

# 1 数値計算の基礎と地球惑星科学への応用

## 3 II. プレーートの冷却の数値シミュレーション・その2

4 プレーートの熱的進化を今度は Fortran プログラムによってシミュレートしてみよう。こ  
5 のプログラムは、熱伝導方程式の解をフーリエ級数によって表現している。

### 7 10. Fortran によるプレートの冷却の数値シミュレーション

8 前は Excel を使って計算したが、Fortran プログラムを動かして計算を行う。

#### 10 10.1 Fortran プログラムによる数値計算

11 ここでは、フーリエ級数による熱伝導方程式の解を離散化した式

$$12 \quad T_i(t) = T_0 + (T_M - T_0) \left\{ \frac{z_i}{L} + \sum_{n=1}^M \frac{2}{n\pi} \sin \left[ \frac{n\pi z_i}{L} \right] \exp \left[ \frac{n^2 \pi^2 \kappa t}{L^2} \right] \right\} \quad (10.1)$$

13 を Fortran により数値計算する方法を考える。ここで、与えるべき物理パラメータ  
14 は、

- 15 (1) 熱拡散率  $\kappa$
- 16 (2) 密度  $\rho$
- 17 (3) 定圧比熱  $C_p$
- 18 (4) マントルあるいはプレートの厚さ  $L$
- 19 (5) プレーートの年代  $t$

20 である。さらに、温度を計算する点の数  $m_z$  も必要である。点の間隔  $dz$  は、

$$21 \quad dz = \frac{L}{m_z} \quad (10.2)$$

22 であり、 $i$  番目の点の深さは

$$23 \quad z = dz \times i \quad (10.3)$$

24 と計算できる。

25 プログラムの中身はあとで説明するが、プログラムは以下のような処理の流れにな  
26 っている。

- 27 (1) パラメータをファイルから読み込む
- 28 (2)  $\kappa$  や  $dz$  を計算
- 29 (3)  $i$ -方向(深さ方向)への繰り返し
- 30 (4) フーリエ級数部分の計算：サブルーチン

31 (5) 温度の計算

32 (6) 深さと温度を出力する

33

## 34 10.2 パソコンによるプログラムの実行

35 プログラムファイルをダウンロードしたあと、プログラムを実行してプレートの温度  
36 を計算する

37

38 (1) プログラム格納用フォルダの作成を作成する。

39 フォルダ“fort”のなかに新しいフォルダ“Plate”を作る。

40 (2) プログラムとデータのダウンロード

41 ブラウザで URL

42 <https://home.hiroshima-u.ac.jp/nakakuki/Lectures/Geosys.%20ex..html>

43 を開く。

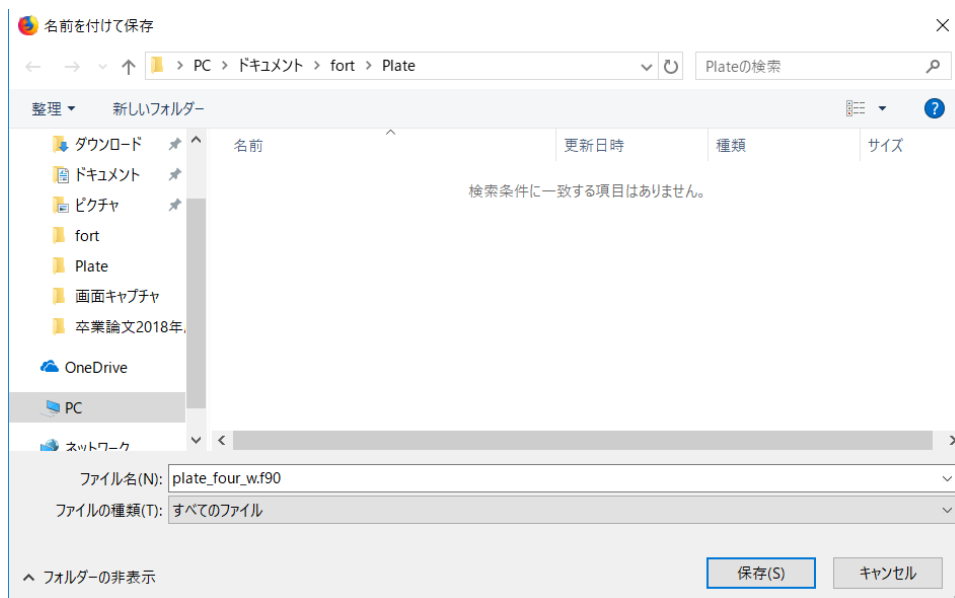
44 ページの中にある次の3つの Windows 用ファイルへのリンクのどれかをクリック  
45 (Mac の場合は Linux 用を使用のこと。2種類のファイルの違いは改行コード)す  
46 る。

