集大成とした作品である。

(生徒作品)

使用言語(行数): N88BASIC(1227)

使用機種: PC-9821

[作成の目的]

1. 授業で学んだ時は、理解するのが難しかった”4サイクルエンジン”の動きを、更に深く勉強しようと思いました。
2. 今まで勉強してきた”N88BASIC”の締めくくりとして、何か大きなプログラムを作りたいと思いました。

3. 1. 4 処理の概要

- (1) はじめに、プログラム名、学校名、作成者名を表示し、バックでエンジンのピストンを動かすことによりアピールしました。
- (2) 3つのファイルを共に印刷出来るようにしました。
- (3) 4サイクルエンジンの動きのシミュレーションの所では止め進め印刷することが出来ます。※詳細は別紙

[プログラムの特徴と特に工夫・努力した点]

特徴: リターンキーで一つ一つ進められるようにしたので、エンジンについて分からない人でも4サイクルの仕組みを理解することができると思います。

[工夫・努力]

- (1) クランクの回転(6つの点を与えて円弧を書けるようにした。)
- (2) 消すとき(繰り返して黒で書いて消した)
- (3) 弁の動き(クランクの回転運動を変数にした)
- (4) ピストンの上下運動(クランクの回転と角度をもとにして動かした)

[利用の効果]

教科書等では、図でしか書かれてませんが、このプログラムでは、実際に自分で動かしながら一つ一つを見る事が出来、また印刷も出来るので、授業等で用いればエンジンの動きを深く理解出来ると思います。

3. 2. 2 プログラム設計報告書

(1) プログラム作成の目的

1. 現在広く利用されている自動車には、”4サイクルエンジンが使われています。学校の授業でもこの自動車エンジンの”4サイクル”については学習しましたが、教科書等には図でしか載せられていなかった為に、実際の動きがわかりにくく、深く理解できませんでした。

それで、この”4サイクル”について、更に深く理解しようと思いました。そこで、「4サイクルエンジンのシミュレーション」を作りたいと思い作成しました。

2. 高校1年、2年で、プログラム言語として「N88BASIC」の勉強と締めくくりとして、何か大きなプログラムを自分で作成したいと思い作成しました。

(2) 処理の概要(詳細)

1. はじめにプログラム名、学校名、作成者名を表示し、バックでエンジンのピストンを動かす事により、アピールしました。

☆処理の概要(詳細)

1. はじめにプログラム名、学校名、作成者名を表示し、バックでエンジンのピストンを動かす事により、アピールしました。

2. メニューでは、矢印キーで選択できるようにしました。またビープ音も出して選択し易いようにしました。

3. 3つのファイルを共に印刷出来るようにしました。特に「4サイクルエンジンの動きのシミュレーション」の所では、リターンキーが押される度に動くようにプログラムしたので、利用者の好きところで、止めたり、進めたり、印刷したりする事が出来ます。

4. 「ピストンの行程・死点・行程容積」の所では、図だけでなく、用語の解説と計算式も表示できるようにし、またそれを印刷できるようにしました。

(3) 新しいアイディア

1. クランクの回転

クランクの図形は2つの円弧とそれを結ぶ線で成り立っています。普通円弧を描くには、中心の座標を与えて、それと共に角度を与えて円弧を描きますが、クランクの形のまま円弧も回転させるには、角度を与えて描くよりも、3つの点を与えてその点から中心を求め、アークサイン(\sin^{-1})を使って、点の距離から、角度を求める方法を利用できると考えました。3つの点を与え

て、一つの円弧を描くサブルーチンをアレンジして、6つの点を与えれば、ふたつの円弧を描けるようなサブルーチンを作成しました。円弧の端の二点の座標が分かるため、円弧を結ぶ線もスムーズに書くことができました。

これによって、クランクの形のまま、なおかつ円弧も共に回転できるようにしました。

2. paint命令について

各行程をペイントするとき、ただの色で塗るだけでなく、16進数で8×8ドットのタイリングパターンを指定して塗る事により、吸気、圧縮、膨張、排気行程のそれぞれに合わせた模様で塗るようにしました。

3. 消去する時

一般的にc l s命令で消去するのが使われますが、このプログラムでは、一回描いたものを重ねて”黒”で描く事によって図形を消去してまた次の図形が描けるようにしました。この命令を使う事によって、動かない（消さなくても良い）部分は消さず、消す必要のあるところを消す事が出来るので、無駄な処理を削減する事が出来ました。

4. クランクの回転と、ピストンの上下運動の関係

クランクが回転するときの角度から、ピストンが動く範囲を何等分すれば良いかが分かります。（クランクが180度回転すると、ピストンは上から下へ、また下から上へ1行程分移動するので、180度を何回で回転するかを求めそれを元にピストンの上下運動を指定する事が出来ます。）

(4) 苦心した点

1. 円弧と直線によって成り立っているクランクを、中心の点を中心に回転させるのに苦労しました。

2. クランクにおいて、3つの点を与えて円弧を描くサブルーチンを、6つの点を与えることによって2つの円弧を同時に描いてしまうサブルーチンにアレンジするのに時間がかかりました。どの関数が何を表しているのかを一つ一つ分析し、6つの点を与えられれば2つの円弧を描くようにプログラムするのは本当に大変でした。

3. クランクを描くとき、6つの点を与えて、それを元に円弧を描きますが、円弧を描くためには、それぞれの点の角度を求めなくてはなりません。このプログラムでは、アークサインを求めて円弧を描くようにしました。しかし、N88 BASICには、アークサインを求める関数がありません。それでアークタンジェントからアークサインを求められるように逆三角関数を変えなければなりませんでした。その関数を理解するのが難しかった

です。

4. クランクの回転とピストンの上下運動の連動についてですが、回転している座標がどの位置にあるかを元にして、ピストンを上に上げたり下に下げたりする条件の指定（IF-THEN 文）に苦労しました。

ピストンが上へ上がらなければならないときに、下へ下がってしまうというような問題がたびたび出てきて、条件文を作成するには本当に苦労しました。

5. クランクの回転に合わせて弁を移動させる事と共に、4行程をグラフィックスで表すタイミングを指定するのに苦労しました。

(5) 良くできた点

1. クランクの回転

クランクの回転が出来なければ、この「4サイクルエンジンのシミュレーション」は恐らく完成しなかったと思います。このプログラムでは、6つの点を与え、それを元に円弧を描き、線で結ぶことができるようなサブルーチンを作りました。そのおかげで、クランクの形のまま回転させることができました。これは本当に良くやったと思っています。

2. クランクの回転とピストンの上下運動の関係

クランクが何度ずつ回転していくかを元に、180度の中で、何回回転移動するかを求めました。その値でピストンの上下の距離（1行程の距離）を割って、ピストンをいくつずつ動かせば良いか変数を使って求めることができるようにしました。

3. 弁の動き、各行程のグラフィックス表示

クランクの回転に合わせて、弁の動くタイミングと各行程のグラフィックス表示するために、変数”B”を使って、リターンが押される度に1ずつ足していき、クランクが2回転したところでその変数”B”を元に戻す方法をとった事も非常に良くできたと思います。

4. 「ピストンの行程・死点・行程容積」における用語の解説

解説の図と用語の解説を”S”と”R”キーで交互に表示できるようにしたので、利用者が好きなように図と解説を見比べる事が出来ます。

(6) 作品使用の効果例

1. 4サイクルエンジンの動きを自分で動かしながら深く理解する事が出来ます。

2. 用語の解説やエンジンの各部の名称も入れたので、動きだけでなく、用語や計算式なども理解できると思います。

3. 自分で動かしながら、また好きなところで印刷でき

るので、授業等で用いれば4サイクルエンジンについて深く理解できると思います。

・変数表1

－ 8 －

TITLE. BAS

・変数表2

－ 9 －

MENU. BAS

・変数表3

ENZNAME. BAS

・変数表4

KOUTEI. BAS

・変数表5

－ 10 －

ENZSIM. BAS

・流れ図1

－ 12 －

TITLE. BAS

・流れ図2

－ 13 －

MENU. BAS

・流れ図3

－ 14 －

ENZNAME. BAS

・流れ図4

－ 15 －

KOUTEI. BAS

・流れ図5

－ 16 －

ENZSIM. BAS

変数表1 (TITLE. BAS)

P 円周率

R クランクを回転させるための角度

KC クランクの色

C クランク以外の色

H クランクの回転を元にピストンを上下する

L ピストンの上下の幅を何等分して上下運動すれば良いか

X (1) - X (9) . . . X軸の座標

Y (1) - Y (9) . . . Y軸の座標

AA - AH、BA - BH・円の中心を求めるための途中式の結果

AX0、AY0 円弧Aの中心

BX0、BY0 円弧Bの中心

AR、BR 円弧A、Bの半径

AR1 - AR3 円弧Aの各点の角度

BR1 - BR3 円弧Bの各点の角度

X、Y 回転のサブルーチンで計算するために座標を代入する

XX、YY 回転させて得た座標

AX、BX アークサインを求めるためにX座標を代入するもの

AZ、BZ アークサインを求めるための角度

AXFL、BXFL . . . 求めている点が何象限かで補正するための変数

AS、BS 補正するための変数

MC ”何かキーを押してください。”の色

A\$ 入力待ち

I、E、S ループ用

変数表2 (MENU. BAS)

MKEY 選択している選択肢がどこかを指定する

MCA ”エンジン各部の名称”の色

MCB ”ピストンの行程・死点・行程容積”の色

MCC ”4サイクルエンジンの動きのシミュレーション”の色

MCD ”プログラムの終了”の色

M\$ キー入力待ち

MK\$ メニューより選ばれたファイルを呼び出すため

S\$ ”何かキーを押してください”の時のキー入力待ち

E、S ループ用

変数表3 (ENZNAME. BAS)

RD ラジアンに変換

A\$ キー入力待ち

E、S ループ用

変数表4 (KOUTEI. BAS)

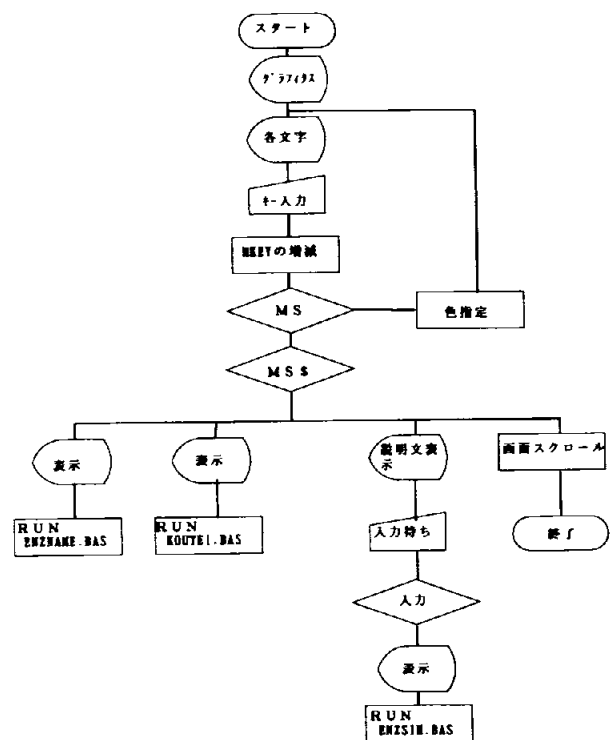
C シリンドルの色

PC ピストンと弁の色

KC クランクとピストンの色

RD ラジアンに変換
 A \$ キー入力待ち
 P \$ 解説のページ番号
 E、S ループ用
 変数表 5 (ENZSIM. BAS)
 SP 回転するときの角度 (3 段階ある) 途中で変換するため
 SP \$ " 低 速 " . " 中 速 " . " 高 速 " の文字変数
 P 円周率
 R クランクを回転させるための角度 (SP を代入する)
 C グラフィックスの色
 H クランクの回転を元にピストンを上下する
 RC 180 度の中で何回移動するか
 B リターンキーが押された時に足していき、これによって、弁と、何行程にいるかを指定する
 L ピストンの上下の幅を何等分して上下運動すれば良いか
 X (1) - X (9) . . . X 軸の座標
 Y (1) - Y (9) . . . Y 軸の座標
 L B X (1) - L B Y (8) . 左の弁の座標
 R B X (1) - R B X (8) . 右の弁の座標
 A A - A H、B A - B H . 円の中心を求めるための途中式の結果
 A X 0、A Y 0 円弧 A の中心
 B X 0、B Y 0 円弧 B の中心
 A R、B R 円弧 A、B の半径
 A R 1 - A R 3 円弧 A の各点の角度
 B R 1 - B R 3 円弧 B の各点の角度
 X、Y 回転のサブルーチンで計算するために座標を代入する
 X X、Y Y 回転させて得た座標
 A X、B X アークサインを求めるために X 座標を代入するもの
 A Z、B Z アークサインを求めるための角度
 A X F L、B X F L . . . 求めている点が何象限かで補正するための変数
 A S、B S 補正するための変数
 M 弁の平行移動の X 軸

N 弁の平行移動の Y 軸
 Y A 吸気と排気に表示される矢印の始点
 T I L E \$ タイルパターン (P A I N T O) で利用
 A \$ 入力待ち
 R \$ 回転速度の変更の時のキー入力待ち
 E、S ループ用



流れ図 2 1



** MENU **

- ☆ エンジンの各部の名称
- ☆ ピストンの行程・死点・行程容積
- ☆ 4サイクルエンジンの動きのシミュレーション
- ☆ プログラムの終了

上・下の矢印キーで、どれかを選択し、リターンキーを押して、決定して下さい。

図 2 1 1 実行画面 2 1 1

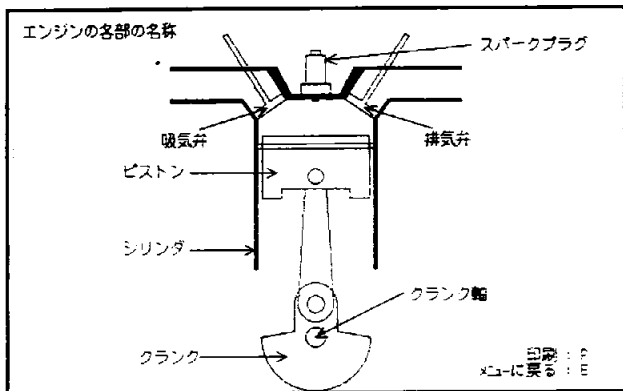


図 2 1 2 実行画面 2 1 2

このプログラムは、4サイクルガソリンエンジンの動きを示したもので、ピストンが2回の往復運動をする間に、混合気の吸い込みから、燃焼ガスの排出まで行って動力を発生することを示したものです。

なお、操作方法については画面の右下に表示されますので、それを見て操作して下さい。

何かキーを押して下さい。

図 2 1 6 実行画面 2 1 6

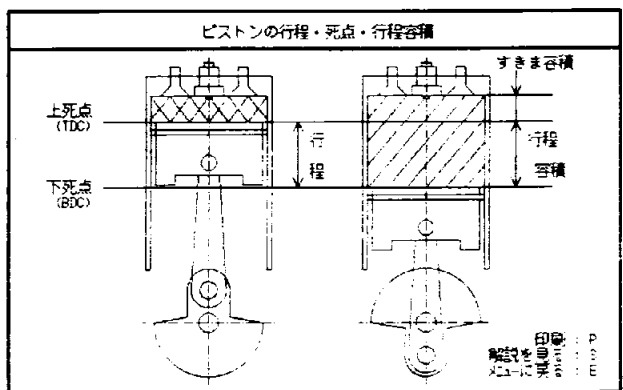


図 2 1 3 実行画面 2 1 3

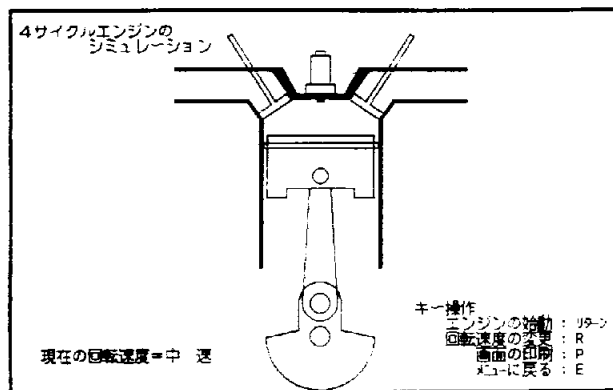


図 2 1 7 実行画面 2 1 7

用語の解説 & 計算式			Page.1
1)	上死点 (top dead center)	ピストンがクランク軸から最も離れた位置
2)	下死点 (bottom dead center)	ピストンがクランク軸に最も近い位置
3)	行程 (stroke)	上死点から下死点までの距離、あるいは上死点から下死点までの運動
4)	行程容積 (piston swept volume)	ピストンが1行程した時に押しける容積 ※この容積でエンジンの大きさをあらわすことがある
前頁: 上矢印 次頁: 下矢印 印刷: P 図に戻る: R			

図 2 1 4 実行画面 2 1 4

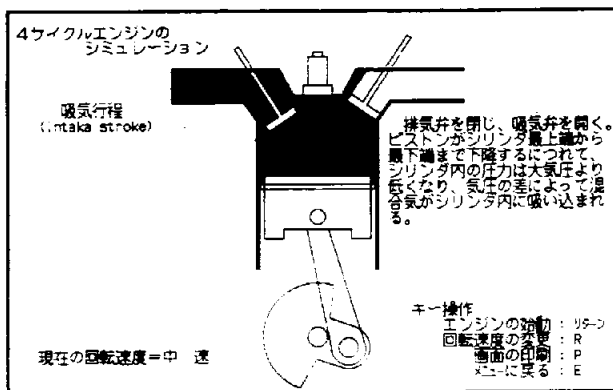


図 2 1 8 実行画面 2 1 8

用語の解説 & 計算式			Page.2
5)	すきま容積 (nominal clearance volume)	ピストンが上死点にある時、シリンダの頂部に残された空間(燃焼室)の容積
6)	シリンダ容積 (nominal cylinder volume)	すきま容積と行程容積の和、いかえるとピストンが下死点にある時のシリンダ内の容積 (すきま容積) + (行程容積) = (シリンダ容積)
7)	圧縮比 (compression ratio)	シリンダ容積とすきま容積との比で示す、混合気を圧縮する度合い。 圧縮比はエンジンの出力や燃費の経済性などにかかわりがあり、自動車用のガソリンエンジンでは、8-18くらいである。 (シリンダ容積) ÷ (すきま容積) = (圧縮比)
前頁: 上矢印 次頁: 下矢印 印刷: P 図に戻る: R			

図 2 1 5 実行画面 2 1 5

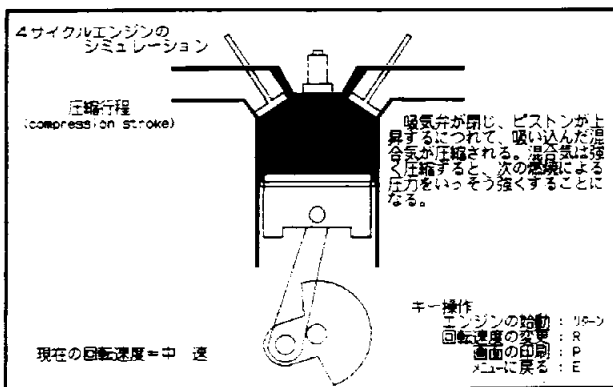


図 2 1 9 実行画面 2 1 9

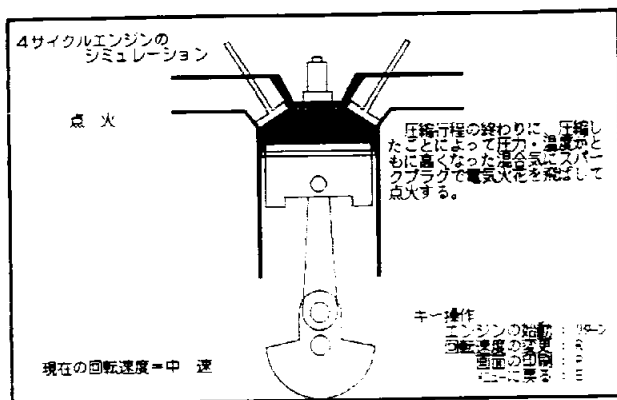


図2.2.0 実行画面 2.2.0

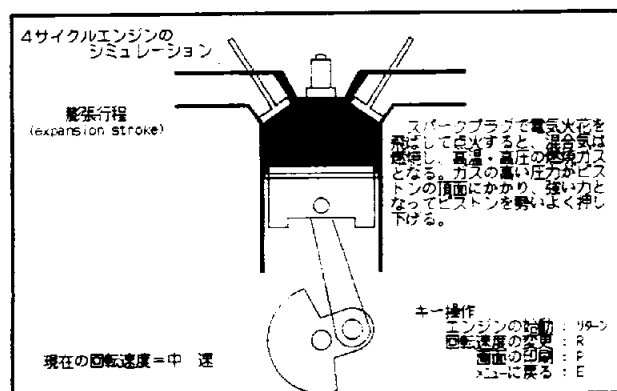


図2.2.1 実行画面 2.2.1

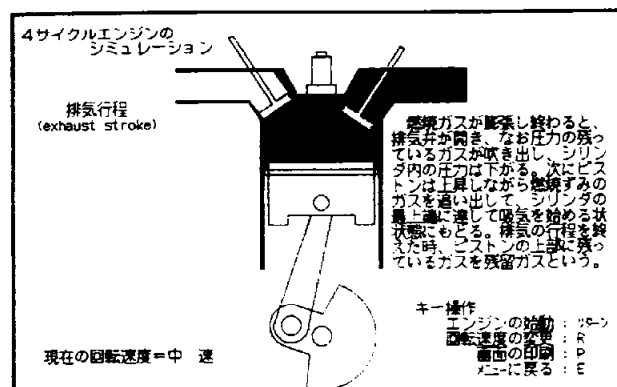


図2.2.2 実行画面 2.2.2

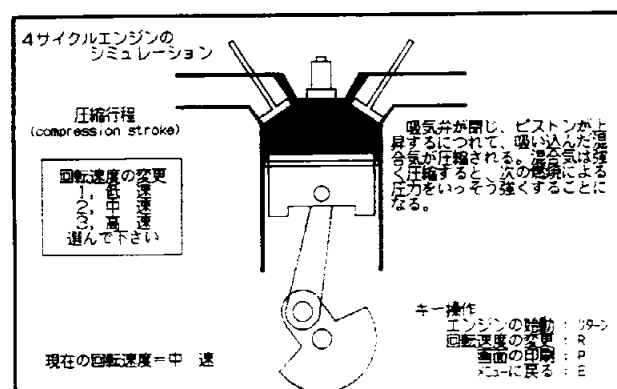


図2.2.3 実行画面 2.2.3

受賞の要因

エンジンの構造、機能の理解については教科書以外に模型等で広く指導が行われている。この作品は報告書にあるように日頃の教科学習の中で理解が難しい点と興味ある項目に着目し、プログラミング学習の応用として本作品を制作した。またクランクの動き等に苦心の後がある。

3. 3. 1 車椅子自動階段昇降機シミュレーション

平成9年度(第18回)

全国高校生プログラミングコンテスト 優良賞

(作品の概要)

日本の高齢化社会を迎えるあたり、福祉に目を向けることは大切である。本校では「産業社会と人間」の教科の中で障害者との体験という項目がある。この生徒はその体験から障害者を手助けしたいと考え工業の観点から本作品を制作した。

(生徒作品)

使用言語(行数): N88 BASIC (3735)

使用機種: PC-9821

プログラム名: 車椅子自動階段昇降機シミュレーション

これからの高齢化社会に対して、どのように対応していけばよいのか、また身体不自由者の事も考えこのシミュレーションプログラムの作成に取り組みました。

(処理の概要)

1. シミュレーションでは変更が可能
2. だいたいの所で印刷が可能
3. どこからでもほとんどMENU画面に戻れる

[作成の目的]

これからは高齢化社会に入るので、それなりにみんなに自覚してもらおうと思い、このプログラムの作成に取り組み、作り上げていきました。

[新しいアイデア]

ほとんどの画面において、画像と文字を分けての印刷を可能にしました。

メニュー画面を多くし、また簡単な操作で動作させる事ができるようにしました。

すぐにメニュー画面に戻れるようにしました。

動画の画像を分けて、立体的に見せました。

視点も変えて、色々な形で自動階段昇降機を見る事が出来るようにしました。

[良く出来た点]

グラフィックスの良さや動画、それにメニュー画面の構

成も、自分では良く出来たのではないかと思います。

グラフィクスなどは画面のちらつきを抑え、きれいに
見やすく出来ました。すべての操作がとても分かりやす
くなりました。インパクトのあるオープニングが出来ま
した。

カラフルで見やすいプログラムが出来たと思っていま
す。

[苦心した点]

このプログラムを作るにあたり、資料が何もないので
0から考えていかなければならず、苦心しました。

DATAがとても多いので、DATAの打ち直しや、
計算し直しにとても苦心しました。

プログラムに対して、純粹に取り組める時間が少な
かったので、仕上げるのが大変でした。

動画の座標を求めるのはむずかしく、とても苦心しま
した。

[反省点など]

このプログラムに対しての取り組みも遅れたため、
色々なアイデアがあるのに、それを生かすことが出来な
くてとても残念です。

またこれでプログラミングコンテストに応募するのは
最後なので大変残念です。

[作品使用の効果例]

実際にこのシュミュレーションされた機械が作動すれ
ば、中途半端な階段などに取り付ければとても使いやす
い物になると思います。また、身体障害者や年寄りなど
の苦労も少なくなると思われます。

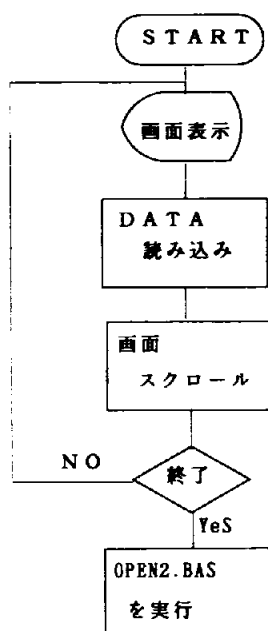


図3 流れ図

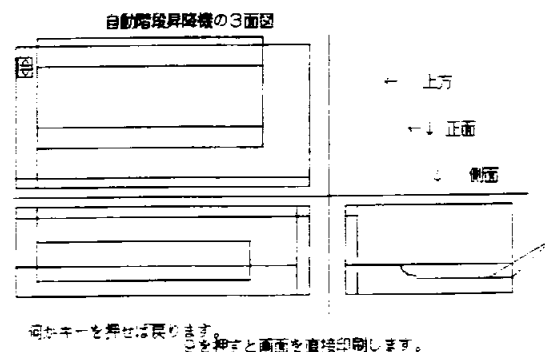


図3 1 1 実行画面 3 1

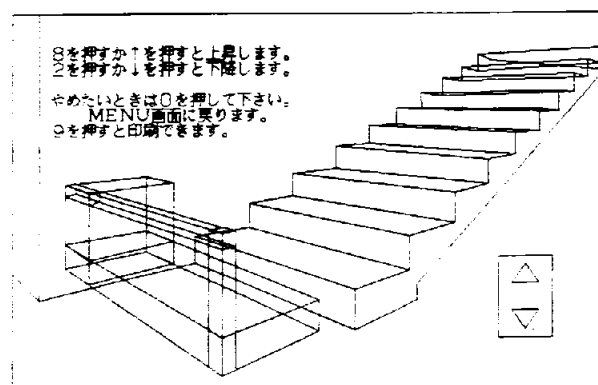


図3 1 2 実行画面 3 2

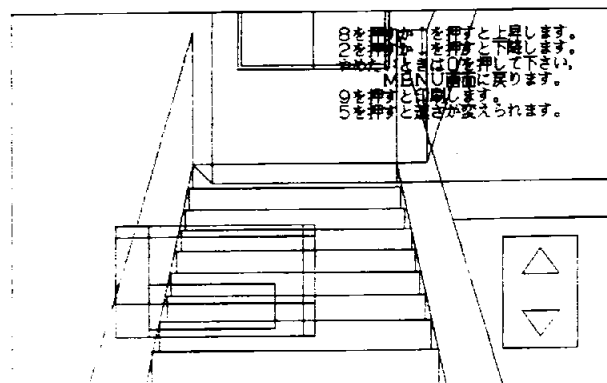


図3 1 3 実行画面 3 3

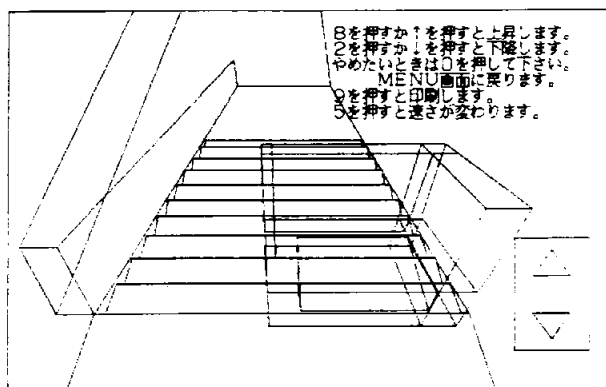


図3.14 実行画面 3.4

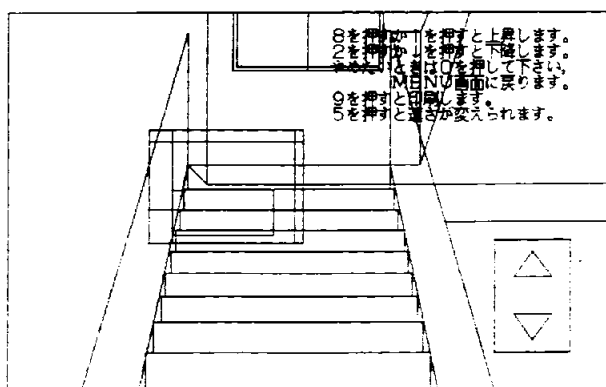


図3.15 実行画面 3.5

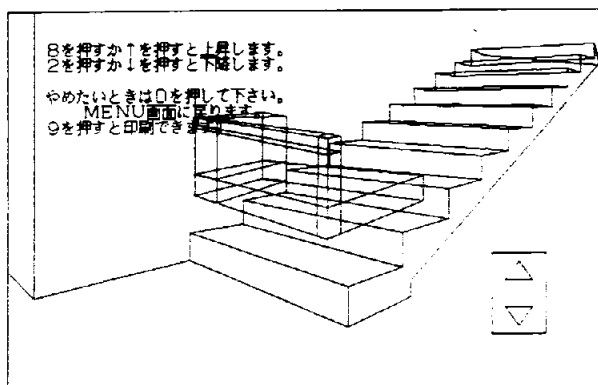


図3.16 実行画面 3.6

受賞の要因

この生徒は農業系列と商業系列の教科多く選択し、プログラミングにかなり関心があった。そこで発想が1系列にとらわれず、本テーマの選定となった。グラフィクスにかなり工夫の後がみられる。

4. 結果と考察

4.1 過去の入賞状況

以下に過去5年間の全国高校生プログラミングコンテストに出展した結果を示す。

93年(第15回)

1. 佳作 (平歯車の設計製図)

94年(第16回)

1. 優良賞(2段減速手巻きウインチの設計製図)

2. 佳作(溶接ロボットシミュレーション)

3. 学校優良賞(団体賞)

95年(第17回)

1. 優秀賞(NC旋盤簡易入力システム)

2. 優良賞(テーブルリフトのシミュレーション)

3. 佳作(自宅から学校までの時刻表)

4. 学校優秀賞(団体賞)

96年(第18回)

1. 優良賞 (1) 等角図作成プログラム

(2) 産業用ロボット簡易シミュレー

97年(第19回)

1. 優良賞 (1) 簡易BASIC

(2) 4サイクルエンジンシミュレーション

2. 佳作 車椅子自動階段昇降機シミュレーション

2. 学校優秀賞(団体賞)