47 plate\_main2\_w.f90

48 plate\_four\_w.f90

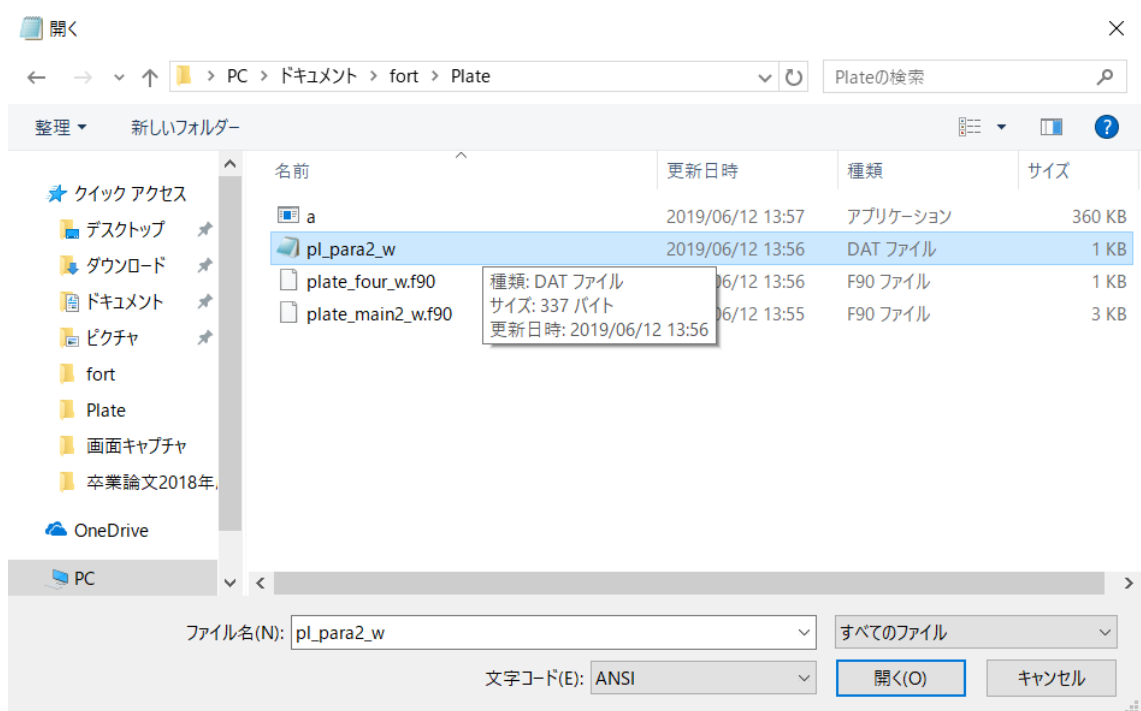
49 pl\_para2\_w.dat

50 ブラウザに表示させたあと、「名前をつけてページを保存」を選ぶ。保存ダイアロ  
51 グで、ファイルの種類として、「すべてのファイル」を選ぶ。



53

54 (3) メモ帳で“pl\_para2\_w.dat”を開く。このとき、[すべてのファイル]を選ぶ。



55

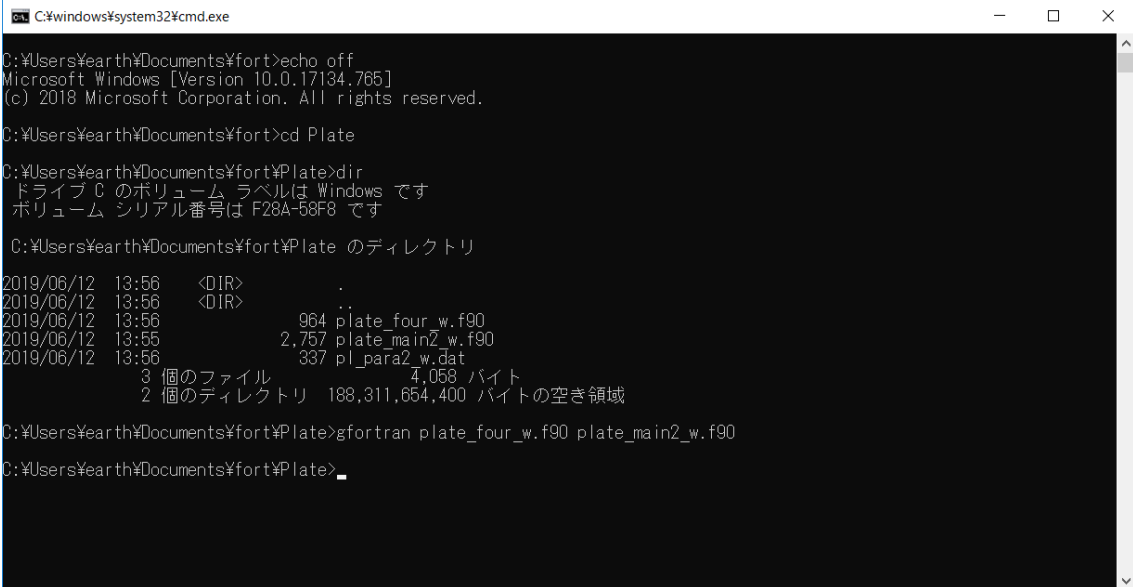
56

57 (4) 年代と結果の出力先ファイル名を設定する。“age of plate”の下にある数値と、  
58 “output file name”の下にある文字列である。後者のコーテーションマークは消して  
59 はいけない。変更したら保存する。

```
pl_para2_w - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
'density of the mantle (kg/m**3)'
3300.0d0
'specific heat (J/kg)'
1200.0d0
'thermal diffusivity (m^2/s)'
1.0d-6
'age of plate (Ma)'
20.0d0
'surface temperature (deg. C)'
0.0d0
'mantle temperature (deg. C)'
1300.0d0
'number of depth interval'
150
'plate thickness (km)'
300.0d0
'output file name'
'plate_A02_L300.data'
```

60

- 61 (5) Fortran をプログラムのフォルダ“fort”にある mingw-w64 をダブルクリックしてコ  
62 マンドプロンプトを起動する。  
63 (6) “cd Plate”と入力し、ディレクトリ Plate へ移動する。  
64 “gfortran plate\_four\_w.f90 plate\_main2\_w.f90”または“gfortran p\*w.f90” などと入力し、 2  
65 つのプログラムをコンパイルする。



```
C:\windows\system32\cmd.exe
C:\Users\yearth\Documents\fort>echo off
Microsoft Windows [Version 10.0.17134.765]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\yearth\Documents\fort>cd Plate

C:\Users\yearth\Documents\fort\Plate>dir
ドライブ C のボリューム ラベルは Windows です
ボリューム シリアル番号は F28A-58F8 です

C:\Users\yearth\Documents\fort\Plate のディレクトリ

2019/06/12 13:56 <DIR>          .
2019/06/12 13:56 <DIR>          ..
2019/06/12 13:56             964 plate_four_w.f90