4.2 作品の推移

95年までは職業科(機械科)の時の作品である。総合科学科となり96,97についても継続してプログラミング指導は実績を上げている。

従来、情報教育の中のプログラミング教育は工業高等学校における「情報科」等で中心に行われている。本校工業科は従前の「機械科」からの学科改編として「総合科学科」となった。工業系を専攻する多くの生徒が機械科あるいはメカトロニクス科という概念で入学を希望していた。そのため、かなり機械科の要素が色濃く残っている。

4.3 教育効果の観点

プログラミングコンテストに挑戦したほとんどの生徒が長期間にかけて作成することは初めての経験であった。

生徒の中には朝9時より午後8時まで昼食時以外はパソコンに向き合う生徒もいた。その思考過程の中で打ち込む姿は教育効果として大きいものである。

今回、生徒A(簡易BASIC作成)その後多くの業績を残した。

(1) 情報技術1級合格かつ特別賞授賞

(1) 情報技術1級合格かつ特別賞受賞

情報技術検定1級に合格することは本校では希であり、更に合格者の中で高い点数獲得者に与えられる特別賞を授賞することは本校初である。

(2) 通産省第2種情報処理技術者試験合格

「通産省技術者試験は情報処理技術者をを目指す者が目標とする国家試験である。受験者は情報処理の業務に携わる企業等の技術者、大学生、専門学校生、高校生等多岐にわたっている」高校生の合格はきわめて希であり、本校初めての合格者である。

生徒の感想（プログラミングコンテストに参加して）

生徒A

「プログラムを作るにあたっては、何度も問題にぶつかりました。たくさんの資料も調べ、夏休みも学校に来て作らなければならず、本当に苦労しました。しかしあきらめずに頑張って完成させた時は、今までの苦労も忘れるほどの達成感を味わうことができました。」

また生徒Bはテーマを持ったプログラムを製作することにより付随する問題の提起、問題分析、参考資料の調査を行い問題解決への達成感を経験した。

この生徒は対話の中で感じたことは賞を取ることが第1の目標ではなく、これまでのプログラミング学習の集大成として「課題研究」における作品制作を位置づけたことである。

生徒B

「夏休み中、毎日のように学校に来てプログラムを考えて、家に帰っても本を調べたりして大変だったが自分の好きな事だったので非常に充実した夏安みだった。

何か作るというのはとても大変なことがとても良く分かった。これから人の役に立つようなプログラムも作ってみたい。」

4. 4 入賞要因の分析

本校、工業系列は従前の「機械科」の影響が色濃く残っている。従って「情報科」、「電子科」と比較してプログラミングに費やす時間はかなり少ない。

その中で一定レベルの作品を保持するには以下の要因があると考えられる。

(1) 生徒は夏休み登校して作品製作にあたった。そこに相当の頑張りがあった。そこで作品が大きく向上した。

(2) テーマの選定

今回、生徒は「課題研究」の一分野としてプログラミングに取り組んだがあくまでも作品製作が学校の授業の延長線上にあると考え、できるだけ工業科教科から離れないことが望ましいと考えた。そこで作品制作がスムー

ズになった。

(3) 過去の結果の蓄積

作品製作にあたり多くの参考資料を必要とするものである。その中でも過去のプログラミングコンテスト出展作品は多くの教材よりも参考となる事が多い。生徒はこれらを参考にすることにより効率良く作品制作が可能となった。

5. 終わりに

本校の総合学科におけるプログラミング指導は今回の報告で3度目となり、充実したものとなり多く業績を伴うものであった。

また、プログラミング教育は従前の職業科における機械科、機械技術科で継続的に行われた流れを受け継ぐものである。

プログラミングコンテスト作品から分析すると機械科と総合科学科と比較して作品レベルにおいては大差はないと判断する。これは総合学科における工業系列の継続した内容の指導によるものである。

今後の課題として時代に対応した機種による指導方法の確立が必要とされる。

尚、この研究は、平成10年度科学研究費（奨励研究（B））による研究の一部をまとめたものである。

参 考 文 献

- * 筑波大学学校教育部紀要 第14巻（'93）
工業科における「課題研究」の実践研究
- * " 第15巻（'94）
情報技術教育の実践研究 I
- * 筑波大学附属坂戸高校研究紀要第27集（'88）
「課題実習」の実践的研究 I
- * " 第28集（'89）
II
- * " 第29集（'90）
III
- * " 第30集（'92）
機械科における「新設科目」の実践研究 I
- * " 第32集（'94）
「課題実習」の実践
プログラミングコンテスト作品指導とプログラム言語教育を中心として
- * プログラミング学習指導に関する実践研究 I
" 第33・34集（'96）
- * プログラミング学習指導に関する実践研究 I

```

1000 .....
1010 'タイトルプログラム
1020 '菅森義真
1030 'SAVE "TITLE.BAS"
1040 .....
1050 '
1060 STOP ON : ON STOP GOSUB *MENU
1070 CONSOLE 0,25,0,1 : SCREEN 3,0 : CLS 3
1080 P=3.14159 : R=-36 : KC=1 : C=1 : H=0
1090 L=68/(180/ABS (R)) '回転の角度とピストンの上下の間隔
1100 CLS 1
1110 '文字の表示
1120 LOCATE 22,5 : COLOR 4 : PRINT "4 サイクルエンジンのシミュレーション"
1130 LOCATE 22,7 : COLOR 4 : PRINT " Simulation of four-cycle engine "
1140 LOCATE 22,11 : COLOR 6 : PRINT " プログラミングコンテスト用作品 "
1150 LOCATE 22,12 : COLOR 6 : PRINT " 1997年度版 "
1160 LOCATE 22,15 : COLOR 5 : PRINT " 作成者 "
1170 LOCATE 22,16 : COLOR 5 : PRINT " 筑波大学附属坂戸高等学校 "
1180 LOCATE 22,17 : COLOR 5 : PRINT " 3年A組22番 菅森 義真 "
1190 '
1200 'クランクとピストンの最初の座標
1210 X(1)=376 : Y(1)=285
1220 X(2)=321 : Y(2)=340
1230 X(3)=264 : Y(3)=285
1240 X(4)=342 : Y(4)=249
1250 X(5)=321 : Y(5)=228
1260 X(6)=298 : Y(6)=249
1270 X(7)=298 : Y(7)=278
1280 X(8)=342 : Y(8)=278
1290 X(9)=320 : Y(9)=250
1300 '
1310 GOSUB *KURANKU
1320 GOSUB *PISTON
1330 FOR I=1 TO 7
1340 MC=INT(RND*7)+1
1350 LOCATE 46,22 : COLOR MC : PRINT "何かキーを押して下さい。"
1360 NEXT I
1370 A$=INKEY$
1380 IF A$="" THEN 1400
1390 IF A$=" " THEN GOTO *MENU ELSE GOTO *MENU
1400 C=0 : KC=0
1410 GOSUB *BACK
1420 GOTO 1330
1430 '
1440 *BACK
1450 GOSUB *KURANKU
1460 GOSUB *PISTON
1470 C=1 : KC=1
1480 FOR I=1 TO 9
1490 X=X(I)-320 : Y=284-Y(I)
1500 GOSUB *KAITEN 'サブルーチンへ
1510 X(I)=320+XX : Y(I)=284-YY
1520 NEXT I
1530 GOSUB *KURANKU 'サブルーチンへ
1540 IF Y(9)<250 THEN H=H-L 'X(9)の値と、Y(9)の値が
1550 IF X(9)>320 AND Y(9)<318 THEN H=H+L 'どこにあるかで、ピストンを
1560 IF Y(9)>318 THEN H=H+L '上下させるHの値の増減を決
1570 IF X(9)<320 AND Y(9)>250 THEN H=H-L 'める。
1580 GOSUB *PISTON 'サブルーチンへ
1590 RETURN
1600 '
1610 '
1620 'クランクを描くサブルーチン
1630 *KURANKU
1640 '
1650 '色の変数
1660 COLOR ...,KC

```

```

1670 '
1680 ' 円弧 A の中心を求める (AX0,AY0)
1690 AA=X(1)-X(2) : AB=Y(1)-Y(2) : AC=X(2)-X(3) : AD=Y(2)-Y(3) ' 円弧 A 中心
1700 AE=X(1)*X(1)-X(2)*X(2) : AF=Y(1)*Y(1)-Y(2)*Y(2)
1710 AG=X(2)*X(2)-X(3)*X(3) : AH=Y(2)*Y(2)-Y(3)*Y(3)
1720 AY0=((AE+AF)*AC-(AG+AH)*AA)/(AB*AC-AD*AA)/2
1730 AX0=(AE+AF-2*AB*AY0)/AA/2
1740 '
1750 ' 円弧 B の中心を求める (BX0,BY0)
1760 BA=X(4)-X(5) : BB=Y(4)-Y(5) : BC=X(5)-X(6) : BD=Y(5)-Y(6) ' 円弧 B 中心
1770 BE=X(4)*X(4)-X(5)*X(5) : BF=Y(4)*Y(4)-Y(5)*Y(5)
1780 BG=X(5)*X(5)-X(6)*X(6) : BH=Y(5)*Y(5)-Y(6)*Y(6)
1790 BY0=((BE+BF)*BC-(BG+BH)*BA)/(BB*BC-BD*BA)/2
1800 BX0=(BE+BF-2*BB*BY0)/BA/2
1810 '
1820 ' 円弧 A の半径と開始, 終了角を求める (AR,AR1,AR3)
1830 AR=SQR(ABS((X(3)-AX0)*(X(3)-AX0)+(Y(3)-AY0)*(Y(3)-AY0))) ' 半径 R を求める
1840 AX=X(3) : AZ=(AY0-Y(3))/AR : GOSUB *ARCSIN : AR3=AS
1850 AX=X(2) : AZ=(AY0-Y(2))/AR : GOSUB *ARCSIN : AR2=AS
1860 AX=X(1) : AZ=(AY0-Y(1))/AR : GOSUB *ARCSIN : AR1=AS
1870 IF AR3<AR2 AND AR2<AR1 THEN SWAP AR1,AR3 ' AR1,AR2,AR3 の位置関係
1880 IF AR2>AR3 AND AR3>AR1 THEN SWAP AR1,AR3 ' により AR1 と AR3 を交換する
1890 IF AR2<AR1 AND AR1<AR3 THEN SWAP AR1,AR3
1900 '
1910 ' 円弧 B の半径と開始, 終了角を求める (BR,BR1,BR3)
1920 BR=SQR(ABS((X(6)-BX0)*(X(6)-BX0)+(Y(6)-BY0)*(Y(6)-BY0)))
1930 BX=X(6) : BZ=(BY0-Y(6))/BR : GOSUB *ARCSIN : BR3=BS
1940 BX=X(5) : BZ=(BY0-Y(5))/BR : GOSUB *ARCSIN : BR2=BS
1950 BX=X(4) : BZ=(BY0-Y(4))/BR : GOSUB *ARCSIN : BR1=BS
1960 IF BR3<BR2 AND BR2<BR1 THEN SWAP BR1,BR3 ' BR1,BR2,BR3 の位置関係
1970 IF BR2>BR3 AND BR3>BR1 THEN SWAP BR1,BR3 ' により BR1 と AR3 を交換する
1980 IF BR2<BR1 AND BR1<BR3 THEN SWAP BR1,BR3
1990 '
2000 ' 円弧 A を描く
2010 CIRCLE (AX0,AY0),AR,,AR1,AR3
2020 '
2030 ' 円弧 B を描く
2040 CIRCLE (BX0,BY0),BR,,BR1,BR3
2050 '
2060 ' 円弧を結ぶ線
2070 LINE (X(1),Y(1))-(X(8),Y(8)),KC
2080 LINE-(X(4),Y(4)),KC
2090 LINE (X(3),Y(3))-(X(7),Y(7)),KC
2100 LINE-(X(6),Y(6)),KC
2110 '
2120 CIRCLE (320,285),10,, ' 主軸の円弧
2130 RETURN
2140 '
2150 '
2160 ' アークサインを求めるサブルーチン
2170 *ARCSIN
2180 '
2190 ' 円弧 A のアークサイン
2200 AS=ATN(AZ/SQR(ABS(1-AZ*AZ))) ' 円弧 A SIN^-1 (逆SIN) をアークタンジェントで求める
2210 AXFL=AX-AX0 ' 求めている点が第何象限
2220 IF AXFL>=0 AND AS>=0 THEN GOTO 2270 ' にあるかで補正する
2230 IF AXFL>=0 AND AS<0 THEN AS=AS+6.28318 : GOTO 2270
2240 AS=3.14159-AS
2250 '
2260 ' 円弧 B のアークサイン
2270 BS=ATN(BZ/SQR(ABS(1-BZ*BZ))) ' 円弧 B SIN^-1 (逆SIN) をアークタンジェントで求める
2280 BXFL=BX-BX0 ' 求めている点が第何象限
2290 IF BXFL>=0 AND BS>=0 THEN GOTO 2320 ' にあるかで補正する
2300 IF BXFL>=0 AND BS<0 THEN BS=BS+6.28318 : GOTO 2320
2310 BS=3.14159-BS
2320 RETURN
2330 '

```

```

2340 'ピストンを描くサブルーチン
2350 *PISTON
2360 '
2370 '連結棒
2380 CIRCLE (X(9),Y(9)),10,C
2390 CIRCLE (X(9),Y(9)),18,C,180/180*P,360/180*P
2400 LINE ((X(9))+18,(Y(9)))-(331,131+H),C '187+HでYの値を求める。
2410 LINE ((X(9))-18,(Y(9)))-(309,131+H),C
2420 '
2430 'ピストンの外形
2440 LINE (285,131+H)-(355,131+H),C
2450 LINE-(355,141+H),C
2460 LINE-(375,141+H),C
2470 LINE-(375,76+H),C
2480 LINE-(265,76+H),C
2490 LINE-(265,141+H),C
2500 LINE-(285,141+H),C
2510 LINE-(285,131+H),C
2520 LINE (260,89+H)-(380,84+H),C,B 'リングみぞ部の線
2530 CIRCLE (320,118+H),8,C '連結棒との連結部分
2540 RETURN
2550 '
2560 *KAITEN '回転した座標を求めるサブルーチン
2570 XX=X*COS(R/180*P)-Y*SIN(R/180*P)
2580 YY=X*SIN(R/180*P)+Y*COS(R/180*P)
2590 RETURN
2600 '
2610 *MENU
2620 CONSOLE 0,25,0,1 :COLOR 7
2630 LOCATE 0,24
2640 FOR E=1 TO 25
2650 ROLL 16
2660 PRINT CHR$(13)
2670 FOR S=0 TO 200 : NEXT S
2680 NEXT E
2690 RUN "MENU.BAS"

1000 .....
1010 'ピストンの行程・死点・行程容積
1020 '菅森義真
1030 'SAVE "KOUTEI.BAS"
1040 .....
1050 '
1060 STOP ON :ON STOP GOSUB *END.
1070 SCREEN 3,0 : CONSOLE 0,25,0,1 : CLS 3
1080 C=7 : PC=7 : KC=7 : MC=5 : RD=3.14159/180
1090 LINE (0,0)-(633,399),1,B
1100 LINE (2,2)-(631,40),1,B
1110 LINE (2,42)-(631,397),1,B
1120 LOCATE 25,1 : COLOR 5: PRINT "ピストンの行程・死点・行程容積" : COLOR 7
1130 LOCATE 62,21 : COLOR 6 : PRINT " 印刷 : P"
1140 LOCATE 62,22 : COLOR 6 : PRINT "解説を見る : S"
1150 LOCATE 62,23 : COLOR 6 : PRINT "メニューに戻る : E" : COLOR 7
1160 GOSUB *SEN
1170 X=208 : GOSUB *SIRNDA.GRA
1180 X=432 : GOSUB *SIRNDA.GRA
1190 GOSUB *PISTON.GRA
1200 GOSUB *SYASEN
1210 GOSUB *MOZI
1220 A$=INKEY$
1230 IF A$="P" OR A$="p" THEN COPY 3
1240 IF A$="S" OR A$="s" THEN GOTO *YOUGO
1250 IF A$="E" OR A$="e" THEN GOTO *END. ELSE 1220
1260 '
1270 *SEN
1280 '中心線
1290 LINE (208,50)-(208,390),7.,&HF99F

```



```

1300 LINE (432,50)-(432,390),7.,&HF99F
1310 LINE (140,325)-(276,325),7.,&HF99F
1320 LINE (364,325)-(500,325),7.,&HF99F
1330 '補助線
1340 LINE (100,117)-(562,117),2
1350 LINE (100,185)-(562,185),2
1360 LINE (492,90)-(562,90),2
1370 '矢印付きの線
1380 LINE (300,117)-(300,185),2
1390 LINE (295,127)-(300,117),2
1400 LINE-(305,127),2
1410 LINE (295,175)-(300,185),2
1420 LINE-(305,175),2
1430 '
1440 LINE (524,65)-(524,185),2
1450 LINE (519,80)-(524,90),2
1460 LINE-(529,80),2
1470 LINE (519,127)-(524,117),2
1480 LINE-(529,127),2
1490 LINE (519,175)-(524,185),2
1500 LINE-(529,175),2
1510 RETURN
1520 '
1530 *SIRNDA.GRA
1540 'シリンダそのもの
1550 LINE (X-60,270)-(X-65,270),C
1560 LINE-(X-65,70),C
1570 LINE-(X+65,70),C
1580 LINE-(X+65,270),C
1590 LINE-(X+60,270),C
1600 LINE-(X+60,90),C
1610 LINE-(X-60,90),C
1620 LINE-(X-60,270),C
1630 '
1640 'プラグ
1650 LINE (X-3,93)-(X+3,90),PC,B
1660 LINE (X-15,90)-(X+15,80),PC,B
1670 LINE (X-10,80)-(X+10,60),PC,B
1680 LINE (X-5,60)-(X+5,55),PC,B
1690 '
1700 '弁(左)
1710 LINE (X-50,90)-(X-20,90),PC
1720 LINE-(X-30,80),PC
1730 LINE-(X-30,60),PC
1740 LINE-(X-40,60),PC
1750 LINE-(X-40,80),PC
1760 LINE-(X-50,90),PC
1770 '弁(右)
1780 LINE (X+50,90)-(X+20,90),PC
1790 LINE-(X+30,80),PC
1800 LINE-(X+30,60),PC
1810 LINE-(X+40,60),PC
1820 LINE-(X+40,80),PC
1830 LINE-(X+50,90),PC
1840 RETURN
1850 '
1860 *PISTON.GRA
1870 'ピストンが上がったとき
1880 CIRCLE (208,325),10,KC
1890 CIRCLE (208,291),9,KC
1900 CIRCLE (208,159),8,KC
1910 'ピストンの外形
1920 LINE (173,172)-(243,172),KC
1930 LINE-(243,182),KC
1940 LINE-(263,182),KC
1950 LINE-(263,117),KC
1960 LINE-(153,117),KC
1970 LINE-(153,182),KC

```

'主軸
'クランク連結部
'ピストンと連結棒の交点

```

1980 LINE-(173,182),KC
1990 LINE-(173,172),KC
2000 '
2010 LINE (148,130)-(268,125),KC,B
2020 '
2030 ' 連結棒
2040 LINE (197,172)-(190,291),KC
2050 LINE (219,172)-(226,291),KC
2060 ' 円弧
2070 CIRCLE (208,291),18,KC,180*RD,360*RD
2080 ' クランクの作図
2090 CIRCLE (208,325),56,KC,180*RD,360*RD ' クランクの円弧
2100 CIRCLE (208,291),22,KC,0*RD,180*RD ' クランクの円弧 (連結部分)
2110 LINE (186,291)-(186,319),KC ' 円弧をつなげる線 (2150マテ)
2120 LINE-(152,325),KC
2130 LINE (230,291)-(230,319),KC
2140 LINE-(264,325),KC
2150 '
2160 ' ピストンが下がったとき
2170 RD=3.14159/180
2180 CIRCLE (432,325),10,KC ' 主軸連結部
2190 CIRCLE (432,359),9,KC ' クランク連結部の交点
2200 CIRCLE (432,227),8,KC ' ピストンと連結棒の交点
2210 ' ピストンの外形
2220 LINE (397,240)-(467,240),KC
2230 LINE-(467,250),KC
2240 LINE-(487,250),KC
2250 LINE-(487,185),KC
2260 LINE-(377,185),KC
2270 LINE-(377,250),KC
2280 LINE-(397,250),KC
2290 LINE-(397,240),KC
2300 '
2310 LINE (372,198)-(492,193),KC,B
2320 '
2330 ' 連結棒
2340 LINE (421,240)-(414,359),KC
2350 LINE (443,240)-(450,359),KC
2360 ' 円弧
2370 CIRCLE (432,359),18,KC,180*RD,360*RD
2380 '
2390 ' クランクの作図
2400 CIRCLE (432,325),56,KC,0*RD,180*RD ' クランクの円弧
2410 CIRCLE (432,359),22,KC,180*RD,360*RD ' クランクの円弧 (連結部分)
2420 LINE (454,359)-(454,331),KC ' 円弧をつなげる線 (2450マテ)
2430 LINE-(488,325),KC
2440 LINE (410,359)-(411,331),KC
2450 LINE-(376,325),KC
2460 RETURN
2470 '
2480 *MOZI
2490 LOCATE 5,6 : COLOR MC : PRINT "上死点"
2500 LOCATE 6,7 : COLOR MC : PRINT "(TDC)"
2510 LOCATE 5,11 : COLOR MC : PRINT "下死点"
2520 LOCATE 6,12 : COLOR MC : PRINT "(BDC)"
2530 LOCATE 39,8 : COLOR MC : PRINT "行"
2540 LOCATE 39,10 : COLOR MC : PRINT "程"
2550 LOCATE 67,8 : COLOR MC : PRINT "行程"
2560 LOCATE 68,10 : COLOR MC : PRINT "容積"
2570 LOCATE 63,3 : COLOR MC : PRINT "すきま容積"
2580 COLOR 7
2590 RETURN
2600 '
2610 *SYASEN
2620 ' 右
2630 LINE (392,90)-(372,114),5
2640 LINE (412,90)-(372,138),5
2650 LINE (432,90)-(372,162),5

```

```

2660 LINE (452,90)-(372,186),5
2670 LINE (472,90)-(392,185),5
2680 LINE (492,90)-(412,185),5
2690 LINE (492,114)-(432,185),5
2700 LINE (492,138)-(452,185),5
2710 LINE (492,162)-(472,185),5
2720 ' 左
2730 LINE (168,90)-(148,114),5
2740 LINE (188,90)-(168,117),5
2750 LINE (208,90)-(188,117),5
2760 LINE (228,90)-(208,117),5
2770 LINE (248,90)-(228,117),5
2780 LINE (268,90)-(248,117),5
2790 ' 左の重ね書き
2800 LINE (148,90)-(168,117),5
2810 LINE (168,90)-(188,117),5
2820 LINE (188,90)-(208,117),5
2830 LINE (208,90)-(228,117),5
2840 LINE (228,90)-(248,117),5
2850 LINE (248,90)-(268,114),5
2860 RETURN
2870 '
2880 *YOUGO                                ' 用語の解説 & 計算式
2890 CLS 3 : P$="1"
2900 CONSOLE 0,25,0,1 : SCREEN 3 : CLS 3
2910 LINE (0,0)-(639,399),1,B
2920 LINE (2,2)-(637,40),1,B
2930 LINE (2,42)-(637,357),1,B
2940 LINE (2,359)-(637,397),1,B
2950 LOCATE 28,1 : COLOR 5 : PRINT "用語の解説 & 計算式"
2960 GOSUB *PAGE.NOM
2970 LOCATE 4,23 : COLOR 6 : PRINT "    前頁 : 上矢印 ";
2980                                PRINT "    次頁 : 下矢印 ";
2990                                PRINT "    印刷 : P    ";
3000                                PRINT "    図に戻る : R"
3010 GOSUB * P.1
3020 A$=INKEY$
3030 IF A$=CHR$(&H1E) THEN GOSUB *P.1
3040 IF A$=CHR$(&H1F) THEN GOSUB *P.2
3050 IF A$="P" OR A$="p" THEN COPY 3
3060 IF A$="R" OR A$="r" THEN 1060
3070 GOTO 3020
3080 '
3090 *P.1
3100 CONSOLE 3,20,0,1 : COLOR 7 : CLS
3110 P$="1"
3120 GOSUB *PAGE.NOM
3130 LOCATE 4,5 : COLOR 7 : PRINT "1)          上死点          . . . ."
3140 LOCATE 4,6 : COLOR 7 : PRINT "          (top dead center)"
3150 LOCATE 39,5 : COLOR 7 : PRINT "ピストンがクランク軸から最も離れた位置"
3160 LOCATE 4,9 : COLOR 7 : PRINT "2)          下死点          . . . ."
3170 LOCATE 4,10 : COLOR 7 : PRINT "          (bottom dead center)"
3180 LOCATE 39,9 : COLOR 7 : PRINT "ピストンがクランク軸に最も近い位置"
3190 LOCATE 4,13 : COLOR 7 : PRINT "3)          行程          . . . ."
3200 LOCATE 4,14 : COLOR 7 : PRINT "          (stroke)"
3210 LOCATE 39,13 : COLOR 7 : PRINT "上死点から下死点までの距離、あるいは上"
3220 LOCATE 39,14 : COLOR 7 : PRINT "死点から下死点までの運動"
3230 LOCATE 4,17 : COLOR 7 : PRINT "4)          行程容積          . . . ."
3240 LOCATE 4,18 : COLOR 7 : PRINT "          (piston swept volume)"
3250 LOCATE 39,17 : COLOR 7 : PRINT "ピストンが1行程した時に押しのける容積"
3260 LOCATE 39,18 : COLOR 7 : PRINT "※この容積でエンジンの大きさをあら"
3270 LOCATE 39,19 : COLOR 7 : PRINT "わすことがある"
3280 RETURN
3290 '
3300 *P.2
3310 CONSOLE 3,20,0,1 : COLOR 7 : CLS
3320 P$="2"
3330 GOSUB *PAGE.NOM

```

```

3340 LOCATE 4,5 : COLOR 7 : PRINT "5) すきま容積 . . . ."
3350 LOCATE 4,6 : COLOR 7 : PRINT " (nominal clearance volume)"
3360 LOCATE 39,5 : COLOR 7 : PRINT "ピストンが上死点にある時、シリンダの頂"
3370 LOCATE 39,6 : COLOR 7 : PRINT "部に残された空間 (燃焼室) の容積"
3380 LOCATE 4,9 : COLOR 7 : PRINT "6) シリンダ容積 . . . ."
3390 LOCATE 4,10 : COLOR 7 : PRINT " (nominal cylinder volume)"
3400 LOCATE 39,9 : COLOR 7 : PRINT "すきま容積と行程容積の和、いいかえると"
3410 LOCATE 39,10 : COLOR 7 : PRINT "ピストンが下死点にある時のシリンダ内の"
3420 LOCATE 39,11 : COLOR 7 : PRINT "容積"
3430 LOCATE 39,12 : COLOR 3 : PRINT " (すきま容積) + (行程容積) ="
3440 LOCATE 39,13 : COLOR 3 : PRINT " (シリンダ容積)"
3450 LOCATE 4,15 : COLOR 7 : PRINT "7) 圧縮比 . . . ."
3460 LOCATE 4,16 : COLOR 7 : PRINT " (compression ratio)"
3470 LOCATE 39,15 : COLOR 7 : PRINT "シリンダ容積とすきま容積との比で示す、"
3480 LOCATE 39,16 : COLOR 7 : PRINT "混合気を圧縮する度合い。"
3490 LOCATE 39,17 : COLOR 7 : PRINT "圧縮比はエンジンの出力や燃料の経済性な"
3500 LOCATE 39,18 : COLOR 7 : PRINT "どとかかわりがあり、自動車用のガソリン"
3510 LOCATE 39,19 : COLOR 7 : PRINT "エンジンでは、8-10くらいである。"
3520 LOCATE 39,20 : COLOR 3 : PRINT " (シリンダ容積) ÷ (すきま容積) ="
3530 LOCATE 39,21 : COLOR 3 : PRINT " (圧縮比)"
3540 RETURN
3550 '
3560 *PAGE.NOM
3570 LOCATE 70,1 : COLOR 5 : PRINT "Page.":P$
3580 RETURN
3590 '
3600 *END.
3610 CONSOLE 0,25,0,1
3620 LOCATE 0,24
3630 FOR E=1 TO 25
3640 ROLL 16
3650 PRINT CHR$(13)
3660 FOR S=0 TO 200 : NEXT S
3670 NEXT E
3680 CLS 3
3690 RUN "MENU.BAS"

```

```

1000 .....
1010 'エンジンの各部の名称
1020 '菅森義真
1030 'SAVE "ENZNAME.BAS"
1040 .....
1050 '
1060 STOP ON : ON STOP GOSUB *END.
1070 CONSOLE 0,25,0,1 : SCREEN 3,0 : CLS 3
1080 RD=3.14159/180
1090 LINE (0,0)-(639,399),1,B
1100 LINE (2,2)-(637,397),1,B
1110 LOCATE 2,1 : COLOR 5 : PRINT "エンジンの各部の名称"
1120 LOCATE 67,22 : COLOR 6 : PRINT "印刷 : P"
1130 LOCATE 61,23 : COLOR 6 : PRINT "メニューに戻る : E" : COLOR 7
1140 GOSUB *SIRNDA.GRA
1150 GOSUB *PISTON.GRA
1160 GOSUB *YA
1170 GOSUB *MOZI
1180 A$=INKEY$
1190 IF A$="P" OR A$="p" THEN COPY 3
1200 IF A$="E" OR A$="e" THEN GOTO *END. ELSE GOTO 1180
1210 '
1220 *SIRNDA.GRA
1230 'シリンダ部の作図
1240 '左側シリンダの壁
1250 LINE (260,270)-(257,115),1,BF
1260 LINE (170,95)-(245,98),1,BF
1270 LINE (260,115)-(245,95),1
1280 LINE (257,115)-(242,95),1
1290 PAINT (258,114),1,1

```

1300 '
 1310 '右側シリンダの壁
 1320 LINE (380,270)-(383,115),1,BF
 1330 LINE (470,95)-(395,98),1,BF
 1340 LINE (380,115)-(395,95),1
 1350 LINE (383,115)-(398,95),1
 1360 PAINT (382,114),1,1
 1370 '
 1380 '吸気・排気部 & 弁
 1390 '
 1400 '左側
 1410 LINE (170,65)-(270,62),1,BF
 1420 '右側
 1430 LINE (470,65)-(370,62),1,BF
 1440 '
 1450 '弁 (左)
 1460 LINE (260,115)-(290,95),7
 1470 LINE-(285,89),7
 1480 LINE-(273,97),7
 1490 LINE-(228,27),7
 1500 LINE-(224,30),7
 1510 LINE-(269,100),7
 1520 LINE-(255,109),7
 1530 LINE-(260,115),7
 1540 '弁 (右)
 1550 LINE (380,115)-(350,95),7
 1560 LINE-(355,89),7
 1570 LINE-(367,97),7
 1580 LINE-(412,27),7
 1590 LINE-(416,30),7
 1600 LINE-(371,100),7
 1610 LINE-(385,109),7
 1620 LINE-(380,115),7
 1630 '
 1640 'プラグ部分 (下側)
 1650 LINE (270,65)-(290,95),1
 1660 LINE-(350,95),1
 1670 LINE-(370,65),1
 1680 '上側の線
 1690 LINE (370,62)-(360,62),1
 1700 LINE-(345,90),1
 1710 LINE-(295,90),1
 1720 LINE-(280,62),1
 1730 LINE-(270,62),1
 1740 '
 1750 'プラグの部分の塗りつぶし
 1760 PAINT (290,89),1
 1770 '
 1780 'プラグ
 1790 LINE (310,95)-(330,90),6,B
 1800 LINE (305,90)-(335,80),6,B
 1810 LINE (310,80)-(330,50),6,B
 1820 LINE (315,50)-(325,45),6,B
 1830 LINE (317,95)-(323,97),6,B
 1840 RETURN
 1850 '
 1860 *PISTON.GRA
 1870 CIRCLE (320,340),10,7
 1880 CIRCLE (320,306),9,7
 1890 CIRCLE (320,174),8,7
 1900 'ピストンの外形
 1910 LINE (285,187)-(355,187),7
 1920 LINE-(355,197),7
 1930 LINE-(375,197),7
 1940 LINE-(375,132),7
 1950 LINE-(265,132),7
 1960 LINE-(265,197),7
 1970 LINE-(285,197),7