2019/06/12 13:55             2,757 plate_main2_w.f90
2019/06/12 13:56             337 pl_para2_w.dat
                3 個のファイル          4,058 バイト
                2 個のディレクトリ 188,311,654,400 バイトの空き領域

C:\Users\yearth\Documents\fort\Plate>gfortran plate_four_w.f90 plate_main2_w.f90

C:\Users\yearth\Documents\fort\Plate>_
```

- 66  
67  
68 (7) “dir” と入力し、実行形式ファイル a.exe があるか確認する。  
69 (8) “a.exe” と入力し、プログラムを実行する。  
70  
71

72 10.2 計算結果の表示

73 gnuplot を使用して結果を可視化する.

74

75 (1) “dir” と入力し, 出力ファイルを確認する.

```
C:\windows\system32\cmd.exe
C:\Users\yearth\Documents\fort\Plate>a.exe
density of the mantle (kg/m**3) = 3300.0000000000000
specific heat (J/kg) = 1200.0000000000000
thermal diffusivity (m^2/s) = 9.9999999999999995E-007
age of plate (Ma) = 160.00000000000000
surface temperature (deg. C) = 0.0000000000000000
mantle temperature (deg. C) = 1300.0000000000000
number of depth interval = 150
plate thickness (km) = 300.00000000000000
output file name = plate_A160_L300.data
Note: The following floating-point exceptions are signalling: IEEE_UNDERFLOW_FLAG IEEE_DENORMAL

C:\Users\yearth\Documents\fort\Plate>dir
ドライブ C のボリューム ラベルは Windows です
ボリューム シリアル番号は F28A-58F8 です

C:\Users\yearth\Documents\fort\Plate のディレクトリ

2019/06/12 13:58 <DIR> .
2019/06/12 13:58 <DIR> ..
2019/06/12 13:57 368,006 a.exe
2019/06/12 13:58 8,154 plate_A160_L300.data
2019/06/12 13:56 964 plate_four_w.f90
2019/06/12 13:55 2,757 plate_main2_w.f90
2019/06/12 13:56 337 pl_para2_w.dat
5 個のファイル 380,217 バイト
2 個のディレクトリ 188,308,836,352 バイトの空き領域

C:\Users\yearth\Documents\fort\Plate>
```

76

77

78 (2) 同じディレクトリで “gnuplot” と入力して gnuplot を起動する

79 (3) 以下のように入力する.

80 plot 'plate\_A020\_L300.dat' with line

81 replot 'plate\_A040\_L300.dat' with line

82 ...

```
C:\windows\system32\cmd.exe - gnuplot
2019/06/12 13:57 368,006 a.exe
2019/06/12 14:03 8,154 plate_A020_L300.data
2019/06/12 14:04 8,154 plate_A040_L300.data
2019/06/12 14:05 8,154 plate_A080_L300.data
2019/06/12 14:03 8,154 plate_A160_L300.data
2019/06/12 13:56 964 plate_four_w.f90
2019/06/12 13:55 2,757 plate_main2_w.f90
2019/06/12 14:05 336 pl_para2_w.dat
8 個のファイル 404,678 バイト
2 個のディレクトリ 188,296,757,248 バイトの空き領域

C:\Users\yearth\Documents\fort\Plate>gnuplot

G N U P L O T
Version 5.2 patchlevel 7 last modified 2019-05-29

Copyright (C) 1986-1993, 1998, 2004, 2007-2018
Thomas Williams, Colin Kelley and many others

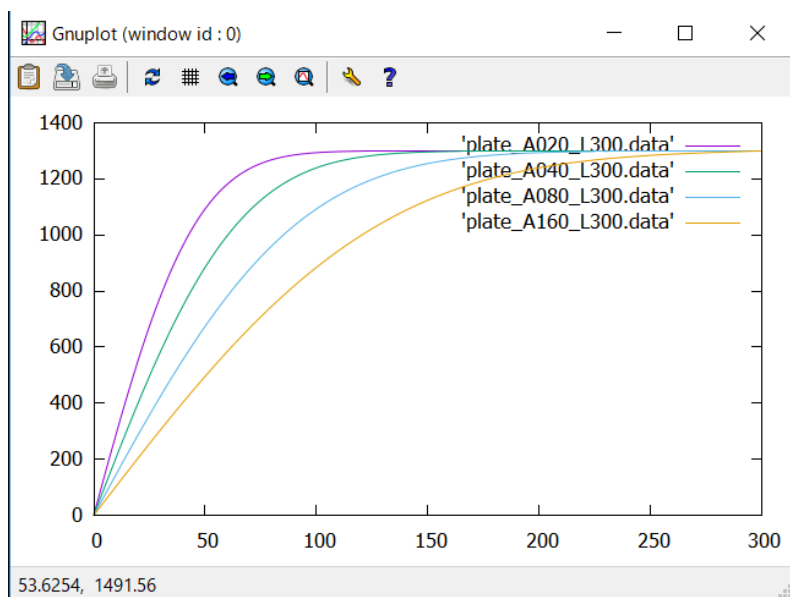
gnuplot home: http://www.gnuplot.info
faq, bugs, etc.: type "help FAQ"
immediate help: type "help" (plot window: hit 'h')

Terminal type is now 'wxt'
Encoding set to 'sjis'.
gnuplot> plot 'plate_A020_L300.data' with line
gnuplot> replot 'plate_A040_L300.data' with line
gnuplot> replot 'plate_A080_L300.data' with line
gnuplot> replot 'plate_A160_L300.data' with line
gnuplot>
```

83

84

85 (4) 以下のような図が出たら完成である.