'四角を先に書いて塗りつぶす

'主軸
 'クランク連結部
 'ピストンと連結棒の交点

```

1980 LINE-(285,187),7
1990 '
2000 LINE (260,145)-(380,140),7,B
2010 '
2020 '連結棒
2030 LINE (309,187)-(302,306),7
2040 LINE (331,187)-(338,306),7
2050 '
2060 '円弧
2070 CIRCLE (320,306),18,7,180*RD,360*RD
2080 '
2090 'クランクの作図
2100 CIRCLE (320,340),56,7,180*RD,360*RD
2110 CIRCLE (320,306),22,7,0*RD,180*RD
2120 LINE (342,306)-(342,334),7
2130 LINE-(376,340),7
2140 LINE (298,306)-(298,334),7
2150 LINE-(264,340),7
2160 RETURN
2170 '
2180 *YA
2190 'クランクの矢印
2200 LINE (200,360)-(290,360),2
2210 LINE (280,355)-(290,360),2
2220 LINE-(280,365),2
2230 'クランク軸の矢印
2240 LINE (320,340)-(410,295),2
2250 LINE (328,330)-(320,340),2
2260 LINE-(332,340),2
2270 'シリンダの矢印
2280 LINE (190,250)-(257,250),2
2290 LINE (247,245)-(257,250),2
2300 LINE-(247,255),2
2310 '吸気弁の矢印
2320 LINE (190,170)-(280,170),2
2330 LINE (270,165)-(280,170),2
2340 LINE-(270,175),2
2350 '吸気弁の矢印
2360 LINE (270,104)-(200,125),2
2370 LINE (257,103)-(270,104),2
2380 LINE-(261,112),2
2390 '排気弁の矢印
2400 LINE (370,104)-(440,125),2
2410 LINE (383,103)-(370,104),2
2420 LINE-(379,112),2
2430 'スパークプラグの矢印
2440 LINE (330,55)-(480,40),2
2450 LINE (341,49)-(330,55),2
2460 LINE-(342,59),2
2470 RETURN
2480 '
2490 *MOZI
2500 LOCATE 20,8 : COLOR 3 : PRINT "吸気弁"
2510 LOCATE 15,10 : COLOR 3 : PRINT "ピストン"
2520 LOCATE 15,15 : COLOR 3 : PRINT "シリンダ"
2530 LOCATE 17,22 : COLOR 3 : PRINT "クランク"
2540 LOCATE 52,18 : COLOR 3 : PRINT "クランク軸"
2550 LOCATE 54,8 : COLOR 3 : PRINT "排気弁"
2560 LOCATE 61,2 : COLOR 3 : PRINT "スパークプラグ"
2570 RETURN
2580 '
2590 *END.
2600 LOCATE 0,24
2610 FOR E=1 TO 25
2620 ROLL 16
2630 PRINT CHR$(13)
2640 FOR S=0 TO 200 : NEXT S
2650 NEXT E

```

'クランクの円弧
 'クランクの円弧 (連結部分)
 '円弧をつなげる線 (1360マテ)

```

2660 CLS 3
2670 RUN "MENU.BAS"

```

```

1000 .....
1010 ' 4 サイクルエンジンのシミュレーション (自動)
1020 ' 菅森義真
1030 ' SAVE "AUTOSIM.BAS"
1040 .....
1050 '
1060 '
1070 ' まず動かない部分を先に描く
1080 STOP ON : ON STOP GOSUB *END.
1090 SP=36
1100 SCREEN 3,0 : CONSOLE 0,25,0,1 : CLS 3
1110 GOSUB *SIRNDA
1120 '
1130 CLS
1140 LINE (0,0)-(633,399),1,B
1150 LINE (2,2)-(631,397),1,B
1160 LOCATE 1,1 : COLOR 4 : PRINT " 4 サイクルエンジンの "
1170 LOCATE 1,2 : COLOR 4 : PRINT " シミュレーション"
1180 LOCATE 52,1 : COLOR 4 : PRINT " (自 動)"
1190 LOCATE 52,19 : COLOR 2 : PRINT " キー操作"
1200 LOCATE 52,20 : COLOR 6 : PRINT " エンジンの始動 : リターン"
1210 LOCATE 52,21 : COLOR 6 : PRINT " ストップ : S "
1220 LOCATE 52,22 : COLOR 6 : PRINT " 回転速度の変更 : R "
1230 LOCATE 52,23 : COLOR 6 : PRINT " メニューに戻る : E "
1240 IF SP=12 THEN SP$="低 速" : GOTO 1270
1250 IF SP=36 THEN SP$="中 速" : GOTO 1270
1260 IF SP=90 THEN SP$="高 速" : GOTO 1270
1270 LOCATE 4,22 : COLOR 4 : PRINT "現在の回転速度=";
1280 COLOR 2 : PRINT SP$
1290 COLOR 7
1300 '
1310 ' 次に動く部分を描いていく (クランク、ピストン、弁)
1320 CONSOLE 3,16,0,1
1330 P=3.14159 : R=-SP : C=7 : H=0 : RC=180/ABS(R) : B=0
1340 L=68/RC ' 回転の角度とピストンの上下の間隔
1350 ' クランクとピストンの座標
1360 X(1)=376 : Y(1)=341
1370 X(2)=321 : Y(2)=396
1380 X(3)=264 : Y(3)=341
1390 X(4)=342 : Y(4)=305
1400 X(5)=321 : Y(5)=284
1410 X(6)=298 : Y(6)=305
1420 X(7)=298 : Y(7)=334
1430 X(8)=342 : Y(8)=334
1440 X(9)=320 : Y(9)=306
1450 ' 左の弁の座標
1460 LBX(1)=260 : LBY(1)=115
1470 LBX(2)=290 : LBY(2)=95
1480 LBX(3)=285 : LBY(3)=89
1490 LBX(4)=273 : LBY(4)=97
1500 LBX(5)=228 : LBY(5)=27
1510 LBX(6)=224 : LBY(6)=30
1520 LBX(7)=269 : LBY(7)=100
1530 LBX(8)=255 : LBY(8)=109
1540 ' 右の弁の座標
1550 RBX(1)=380 : RBY(1)=115
1560 RBX(2)=350 : RBY(2)=95
1570 RBX(3)=355 : RBY(3)=89
1580 RBX(4)=367 : RBY(4)=97
1590 RBX(5)=412 : RBY(5)=27
1600 RBX(6)=416 : RBY(6)=30
1610 RBX(7)=371 : RBY(7)=100
1620 RBX(8)=385 : RBY(8)=109

```

```

1630 '
1640 GOSUB *KURANKU                                ' サブ ルーチンへ
1650 '
1660 GOSUB *PISTON                                ' サブ ルーチンへ
1670 GOSUB *BEN                                    ' サブ ルーチンへ
1680 AS=INKEY$                                     ' 始動するためのキー入力待ち
1690 GOSUB *KEY.HAN
1700 IF AS=CHR$(13) THEN 1710 ELSE 1680
1710 AS=INKEY$                                     ' 自動が始まってからのキー入力待ち
1720 GOSUB *KEY.HAN
1730 C=0
1740 IF B=0 THEN GOSUB *BEN                        : GOTO 1810
1750 IF B=>1 AND B<RC THEN : GOSUB *KYUKI.DEL      : GOTO 1810
1760 IF B>RC AND B<RC*2 THEN GOSUB *ASSYUKU.DEL   : GOTO 1810
1770 IF B=RC*2 THEN GOSUB *TENKA.DEL              : GOTO 1810
1780 IF B=>RC*2 AND B<RC*3 THEN GOSUB *BOUCHO.DEL : GOTO 1810
1790 IF B>RC*3 AND B<RC*4 THEN GOSUB *HAIKI.DEL   : GOTO 1810
1800 IF B=RC*4 THEN GOSUB *BEN                    : GOTO 1810
1810 GOSUB *KURANKU
1820 GOSUB *PISTON
1830 C=7
1840 LINE (170,65)-(270,62),1,BF.                ' 線をきれいにするため
1850 LINE (470,65)-(370,62),1,BF
1860 FOR I=1 TO 9
1870     X=X(I)-320 : Y=340-Y(I)
1880     GOSUB *KAITEN                                ' サブ ルーチンへ
1890     X(I)=320+XX : Y(I)=340-YY
1900 NEXT I
1910 GOSUB *KURANKU                                ' サブ ルーチンへ
1920 IF Y(9)<306 THEN H=H-L                        ' X(9)の値と、Y(9)の値が
1930 IF X(9)>320 AND Y(9)<374 THEN H=H+L          ' どこにあるかで、ピストンを
1940 IF Y(9)=>374 THEN H=H+L                      ' 上下させるHの値の増減を決
1950 IF X(9)<320 AND Y(9)>306 THEN H=H-L          ' める。
1960 GOSUB *PISTON                                ' サブ ルーチンへ
1970 B=B+1
1980 IF B>RC*4 THEN B=1                            ' Bの値によって、何行程かを指定する。
1990 IF B=1 THEN GOSUB *KYUKI.MZI
2000 IF B=>1 AND B<RC THEN GOSUB *KYUKI.KOU      :GOTO 2120
2010 IF B=RC+1 THEN GOSUB *ASSYUKU.MZI
2020 IF B>RC AND B<RC*2 THEN GOSUB *ASSYUKU.KOU  :GOTO 2120
2030 IF B=RC*2 THEN GOSUB *TENKA.MZI
2040 IF B=RC*2 THEN GOSUB *TENKA.GRA
2050 IF B=RC*2+1 THEN GOSUB *BOUCHO.MZI
2060 IF B=>RC*2 AND B<RC*3 THEN GOSUB *BOUCHO.KOU :GOTO 2120
2070 IF B=RC*3+1 THEN GOSUB *HAIKI.MZI
2080 IF B>RC*3 AND B<RC*4 THEN GOSUB *HAIKI.KOU  :GOTO 2120
2090 IF B=RC*4 THEN GOSUB *BEN
2100 LINE (320-10,115-20)-(320+10,115-25),6,B    ' プラグの先端
2110 LINE (320-3,115-20)-(320+3,115-20+2),6,B
2120 GOTO 1710
2130 '
2140 '
2150 ' クランクを描くサブ ルーチン
2160 *KURANKU
2170 '
2180 ' 色の変数
2190 COLOR ...,C
2200 '
2210 ' 円弧 A の中心を求める (AX0,AY0)
2220 AA=X(1)-X(2) : AB=Y(1)-Y(2) : AC=X(2)-X(3) : AD=Y(2)-Y(3)  ' 円弧 A 中心
2230 AE=X(1)*X(1)-X(2)*X(2) : AF=Y(1)*Y(1)-Y(2)*Y(2)
2240 AG=X(2)*X(2)-X(3)*X(3) : AH=Y(2)*Y(2)-Y(3)*Y(3)
2250 AY0=((AE+AF)*AC-(AG+AH)*AA)/(AB*AC-AD*AA)/2
2260 AX0=(AE+AF-2*AB*AY0)/AA/2
2270 '
2280 ' 円弧 B の中心を求める (BX0,BY0)
2290 BA=X(4)-X(5) : BB=Y(4)-Y(5) : BC=X(5)-X(6) : BD=Y(5)-Y(6)  ' 円弧 B 中心
2300 BE=X(4)*X(4)-X(5)*X(5) : BF=Y(4)*Y(4)-Y(5)*Y(5)

```



```

2310 BG=X(5)*X(5)-X(6)*X(6) : BH=Y(5)*Y(5)-Y(6)*Y(6)
2320 BY0=((BE+BF)*BC-(BG+BH)*BA)/(BB*BC-BD*BA)/2
2330 BX0=(BE+BF-2*BB*BY0)/BA/2
2340 '
2350 '円弧 A の半径と開始, 終了角を求める (AR,AR1,AR3)
2360 AR=SQR(ABS((X(3)-AX0)*(X(3)-AX0)+(Y(3)-AY0)*(Y(3)-AY0))) '半径 R を求める
2370 AX=X(3) : AZ=(AY0-Y(3))/AR : GOSUB *ARCSIN : AR3=AS
2380 AX=X(2) : AZ=(AY0-Y(2))/AR : GOSUB *ARCSIN : AR2=AS
2390 AX=X(1) : AZ=(AY0-Y(1))/AR : GOSUB *ARCSIN : AR1=AS
2400 IF AR3<AR2 AND AR2<AR1 THEN SWAP AR1,AR3 'AR1,AR2,AR3の位置関係
2410 IF AR2>AR3 AND AR3>AR1 THEN SWAP AR1,AR3 'によりAR1とAR3を交換する
2420 IF AR2<AR1 AND AR1<AR3 THEN SWAP AR1,AR3
2430 '
2440 '円弧 B の半径と開始, 終了角を求める (BR,BR1,BR3)
2450 BR=SQR(ABS((X(6)-BX0)*(X(6)-BX0)+(Y(6)-BY0)*(Y(6)-BY0)))
2460 BX=X(6) : BZ=(BY0-Y(6))/BR : GOSUB *ARCSIN : BR3=BS
2470 BX=X(5) : BZ=(BY0-Y(5))/BR : GOSUB *ARCSIN : BR2=BS
2480 BX=X(4) : BZ=(BY0-Y(4))/BR : GOSUB *ARCSIN : BR1=BS
2490 IF BR3<BR2 AND BR2<BR1 THEN SWAP BR1,BR3 'BR1,BR2,BR3の位置関係
2500 IF BR2>BR3 AND BR3>BR1 THEN SWAP BR1,BR3 'によりBR1とBR3を交換する
2510 IF BR2<BR1 AND BR1<BR3 THEN SWAP BR1,BR3
2520 '
2530 '円弧 A を描く
2540 CIRCLE (AX0,AY0),AR,,AR1,AR3
2550 '
2560 '円弧 B を描く
2570 CIRCLE (BX0,BY0),BR,,BR1,BR3
2580 '
2590 '円弧を結ぶ線
2600 LINE (X(1),Y(1))-(X(8),Y(8)),C
2610 LINE-(X(4),Y(4)),C
2620 LINE (X(3),Y(3))-(X(7),Y(7)),C
2630 LINE-(X(6),Y(6)),C
2640 '
2650 CIRCLE (320,340),10,, '主軸の円弧
2660 RETURN
2670 '
2680 '
2690 'アークサインを求めるサブルーチン
2700 *ARCSIN
2710 '
2720 '円弧 A のアークサイン
2730 AS=ATN(AZ/SQR(ABS(1-AZ*AZ))) '円弧 A SIN^-1 (逆SIN)をアークタンジェントで求める
2740 AXFL=AX-AX0 '求めている点が第何象限
2750 IF AXFL>=0 AND AS>=0 THEN GOTO 2800 'にあるかで補正する
2760 IF AXFL>=0 AND AS<0 THEN AS=AS+6.28318 :GOTO 2800
2770 AS=3.14159-AS
2780 '
2790 '円弧 B のアークサイン
2800 BS=ATN(BZ/SQR(ABS(1-BZ*BZ))) '円弧 B SIN^-1 (逆SIN)をアークタンジェントで求める
2810 BXFL=BX-BX0 '求めている点が第何象限
2820 IF BXFL>=0 AND BS>=0 THEN GOTO 2850 'にあるかで補正する
2830 IF BXFL>=0 AND BS<0 THEN BS=BS+6.28318 : GOTO 2850
2840 BS=3.14159-BS
2850 RETURN
2860 '
2870 'ピストンを描くサブルーチン
2880 *PISTON
2890 '
2900 '
2910 '連結棒
2920 CIRCLE (X(9),Y(9)),10,C
2930 CIRCLE (X(9),Y(9)),18,C,180/180*P,360/180*P
2940 LINE ((X(9))+18,(Y(9)))-(331,187+H),C '187+Hで Y の値を求める。
2950 LINE ((X(9))-18,(Y(9)))-(309,187+H),C
2960 '
2970 'ピストンの外形
2980 LINE (285,187+H)-(355,187+H),C

```

```

2990 LINE-(355.197+H),C
3010 LINE-(375.132+H),C
3020 LINE-(265.132+H),C
3030 LINE-(265.197+H),C
3040 LINE-(285.197+H),C
3050 LINE-(285.187+H),C
3060 LINE (260.5,140+H)-(379.3,145+H),C,B      'リングみぞ部の線
3070 CIRCLE (320.174+H).8,C                      '連結棒との連結部分
3080 RETURN
3090 '
3100 *KAITEN                                     '回転した座標を求めるサブルーチン
3110 XX=X*COS(R/180*P)-Y*SIN(R/180*P)
3120 YY=X*SIN(R/180*P)+Y*COS(R/180*P)
3130 RETURN
3140 '
3150 *BEN                                         '弁を描くサブルーチン
3160 ' 弁 (左)
3170 LINE (LBX(1),LBY(1))-(LBX(2),LBY(2)),C
3180 LINE-(LBX(3),LBY(3)),C
3190 LINE-(LBX(4),LBY(4)),C
3200 LINE-(LBX(5),LBY(5)),C
3210 LINE-(LBX(6),LBY(6)),C
3220 LINE-(LBX(7),LBY(7)),C
3230 LINE-(LBX(8),LBY(8)),C
3240 LINE-(LBX(1),LBY(1)),C
3250 ' 弁 (右)
3260 LINE (RBX(1),RBY(1))-(RBX(2),RBY(2)),C
3270 LINE-(RBX(3),RBY(3)),C
3280 LINE-(RBX(4),RBY(4)),C
3290 LINE-(RBX(5),RBY(5)),C
3300 LINE-(RBX(6),RBY(6)),C
3310 LINE-(RBX(7),RBY(7)),C
3320 LINE-(RBX(8),RBY(8)),C
3330 LINE-(RBX(1),RBY(1)),C
3340 RETURN
3350 '
3360 *SIRNDA                                     'シリンダを描くサブルーチン
3370 '
3380 ' (動かない部分)
3390 '
3400 'シリンダ部の作図
3410 '
3420 '左側シリンダの壁
3430 LINE (260,270)-(257,115),1,BF
3440 LINE (170,95)-(245,98),1,BF
3450 LINE (260,115)-(245,95),1
3460 LINE (257,115)-(242,95),1
3470 PAINT (258,114),1,1
3480 '
3490 '右側シリンダの壁
3500 LINE (380,270)-(383,115),1,BF
3510 LINE (470,95)-(395,98),1,BF
3520 LINE (380,115)-(395,95),1
3530 LINE (383,115)-(398,95),1
3540 PAINT (382,114),1,1
3550 '
3560 '吸気部
3570 '
3580 ' (左側)
3590 LINE (170,65)-(270,62),1,BF
3600 '
3610 ' (右側)
3620 LINE (470,65)-(370,62),1,BF
3630 '
3640 'プラグ部分 (下側)
3650 LINE (260+10,115-50)-(260+30,115-20),1
3660 LINE-(380-30,115-20),1
3670 LINE-(380-10,115-50),1

```

```

3680 ' 上側の線
3690 LINE (380-10,115-53)-(380-30+10,115-53),1
3700 LINE-(380-30-5,115-25),1
3710 LINE-(260+30+5,115-25),1
3720 LINE-(260+30-10,115-53),1
3730 LINE-(260+10,115-53),1
3740 '
3750 ' プラグの部分の塗りつぶし
3760 PAINT (260+30,115-26),1
3770 '
3780 ' プラグ
3790 LINE (320-10,115-20)-(320+10,115-25),6,B
3800 LINE (320-15,115-25)-(320+15,115-25-10),6,B
3810 LINE (320-10,115-25-10)-(320+10,115-25-10-30),6,B
3820 LINE (320-5,115-25-10-30)-(320+5,115-25-10-30-5),6,B
3830 LINE (320-3,115-20)-(320+3,115-20+2),6,B
3840 RETURN
3850 '
3860 *KYUKI.KOU ' 吸気行程のサブルーチン
3870 M=10: N=12 : YA=190
3880 FOR J=1 TO 8
3890 BX0=LBX(J) : BY0=LBY(J)
3900 GOSUB *HEIKOU.IDO
3910 LBX(J)=BX : LBY(J)=BY
3920 NEXT J
3930 TILE$=CHR$(&HCC)+CHR$(&H0)+CHR$(&H33)+CHR$(&H0)
3940 GOSUB *BEN
3950 GOSUB *YAZIRUSHI
3960 GOSUB *KPLINE
3970 PAINT (320,125),TILE$.C
3980 RETURN
3990 '
4000 *LBEN.DEL
4010 FOR J=1 TO 9
4020 LBX(J)=LBX(J)-M
4030 LBY(J)=LBY(J)-N
4040 NEXT J
4050 RETURN
4060 '
4070 *KYUKI.DEL
4080 PAINT (320,125),0.7
4090 GOSUB *KPLINE
4100 GOSUB *BEN
4110 GOSUB *YAZIRUSHI
4120 GOSUB *LBEN.DEL
4130 RETURN
4140 '
4150 *ASSYUKU.KOU ' 圧縮行程のサブルーチン
4160 TILE$=CHR$(&H81)+CHR$(&H42)+CHR$(&H24)+CHR$(&H18)+CHR$(&H18)+CHR$(&H24)+CHR$(&H42)+CHR$(&H81)
4170 GOSUB *BEN
4180 GOSUB *PALINE
4190 PAINT (320,125),TILE$.C
4200 RETURN
4210 '
4220 *ASSYUKU.DEL
4230 PAINT (320,125),0.7
4240 GOSUB *PALINE
4250 GOSUB *BEN
4260 RETURN
4270 '
4280 *BOUCHO.KOU ' 膨張行程のサブルーチン
4290 TILE$=CHR$(&H81)+CHR$(&H42)+CHR$(&H24)+CHR$(&H18)+CHR$(&H18)+CHR$(&H24)+CHR$(&H42)+CHR$(&H81)
4300 GOSUB *BEN
4310 GOSUB *PALINE
4320 PAINT (320,125),TILE$.C
4330 RETURN

```

```

4340 '
4350 *BOUCHO.DEL
4360 PAINT (320,125),0.7
4370 GOSUB *PALINE
4380 GOSUB *BEN
4390 RETURN
4400 '
4410 *HAIKI.KOU                                ' 排気行程のサブルーチン
4420 M=-10: N=12 : YA=420
4430 FOR J=1 TO 8
4440   BX0=RBX(J) : BY0=RBX(J)
4450   GOSUB *HEIKOU.IDO
4460   RBX(J)=BX : RBY(J)=BY
4470 NEXT J
4480 TILE$=CHR$(&HCC)+CHR$(&H0)+CHR$(&H33)+CHR$(&H0)
4490 GOSUB *BEN
4500 GOSUB *YAZIRUSHI
4510 GOSUB *HPLINE
4520 PAINT (320,125),TILE$.C
4530 RETURN
4540 '
4550 *HAIKI.DEL
4560 PAINT (320,125),0.7
4570 GOSUB *HPLINE
4580 GOSUB *BEN
4590 GOSUB *YAZIRUSHI
4600 GOSUB *RBEN.DEL
4610 RETURN
4620 '
4630 *RBEN.DEL
4640 FOR J=1 TO 8
4650   RBX(J)=RBX(J)-M
4660   RBY(J)=RBY(J)-N
4670 NEXT J
4680 RETURN
4690 '
4700 *HEIKOU.IDO                                ' 平行移動させるためのサブルーチン
4710 BX=BX0+M
4720 BY=BY0+N
4730 RETURN
4740 '
4750 *YAZIRUSHI                                ' 吸気と排気行程の時に表示され
4760 LINE (YA,80)-(YA+30,80),C                ' る矢印
4770 LINE (YA+20,74)-(YA+30,80),C
4780 LINE-(YA+20,86),C
4790 RETURN
4800 '
4810 *PALINE                                    ' 2画面を合成しているので、
4820 LINE (260.5,140+H)-(260.5,115),C        ' 枠を描かないと全部が塗
4830 LINE-(290,95),C                          ' れてしまうので、枠となる
4840 LINE-(350,95),C                          ' 線を描く
4850 LINE-(379.3,115),C                      ' 圧縮と膨張の時の枠
4860 LINE-(379.3,140+H),C
4870 RETURN
4880 '
4890 '
4900 *KPLINE                                    ' 吸気の時枠
4910 LINE (260.5,140+H)-(260.5,115),C
4920 LINE-(245,95),C
4930 LINE-(170,95),C
4940 LINE-(170,65),C
4950 LINE-(270,65),C
4960 LINE-(290,95),C
4970 LINE-(350,95),C
4980 LINE-(379.3,115),C
4990 LINE-(379.3,140+H),C
5000 RETURN
5010 '

```

```

5020 *HPLINE                                ' 排気の時 の 枠
5030 LINE (260.5,140+H)-(260.5,115).C
5040 LINE-(290,95).C
5050 LINE-(350,95).C
5060 LINE-(370,65).C
5070 LINE-(470,65).C
5080 LINE-(470,95).C
5090 LINE-(395,95).C
5100 LINE-(379.3,115).C
5110 LINE-(379.3,140+H).C
5120 RETURN
5130 '
5140 *TENKA                                ' 点火したときの様子を 描く
5150 LINE (310,95)-(295,110).C
5160 LINE-(308,105).C
5170 LINE-(305,115).C
5180 LINE-(315,108).C
5190 LINE-(320,118).C
5200 LINE-(325,108).C
5210 LINE-(335,115).C
5220 LINE-(332,105).C
5230 LINE-(345,110).C
5240 LINE-(330,95).C
5250 LINE-(310,95).C
5260 RETURN
5270 '
5280 *TENKA.GRA
5290 GOSUB *TENKA
5300 PAINT (320,105).2.C
5310 GOSUB *BEN
5320 GOSUB *BOUCHO.KOU
5330 RETURN
5340 '
5350 *TENKA.DEL
5360 PAINT (320,105).0.7
5370 GOSUB *BOUCHO.DEL
5380 GOSUB *TENKA
5390 RETURN
5400 '
5410 '
5420 *KYUKI.MZI                            ' 吸気行程の文字
5430 CLS
5440 COLOR 5
5450 LOCATE 2,6 : PRINT "          吸気行程          "
5460 LOCATE 2,7 : PRINT "      (intake stroke)      "
5470 RETURN
5480 '
5490 *ASSYUKU.MZI                          ' 圧縮行程の文字
5500 CLS
5510 COLOR 6
5520 LOCATE 2,6 : PRINT "          圧縮行程          "
5530 LOCATE 2,7 : PRINT "      (compression stroke)  "
5540 RETURN
5550 '
5560 *TENKA.MZI                            ' 点火の文字
5570 CLS
5580 COLOR 2
5590 LOCATE 2,6 : PRINT "          点    火          "
5600 RETURN
5610 '
5620 *BOUCHO.MZI                          ' 膨張行程の文字
5630 CLS
5640 COLOR 3
5650 LOCATE 2,6 : PRINT "          膨張行程          "
5660 LOCATE 2,7 : PRINT "      (expansion stroke)  "
5670 '
5680 RETURN
5690 '

```

```

5700 *HAIKI.MZI                      ' 排気行程の文字
5710 CLS
5720 COLOR 6
5730 LOCATE 2,6 : PRINT "          排気行程          "
5740 LOCATE 2,7 : PRINT "    (exhaust stroke)    "
5750 '
5760 RETURN
5770 '
5780 *SPEED.WRI
5790 LINE (30,150)-(175,250),3,B
5800 LOCATE 6,10 : COLOR 6 : PRINT "回転速度の変更"
5810 LOCATE 6,11 : COLOR 7 : PRINT "  1, 低 速  "
5820 LOCATE 6,12 : COLOR 7 : PRINT "  2, 中 速  "
5830 LOCATE 6,13 : COLOR 7 : PRINT "  3, 高 速  "
5840 LOCATE 6,14 : COLOR 7 : PRINT " 選んで下さい  "
5850 R$=INKEY$
5860 IF R$="1" THEN SP=12 : GOTO 1100
5870 IF R$="2" THEN SP=36 : GOTO 1100
5880 IF R$="3" THEN SP=90 : GOTO 1100
5890 GOTO 5850
5900 '
5910 *KEY.HAN
5920 IF A$="S" OR A$="s" THEN GOSUB *STOP.SIM
5930 IF A$="R" OR A$="r" THEN GOTO *SPEED.WRI
5940 IF A$="E" OR A$="e" THEN GOTO *END.
5950 RETURN
5960 '
5970 *STOP.SIM                        ' 停止させるためのサブルーチン
5980 FOR S=1 TO 1000
5990   A$=INKEY$
6000   IF A$=CHR$(13) THEN RETURN
6010   GOSUB *KEY.HAN
6020   IF S=1000 THEN S=1 : GOTO 5980
6030 NEXT S
6040 '
6050 *END.
6060 CONSOLE 0,25,0,1
6070 LOCATE 0,24
6080 FOR E=1 TO 25
6090   ROLL 16
6100   PRINT CHR$(13)
6110   FOR S=0 TO 200 : NEXT S
6120 NEXT E
6130 CLS 3
6140 RUN "MENU.BAS"

1000 .....
1010 'メニュープログラム
1020 '菅森義真
1030 'SAVE "MENU.BAS"
1040 .....
1050 '
1060 STOP ON : ON STOP GOSUB *END.
1070 MKEY=1 : MCA=7 : MCB=4 : MCC=4 : MCD=4 : MCE=4
1080 CONSOLE 0,25,0,1 : SCREEN 3,0 : CLS 3
1090 LINE (100,20)-(539,60),5,B
1100 PAINT (320,30),1,5
1110 LOCATE 22,2 : COLOR 6 : PRINT " 4 サイクルエンジンのシミュレーション "
1120 LOCATE 22,5 : COLOR 3 : PRINT "          ** MENU **          "
1130 LOCATE 22,8 : COLOR MCA : PRINT "  ☆ エンジンの各部の名称 "
1140 LOCATE 22,10 : COLOR MCB : PRINT "  ☆ ピストンの行程・死点・行程容積 "
1150 LOCATE 22,12 : COLOR MCC : PRINT "  ☆ 4 サイクルエンジンの "
1160 LOCATE 22,13 : COLOR MCC : PRINT "      動きのシミュレーション (手動) "
1170 LOCATE 22,15 : COLOR MCD : PRINT "  ☆ 4 サイクルエンジンの "
1180 LOCATE 22,16 : COLOR MCD : PRINT "      動きのシミュレーション (自動) "
1190 LOCATE 22,18 : COLOR MCE : PRINT "  ☆ プログラムの終了 "
1200 LOCATE 21,21 : COLOR 5 : PRINT " 上・下の矢印キーで、どれかを選択し "

```

```

1210 LOCATE 21,22 : COLOR 5 : PRINT "リターンキーを押して、決定して下さい。"
1220 M$=INKEY$
1230 IF M$=CHR$(&H1F) THEN MKEY=MKEY+1 : BEEP : IF MKEY=6 THEN MKEY=1
1240 IF M$=CHR$(&H1E) THEN MKEY=MKEY-1 : BEEP : IF MKEY=0 THEN MKEY=5
1250 IF M$=CHR$(13) THEN GOTO *SENTAKU
1260 IF MKEY=1 THEN 1270 ELSE 1280
1270 MK$="1" : MCA=7 : MCB=4 : MCC=4 : MCD=4 : MCE=4
1280 IF MKEY=2 THEN 1290 ELSE 1300
1290 MK$="2" : MCA=4 : MCB=7 : MCC=4 : MCD=4 : MCE=4
1300 IF MKEY=3 THEN 1310 ELSE 1320
1310 MK$="3" : MCA=4 : MCB=4 : MCC=7 : MCD=4 : MCE=4
1320 IF MKEY=4 THEN 1330 ELSE 1340
1330 MK$="4" : MCA=4 : MCB=4 : MCC=4 : MCD=7 : MCE=4
1340 IF MKEY=5 THEN 1350 ELSE 1360
1350 MK$="5" : MCA=4 : MCB=4 : MCC=4 : MCD=4 : MCE=7
1360 GOTO 1120
1370 '
1380 *SENTAKU : BEEP
1390 IF MK$="1" THEN GOTO *ENZNAME
1400 IF MK$="2" THEN GOTO *KOUTEI
1410 IF MK$="3" THEN GOTO *ENZSIM
1420 IF MK$="4" THEN GOTO *AUTOSIM
1430 IF MK$="5" THEN GOTO *END.
1440 '
1450 *ENZSIM
1460 COLOR 7 : CLS 3
1470 LOCATE 12,5 : PRINT " このプログラムは、4サイクルガソリンエンジンの動き
"
1480 LOCATE 12,7 : PRINT "を示したもので、ピストンが2回の往復運動をする間に、
"
1490 LOCATE 12,9 : PRINT "混合気の吸い込みから、燃焼ガスの排出まで行って動力を
"
1500 LOCATE 12,11 : PRINT "発生することを示したものです。
"
1510 LOCATE 12,13 : PRINT " なお、操作方法については画面の右下に表示されますの
"
1520 LOCATE 12,15 : PRINT "で、それを見て操作して下さい。
"
1530 GOSUB *SUSUMU
1540 GOSUB *OMATI
1550 RUN "ENZSIM.BAS"
1560 '
1570 *AUTOSIM
1580 GOSUB *OMATI
1590 RUN "AUTOSIM.BAS"
1600 '
1610 *ENZNAME
1620 GOSUB *OMATI
1630 RUN "ENZNAME.BAS"
1640 '
1650 *KOUTEI
1660 GOSUB *OMATI
1670 RUN "KOUTEI.BAS"
1680 '
1690 *SUSUMU
1700 LOCATE 50,18 : COLOR 2 : PRINT "何かキーを押して下さい。": COLOR 7
1710 S$=INKEY$
1720 IF S$="" THEN 1710
1730 IF S$=" " THEN RETURN ELSE RETURN
1740 '
1750 *OMATI
1760 CLS 3
1770 LOCATE 25,10 : COLOR 2 : PRINT "プログラムを起動しています . . . 。"
1780 LOCATE 25,12 : COLOR 2 : PRINT "  少しお待ち下さい。"
1790 FOR S=0 TO 5000 : NEXT S
1800 RETURN
1810 '
1820 *END.

```

```

1830 LOCATE 0.24
1840 FOR E=1 TO 25
1850 ROLL 16
1860 PRINT CHR$(13)
1870 FOR S=0 TO 200 : NEXT S
1880 NEXT E
1890 CONSOLE 0.25,0.1 : SCREEN 3.0 : COLOR 7 : CLS 3 : NEW
1900 END

1000 .....
1010 ' 4 サイクルエンジンのシミュレーション
1020 ' 菅森義真
1030 ' SAVE "ENZSIM.BAS"
1040 .....
1050 '
1060 '
1070 ' まず動かない部分を先に描く
1080 STOP ON : ON STOP GOSUB *END.
1090 SP=36
1100 SCREEN 3.0 : CONSOLE 0.25,0.1 : CLS 3
1110 GOSUB *SIRNDA
1120 '
1130 CLS
1140 LINE (0,0)-(633,399),1,B
1150 LINE (2,2)-(631,397),1,B
1160 LOCATE 1,1 : COLOR 4 : PRINT " 4 サイクルエンジンの "
1170 LOCATE 1,2 : COLOR 4 : PRINT " シミュレーション "
1180 LOCATE 52,19 : COLOR 2 : PRINT " キー操作 "
1190 LOCATE 52,20 : COLOR 6 : PRINT " エンジンの始動 : リターン "
1200 LOCATE 52,21 : COLOR 6 : PRINT " 回転速度の変更 : R "
1210 LOCATE 52,22 : COLOR 6 : PRINT " 画面の印刷 : P "
1220 LOCATE 52,23 : COLOR 6 : PRINT " メニューに戻る : E "
1230 IF SP=12 THEN SP$="低 速" : GOTO 1260
1240 IF SP=36 THEN SP$="中 速" : GOTO 1260
1250 IF SP=90 THEN SP$="高 速" : GOTO 1260
1260 LOCATE 4,22 : COLOR 4 : PRINT "現在の回転速度=";
1270 COLOR 2 : PRINT SP$
1280 COLOR 7
1290 '
1300 ' 次に動く部分を描いていく (クランク、ピストン、弁)
1310 CONSOLE 3,16,0,1
1320 P=3.14159 : R=-SP : C=7 : H=0 : RC=180/ABS(R) : B=0
1330 L=68/RC ' 回転の角度とピストンの上下の間隔
1340 ' クランクとピストンの座標
1350 X(1)=376 : Y(1)=341
1360 X(2)=321 : Y(2)=396
1370 X(3)=264 : Y(3)=341
1380 X(4)=342 : Y(4)=305
1390 X(5)=321 : Y(5)=284
1400 X(6)=298 : Y(6)=305
1410 X(7)=298 : Y(7)=334
1420 X(8)=342 : Y(8)=334
1430 X(9)=320 : Y(9)=306
1440 ' 左の弁の座標
1450 LBX(1)=260 : LBY(1)=115
1460 LBX(2)=290 : LBY(2)=95
1470 LBX(3)=285 : LBY(3)=89
1480 LBX(4)=273 : LBY(4)=97
1490 LBX(5)=228 : LBY(5)=27
1500 LBX(6)=224 : LBY(6)=30
1510 LBX(7)=269 : LBY(7)=100
1520 LBX(8)=255 : LBY(8)=109
1530 ' 右の弁の座標
1540 RBX(1)=380 : RBY(1)=115
1550 RBX(2)=350 : RBY(2)=95
1560 RBX(3)=355 : RBY(3)=89
1570 RBX(4)=367 : RBY(4)=97

```



```

1580 RBX(5)=412 : RBY(5)=27
1590 RBX(6)=416 : RBY(6)=30
1600 RBX(7)=371 : RBY(7)=100
1610 RBX(8)=385 : RBY(8)=109
1620 '
1630 GOSUB *KURANKU 'サブ ルーチンへ
1640 GOSUB *PISTON 'サブ ルーチンへ
1650 GOSUB *BEN 'サブ ルーチンへ
1660 A$=INKEY$
1670 IF A$="P" OR A$="p" THEN COPY 3
1680 IF A$="R" OR A$="r" THEN GOTO *SPEED.WRI
1690 IF A$="E" OR A$="e" THEN GOTO *END.
1700 IF A$=CHR$(13) THEN GOTO 1710 ELSE 1660
1710 C=0
1720 IF B=0 THEN GOSUB *BEN : GOTO 1790
1730 IF B=>1 AND B<RC THEN : GOSUB *KYUKI.DEL : GOTO 1790
1740 IF B>RC AND B<RC*2 THEN GOSUB *ASSYUKU.DEL : GOTO 1790
1750 IF B=RC*2 THEN GOSUB *TENKA.DEL : GOTO 1790
1760 IF B=>RC*2 AND B<RC*3 THEN GOSUB *BOUCHO.DEL : GOTO 1790
1770 IF B>RC*3 AND B<RC*4 THEN GOSUB *HAIKI.DEL : GOTO 1790
1780 IF B=RC*4 THEN GOSUB *BEN : GOTO 1790
1790 GOSUB *KURANKU
1800 GOSUB *PISTON
1810 C=7
1820 LINE (170,65)-(270,62),1,BF '線をきれいにするため
1830 LINE (470,65)-(370,62),1,BF
1840 FOR I=1 TO 9
1850 X=X(I)-320 : Y=340-Y(I)
1860 GOSUB *KAITEN 'サブ ルーチンへ
1870 X(I)=320+XX : Y(I)=340-YY
1880 NEXT I
1890 GOSUB *KURANKU 'サブ ルーチンへ
1900 IF Y(9)<306 THEN H=H-L 'X(9)の値と、Y(9)の値が
1910 IF X(9)>320 AND Y(9)<374 THEN H=H+L 'どこにあるかで、ピストンを
1920 IF Y(9)>374 THEN H=H+L '上下させるHの値の増減を決
1930 IF X(9)<320 AND Y(9)>306 THEN H=H-L 'める。
1940 GOSUB *PISTON 'サブ ルーチンへ
1950 B=B+1
1960 IF B>RC*4 THEN B=1 'Bの値によって、何行程かを指定する。
1970 IF B=1 THEN GOSUB *KYUKI.MZI
1980 IF B=>1 AND B<RC THEN GOSUB *KYUKI.KOU :GOTO 2100
1990 IF B=RC+1 THEN GOSUB *ASSYUKU.MZI
2000 IF B>RC AND B<RC*2 THEN GOSUB *ASSYUKU.KOU :GOTO 2100
2010 IF B=RC*2 THEN GOSUB *TENKA.MZI
2020 IF B=RC*2 THEN GOSUB *TENKA.GRA
2030 IF B=RC*2+1 THEN GOSUB *BOUCHO.MZI
2040 IF B=>RC*2 AND B<RC*3 THEN GOSUB *BOUCHO.KOU :GOTO 2100
2050 IF B=RC*3+1 THEN GOSUB *HAIKI.MZI
2060 IF B>RC*3 AND B<RC*4 THEN GOSUB *HAIKI.KOU :GOTO 2100
2070 IF B=RC*4 THEN GOSUB *BEN
2080 LINE (320-10,115-20)-(320+10,115-25),6,B 'プラグの先端
2090 LINE (320-3,115-20)-(320+3,115-20+2),6,B
2100 GOTO 1660
2110 '
2120 '
2130 'クランクを描くサブ ルーチン
2140 *KURANKU
2150 '
2160 '色の変数
2170 COLOR ...C
2180 '
2190 '円弧Aの中心を求める (AX0,AY0)
2200 AA=X(1)-X(2) : AB=Y(1)-Y(2) : AC=X(2)-X(3) : AD=Y(2)-Y(3) '円弧A 中心
2210 AE=X(1)*X(1)-X(2)*X(2) : AF=Y(1)*Y(1)-Y(2)*Y(2)
2220 AG=X(2)*X(2)-X(3)*X(3) : AH=Y(2)*Y(2)-Y(3)*Y(3)
2230 AY0=((AE+AF)*AC-(AG+AH)*AA)/(AB*AC-AD*AA)/2
2240 AX0=(AE+AF-2*AB*AY0)/AA/2
2250 '

```

```

2260 '円弧 B の中心を求める (BX0,BY0)
2270 BA=X(4)-X(5) : BB=Y(4)-Y(5) : BC=X(5)-X(6) : BD=Y(5)-Y(6) '円弧 B 中心
2280 BE=X(4)*X(4)-X(5)*X(5) : BF=Y(4)*Y(4)-Y(5)*Y(5)
2290 BG=X(5)*X(5)-X(6)*X(6) : BH=Y(5)*Y(5)-Y(6)*Y(6)
2300 BY0=((BE+BF)*BC-(BG+BH)*BA)/(BB*BC-BD*BA)/2
2310 BX0=(BE+BF-2*BB*BY0)/BA/2
2320 '
2330 '円弧 A の半径と開始, 終了角を求める (AR,AR1,AR3)
2340 AR=SQR(ABS((X(3)-AX0)*(X(3)-AX0)+(Y(3)-AY0)*(Y(3)-AY0))) '半径 R を求める
2350 AX=X(3) : AZ=(AY0-Y(3))/AR : GOSUB *ARCSIN : AR3=AS
2360 AX=X(2) : AZ=(AY0-Y(2))/AR : GOSUB *ARCSIN : AR2=AS
2370 AX=X(1) : AZ=(AY0-Y(1))/AR : GOSUB *ARCSIN : AR1=AS
2380 IF AR3<AR2 AND AR2<AR1 THEN SWAP AR1,AR3 'AR1,AR2,AR3 の位置関係
2390 IF AR2>AR3 AND AR3>AR1 THEN SWAP AR1,AR3 'により AR1 と AR3 を交換する
2400 IF AR2<AR1 AND AR1<AR3 THEN SWAP AR1,AR3
2410 '
2420 '円弧 B の半径と開始, 終了角を求める (BR,BR1,BR3)
2430 BR=SQR(ABS((X(6)-BX0)*(X(6)-BX0)+(Y(6)-BY0)*(Y(6)-BY0)))
2440 BX=X(6) : BZ=(BY0-Y(6))/BR : GOSUB *ARCSIN : BR3=BS
2450 BX=X(5) : BZ=(BY0-Y(5))/BR : GOSUB *ARCSIN : BR2=BS
2460 BX=X(4) : BZ=(BY0-Y(4))/BR : GOSUB *ARCSIN : BR1=BS
2470 IF BR3<BR2 AND BR2<BR1 THEN SWAP BR1,BR3 'BR1,BR2,BR3 の位置関係
2480 IF BR2>BR3 AND BR3>BR1 THEN SWAP BR1,BR3 'により BR1 と BR3 を交換する
2490 IF BR2<BR1 AND BR1<BR3 THEN SWAP BR1,BR3
2500 '
2510 '円弧 A を描く
2520 CIRCLE (AX0,AY0),AR,,AR1,AR3
2530 '
2540 '円弧 B を描く
2550 CIRCLE (BX0,BY0),BR,,BR1,BR3
2560 '
2570 '円弧を結ぶ線
2580 LINE (X(1),Y(1))-(X(8),Y(8)),C
2590 LINE-(X(4),Y(4)),C
2600 LINE (X(3),Y(3))-(X(7),Y(7)),C
2610 LINE-(X(6),Y(6)),C
2620 '
2630 CIRCLE (320,340),10,, '主軸の円弧
2640 RETURN
2650 '
2660 '
2670 'アークサインを求めるサブルーチン
2680 *ARCSIN
2690 '
2700 '円弧 A のアークサイン
2710 AS=ATN(AZ/SQR(ABS(1-AZ*AZ))) '円弧 A SIN^-1 (逆SIN)をアークタンジェントで求める
2720 AXFL=AX-AX0 '求めている点が第何象限
2730 IF AXFL>=0 AND AS>=0 THEN GOTO 2780 'にあるかで補正する
2740 IF AXFL>=0 AND AS<0 THEN AS=AS+6.28318 :GOTO 2780
2750 AS=3.14159-AS
2760 '
2770 '円弧 B のアークサイン
2780 BS=ATN(BZ/SQR(ABS(1-BZ*BZ))) '円弧 B SIN^-1 (逆SIN)をアークタンジェントで求める
2790 BXFL=BX-BX0 '求めている点が第何象限
2800 IF BXFL>=0 AND BS>=0 THEN GOTO 2830 'にあるかで補正する
2810 IF BXFL>=0 AND BS<0 THEN BS=BS+6.28318 : GOTO 2830
2820 BS=3.14159-BS
2830 RETURN
2840 '
2850 'ピストンを描くサブルーチン
2860 *PISTON
2870 '
2880 '
2890 '連結棒
2900 CIRCLE (X(9),Y(9)),10,C
2910 CIRCLE (X(9),Y(9)),18,C,180/180*P,360/180*P
2920 LINE ((X(9))+18,(Y(9)))-(331,187+H),C '187+Hで Y の値を求める。
2930 LINE ((X(9))-18,(Y(9)))-(309,187+H),C

```

```

2940 '
2950 'ピストンの外形
2960 LINE (285,187+H)-(355,187+H),C
2970 LINE-(355,197+H),C
2980 LINE-(375,197+H),C
2990 LINE-(375,132+H),C
3000 LINE-(265,132+H),C
3010 LINE-(265,197+H),C
3020 LINE-(285,197+H),C
3030 LINE-(285,187+H),C
3040 LINE (260.5,140+H)-(379.3,145+H),C,B      'リングみぞ部の線
3050 CIRCLE (320,174+H),8,C      '連結棒との連結部分
3060 RETURN
3070 '
3080 *KAITEN      '回転した座標を求めるサブルーチン
3090 XX=X*COS(R/180*P)-Y*SIN(R/180*P)
3100 YY=X*SIN(R/180*P)+Y*COS(R/180*P)
3110 RETURN
3120 '
3130 *BEN      '弁を描くサブルーチン
3140 '弁 (左)
3150 LINE (LBX(1),LBY(1))-(LBX(2),LBY(2)),C
3160 LINE-(LBX(3),LBY(3)),C
3170 LINE-(LBX(4),LBY(4)),C
3180 LINE-(LBX(5),LBY(5)),C
3190 LINE-(LBX(6),LBY(6)),C
3200 LINE-(LBX(7),LBY(7)),C
3210 LINE-(LBX(8),LBY(8)),C
3220 LINE-(LBX(1),LBY(1)),C
3230 '弁 (右)
3240 LINE (RBX(1),RBY(1))-(RBX(2),RBY(2)),C
3250 LINE-(RBX(3),RBY(3)),C
3260 LINE-(RBX(4),RBY(4)),C
3270 LINE-(RBX(5),RBY(5)),C
3280 LINE-(RBX(6),RBY(6)),C
3290 LINE-(RBX(7),RBY(7)),C
3300 LINE-(RBX(8),RBY(8)),C
3310 LINE-(RBX(1),RBY(1)),C
3320 RETURN
3330 '
3340 *SIRNDA      'シリンダを描くサブルーチン
3350 '
3360 ' (動かない部分)
3370 '
3380 'シリンダ部の作図
3390 '
3400 '左側シリンダの壁
3410 LINE (260,270)-(257,115),1,BF
3420 LINE (170,95)-(245,98),1,BF
3430 LINE (260,115)-(245,95),1
3440 LINE (257,115)-(242,95),1
3450 PAINT (258,114),1,1
3460 '
3470 '右側シリンダの壁
3480 LINE (380,270)-(383,115),1,BF
3490 LINE (470,95)-(395,98),1,BF
3500 LINE (380,115)-(395,95),1
3510 LINE (383,115)-(398,95),1
3520 PAINT (382,114),1,1
3530 '
3540 '吸気部
3550 '
3560 ' (左側)
3570 LINE (170,65)-(270,62),1,BF
3580 '
3590 ' (右側)
3600 LINE (470,65)-(370,62),1,BF
3610 '

```

```

3620 'プラグ部分 (下側)
3630 LINE (260+10,115-50)-(260+30,115-20),1
3640 LINE-(380-30,115-20),1
3650 LINE-(380-10,115-50),1
3660 '上側の線
3670 LINE (380-10,115-53)-(380-30+10,115-53),1
3680 LINE-(380-30-5,115-25),1
3690 LINE-(260+30+5,115-25),1
3700 LINE-(260+30-10,115-53),1
3710 LINE-(260+10,115-53),1
3720 '
3730 'プラグの部分の塗りつぶし
3740 PAINT (260+30,115-26),1
3750 '
3760 'プラグ
3770 LINE (320-10,115-20)-(320+10,115-25),6,B
3780 LINE (320-15,115-25)-(320+15,115-25-10),6,B
3790 LINE (320-10,115-25-10)-(320+10,115-25-10-30),6,B
3800 LINE (320-5,115-25-10-30)-(320+5,115-25-10-30-5),6,B
3810 LINE (320-3,115-20)-(320+3,115-20+2),6,B
3820 RETURN
3830 '
3840 *KYUKI.KOU '吸気行程のサブルーチン
3850 M=10: N=12 : YA=190
3860 FOR J=1 TO 8
3870 BX0=LBX(J) : BY0=LBX(J)
3880 GOSUB *HEIKOU.IDO
3890 LBX(J)=BX : LBY(J)=BY
3900 NEXT J
3910 TILE$=CHR$(&HCC)+CHR$(&H0)+CHR$(&H33)+CHR$(&H0)
3920 GOSUB *BEN
3930 GOSUB *YAZIRUSHI
3940 GOSUB *KPLINE
3950 PAINT (320,125),TILE$,C
3960 RETURN
3970 '
3980 *LBEN.DEL
3990 FOR J=1 TO 9
4000 LBX(J)=LBX(J)-M
4010 LBY(J)=LBY(J)-N
4020 NEXT J
4030 RETURN
4040 '
4050 *KYUKI.DEL
4060 PAINT (320,125),0,7
4070 GOSUB *KPLINE
4080 GOSUB *BEN
4090 GOSUB *YAZIRUSHI
4100 GOSUB *LBEN.DEL
4110 RETURN
4120 '
4130 *ASSYUKU.KOU '圧縮行程のサブルーチン
4140 TILE$=CHR$(&H81)+CHR$(&H42)+CHR$(&H24)+CHR$(&H18)+CHR$(&H18)+CHR$(&H24)+CHR$(&H42)+CHR$(&H81)
4150 GOSUB *BEN
4160 GOSUB *PALINE
4170 PAINT (320,125),TILE$,C
4180 RETURN
4190 '
4200 *ASSYUKU.DEL
4210 PAINT (320,125),0,7
4220 GOSUB *PALINE
4230 GOSUB *BEN
4240 RETURN
4250 '
4260 *BOUCHO.KOU '膨張行程のサブルーチン
4270 TILE$=CHR$(&H81)+CHR$(&H42)+CHR$(&H24)+CHR$(&H18)+CHR$(&H18)+CHR$(&H24)+CHR$(&H42)+CHR$(&H81)

```

```

4280 GOSUB *BEN
4290 GOSUB *PALINE
4300 PAINT (320,125),TILE$.C
4310 RETURN
4320 '
4330 *BOUCHO.DEL
4340 PAINT (320,125),0.7
4350 GOSUB *PALINE
4360 GOSUB *BEN
4370 RETURN
4380 '
4390 *HAIKI.KOU
4400 M=-10: N=12 : YA=420
4410 FOR J=1 TO 8
4420   BX0=RBX(J) : BY0=RBX(J)
4430   GOSUB *HEIKOU.IDO
4440   RBX(J)=BX : RBY(J)=BY
4450 NEXT J
4460 TILE$=CHR$(&HCC)+CHR$(&H0)+CHR$(&H33)+CHR$(&H0)
4470 GOSUB *BEN
4480 GOSUB *YAZIRUSHI
4490 GOSUB *HPLINE
4500 PAINT (320,125),TILE$.C
4510 RETURN
4520 '
4530 *HAIKI.DEL
4540 PAINT (320,125),0.7
4550 GOSUB *HPLINE
4560 GOSUB *BEN
4570 GOSUB *YAZIRUSHI
4580 GOSUB *RBEN.DEL
4590 RETURN
4600 '
4610 *RBEN.DEL
4620 FOR J=1 TO 8
4630   RBX(J)=RBX(J)-M
4640   RBY(J)=RBY(J)-N
4650 NEXT J
4660 RETURN
4670 '
4680 *HEIKOU.IDO
4690 BX=BX0+M
4700 BY=BY0+N
4710 RETURN
4720 '
4730 *YAZIRUSHI
4740 LINE (YA,80)-(YA+30,80).C
4750 LINE (YA+20,74)-(YA+30,80).C
4760 LINE-(YA+20,86).C
4770 RETURN
4780 '
4790 *PALINE
4800 LINE (260.5,140+H)-(260.5,115).C
4810 LINE-(290,95).C
4820 LINE-(350,95).C
4830 LINE-(379.3,115).C
4840 LINE-(379.3,140+H).C
4850 RETURN
4860 '
4870 '
4880 *KPLINE
4890 LINE (260.5,140+H)-(260.5,115).C
4900 LINE-(245,95).C
4910 LINE-(170,95).C
4920 LINE-(170,65).C
4930 LINE-(270,65).C
4940 LINE-(290,95).C
4950 LINE-(350,95).C

```

・排気行程のサブルーチン

・平行移動させるためのサブルーチン

・吸気と排気行程の時に表示される矢印

・2画面を合成しているので、
・枠を描かないと全部が塗
・れてしまうので、枠となる
・線を描く
・圧縮と膨張の時の枠

・吸気の時枠

```

4960 LINE-(379.3,115),C
4970 LINE-(379.3,140+H),C
4980 RETURN
4990 '
5000 *HPLINE                                ' 排気の時 of 枠
5010 LINE (260.5,140+H)-(260.5,115),C
5020 LINE-(290.95),C
5030 LINE-(350.95),C
5040 LINE-(370.65),C
5050 LINE-(470.65),C
5060 LINE-(470.95),C
5070 LINE-(395.95),C
5080 LINE-(379.3,115),C
5090 LINE-(379.3,140+H),C
5100 RETURN
5110 '
5120 *TENKA                                ' 点火したときの様子を描く
5130 LINE (310.95)-(295,110),C
5140 LINE-(308,105),C
5150 LINE-(305,115),C
5160 LINE-(315,108),C
5170 LINE-(320,118),C
5180 LINE-(325,108),C
5190 LINE-(335,115),C
5200 LINE-(332,105),C
5210 LINE-(345,110),C
5220 LINE-(330.95),C
5230 LINE-(310.95),C
5240 RETURN
5250 '
5260 *TENKA.GRA
5270 GOSUB *TENKA
5280 PAINT (320,105),2,C
5290 GOSUB *BEN
5300 GOSUB *BOUCHO.KOU
5310 RETURN
5320 '
5330 *TENKA.DEL
5340 PAINT (320,105),0.7
5350 GOSUB *BOUCHO.DEL
5360 GOSUB *TENKA
5370 RETURN
5380 '
5390 '
5400 *KYUKI.MZI                            ' 吸気行程の文字
5410 CLS
5420 COLOR 5
5430 LOCATE 7,6 : PRINT "吸気行程"
5440 LOCATE 4,7 : PRINT "(intake stroke)"
5450 '
5460 COLOR 7
5470 LOCATE 51,7 : PRINT "排気弁を閉じ、吸気弁を開く。"
5480 LOCATE 49,8 : PRINT "ピストンがシリンダ最上端から"
5490 LOCATE 49,9 : PRINT "最下端まで下降するにつれて、"
5500 LOCATE 49,10 : PRINT "シリンダ内の圧力は大気圧より"
5510 LOCATE 49,11 : PRINT "低くなり、気圧の差によって混"
5520 LOCATE 49,12 : PRINT "合気がシリンダ内に吸い込まれ"
5530 LOCATE 49,13 : PRINT "る。"
5540 RETURN
5550 '
5560 *ASSYUKU.MZI                          ' 圧縮行程の文字
5570 CLS
5580 COLOR 6
5590 LOCATE 8,6 : PRINT "圧縮行程"
5600 LOCATE 2,7 : PRINT "(compression stroke)"
5610 '
5620 COLOR 7
5630 LOCATE 49,7 : PRINT " 吸気弁が閉じ、ピストンが上"

```

```

5640 LOCATE 49,8 : PRINT "昇するにつれて、吸い込んだ混"
5650 LOCATE 49,9 : PRINT "合気が圧縮される。混合気は強"
5660 LOCATE 49,10 : PRINT "く圧縮すると、次の燃焼による"
5670 LOCATE 49,11 : PRINT "圧力をいっそう強くすることに"
5680 LOCATE 49,12 : PRINT "なる。"
5690 RETURN
5700 '
5710 *TENKA.MZI                      ' 点火の文字
5720 CLS
5730 COLOR 2
5740 LOCATE 6,6 : PRINT " 点 火 "
5750 '
5760 COLOR 7
5770 LOCATE 49,7 : PRINT " 圧 縮 行 程 の 終 わ り に 、 圧 縮 し "
5780 LOCATE 49,8 : PRINT " た こ と に よ っ て 圧 力 ・ 温 度 が と "
5790 LOCATE 49,9 : PRINT " も に 高 く な っ た 混 合 気 に ス パ ー "
5800 LOCATE 49,10 : PRINT " ク プ ラ グ で 電 気 火 花 を 飛 ば し て "
5810 LOCATE 49,11 : PRINT " 点 火 す る 。 "
5820 RETURN
5830 '
5840 *BOUCHO.MZI                      ' 膨張行程の文字
5850 CLS
5860 COLOR 3
5870 LOCATE 7,6 : PRINT " 膨 張 行 程 "
5880 LOCATE 2,7 : PRINT "(expansion stroke)"
5890 '
5900 COLOR 7
5910 LOCATE 49,7 : PRINT " ス パ ー ク プ ラ グ で 電 気 火 花 を "
5920 LOCATE 49,8 : PRINT " 飛 ば し て 点 火 す る と 、 混 合 気 は "
5930 LOCATE 49,9 : PRINT " 燃 焼 し 、 高 温 ・ 高 圧 の 燃 焼 ガ ス "
5940 LOCATE 49,10 : PRINT " と な る 。 ガ ス の 高 い 圧 力 が ピ ス "
5950 LOCATE 49,11 : PRINT " ト ン の 頂 面 に か か り 、 強 い 力 と "
5960 LOCATE 49,12 : PRINT " な っ て ピ ス ト ン を 勢 い よ く 押 し "
5970 LOCATE 49,13 : PRINT " 下 げ る 。 "
5980 RETURN
5990 '
6000 *HAIKI.MZI                      ' 排気行程の文字
6010 CLS
6020 COLOR 6
6030 LOCATE 7,6 : PRINT " 排 気 行 程 "
6040 LOCATE 3,7 : PRINT "(exhaust stroke)"
6050 '
6060 COLOR 7
6070 LOCATE 49,7 : PRINT " 燃 焼 ガ ス が 膨 張 し 終 わ る と 、 "
6080 LOCATE 49,8 : PRINT " 排 気 弁 が 開 き 、 な お 圧 力 の 残 っ "
6090 LOCATE 49,9 : PRINT " て い る ガ ス が 吹 き 出 し 、 シ リ ン "
6100 LOCATE 49,10 : PRINT " ダ 内 の 圧 力 は 下 が る 。 次 に ピ ス "
6110 LOCATE 49,11 : PRINT " ト ン は 上 昇 し な が ら 燃 焼 ず み の "
6120 LOCATE 49,12 : PRINT " ガ ス を 追 い 出 し て 、 シ リ ン ダ の "
6130 LOCATE 49,13 : PRINT " 最 上 端 に 達 し て 吸 気 を 始 め る 状 "
6140 LOCATE 49,14 : PRINT " 態 に も ど る 。 排 気 の 行 程 を 終 "
6150 LOCATE 49,15 : PRINT " え た 時 、 ピ ス ト ン の 上 部 に 残 っ "
6160 LOCATE 49,16 : PRINT " て い る ガ ス を 残 留 ガ ス と い う 。 "
6170 RETURN
6180 '
6190 *SPEED.WRI
6200 LINE (30,150)-(175,250),3,B
6210 LOCATE 6,10 : COLOR 6 : PRINT " 回 転 速 度 の 変 更 "
6220 LOCATE 6,11 : COLOR 7 : PRINT " 1 . 低 速 "
6230 LOCATE 6,12 : COLOR 7 : PRINT " 2 . 中 速 "
6240 LOCATE 6,13 : COLOR 7 : PRINT " 3 . 高 速 "
6250 LOCATE 6,14 : COLOR 7 : PRINT " 選 ん で 下 さ い "
6260 R$=INKEY$
6270 IF R$="1" THEN SP=12 : GOTO 1100
6280 IF R$="2" THEN SP=36 : GOTO 1100
6290 IF R$="3" THEN SP=90 : GOTO 1100
6300 GOTO 6260
6310 '

```

```
6320 *END.  
6330 CONSOLE 0.25.0.1  
6340 LOCATE 0.24  
6350 FOR E=1 TO 25  
6360   ROLL 16  
6370   PRINT CHR$(13)  
6380   FOR S=0 TO 200 : NEXT S  
6390 NEXT E  
6400 CLS 3  
6410 RUN "MENU.BAS"
```