86

87

## 88 プログラムリスト

89 プログラムのほか、パラメータを入力するためのデータファイルが必要である。それぞれのフ  
90 ァイルを同じディレクトリに置いて、コンパイル、実行すること。

91

### 92 メインプログラム plate\_main2\_w.f90

93

94 ! \*\*\*\*\*

95 ! \* \*

96 ! \* 1-D Half-space cooling model for plate thermal evolution \*

97 ! \* \*

98 ! \*\*\*\*\*

99

100 program plcm

101

102 implicit none

103

104 real(8):: rho, Cp, ak, D

105 real(8):: age, T0, TM

106 real(8):: zk, z, zmax, dz, dzk, t

107 real(8):: hpl, si\_ex

108 integer:: mz, j, ndeg

109 character(40):: apara

110 character(30):: fname

111

112 real(8),allocatable:: Tmp(:)

113

114 ! \*\*\*\*\* read parameters \*\*\*\*\*

115

116 ! Open file

117

118 open ( 10, file = 'pl\_para2\_w.dat' )

119

120 ! read file

```
121
122     read (10,*) apara
123     read (10,*) rho
124     write (6,*) apara, ' = ',rho
125
126     read (10,*) apara
127     read (10,*) Cp
128     write (6,*) apara, ' = ',Cp
129
130     read (10,*) apara
131     read (10,*) D
132     write (6,*) apara, ' = ',D
133
134     read (10,*) apara
135     read (10,*) age
136     write (6,*) apara, ' = ',age
137
138     read (10,*) apara
139     read (10,*) T0
140     write (6,*) apara, ' = ',T0
141
142     read (10,*) apara
143     read (10,*) TM
144     write (6,*) apara, ' = ',TM
145
146     read (10,*) apara
147     read (10,*) mz
148     write (6,*) apara, ' = ',mz
149
150     read (10,*) apara
151     read (10,*) zmax
152     write (6,*) apara, ' = ',zmax
153
```



```

154     read (10,*) apara
155     read (10,*) fname
156     write (6,*) apara, ' = ', fname
157
158 ! convert unit km -> m, Ma -> sec
159
160     t    = age * 1.0d6 * 365.2422d0 * 24.0d0 * 3600.0d0
161     hpl  = zmax * 1.0d3
162
163 ! depth interval
164
165     dz   = hpl / dble( mz )
166     dzk  = dz / 1.0d3
167
168 ! thermal conductivity
169
170     ak = rho * Cp * D
171
172 ! allocate dimension variables
173
174     allocate ( Tmp( 0:mz ) )
175
176 ! maximum degree of Fourier series
177
178     ndeg = mz / 2
179 !
180 ! ***** calculate temperature *****
181 !
182
183 ! at the surface
184
185     Tmp(0) = T0
186     Tmp(mz) = TM

```

```

187
188 ! at z = dz to zmax-dz ( j=1,mz-1 )
189
190     do j = 1,mz-1
191         z = dz * dfloat( j )
192
193 ! Fouries series of saw-tooth function and decay with time
194
195     call     fourexp( si_ex, z, t, D, hpl, ndeg )
196
197 ! composite the steady-state and varying temperatures
198
199     Tmp(j) = T0 + ( TM-T0 ) * ( z/hpl + si_ex )
200
201 !     write (6,*) z(j),Tmp(j),si_ex
202
203     end do
204 !
205 ! ***** Output results to a file
206 !
207
208 ! Open file
209
210     open ( 20, file = fname )
211
212 ! write file
213
214     do j = 0,mz
215         zk = dzk * dfloat( j )
216         write (20,*) zk, Tmp( j )
217     end do
218
219 ! close files

```

```

220
221     close (10)
222     close (20)
223
224     stop
225
226     end program plcm
227
228     サブルーチンプログラム plate_four_w.f90
229
230     !
231     ! ***** Fourier series of saw-tooth function for PLCM *****
232     !
233     subroutine fourexp( si_ex, z, t, D, hpl, ndeg )
234
235         implicit none
236
237         real(8):: si_ex
238         real(8):: z, t
239         real(8):: D, hpl
240         real(8):: s1, sall
241         real(8):: pi
242         integer:: ndeg
243         integer:: i, n
244         real(8),allocatable:: f(:)
245
246     ! ***** allocate *****
247
248         allocate ( f( ndeg ) )
249
250     ! ***** pi *****
251
252         pi = 4.0d0 * atan( 1.0d0 )

```

```

253
254 ! **** calculate  $\sin(n\pi z/H)\exp(-(n\pi/H)^2Dt)$  ****
255
256     do i = 1,ndeg
257         f(i) = 2.0d0 / ( dfloat(i) * pi )           &
258         &      * sin( dfloat(i) * pi * z / hpl )   &
259         &      * exp( - ( dfloat( i ) * pi / hpl )**2 * D * t )
260     end do
261
262 ! **** summation ****
263
264     s1 = 0.0d0
265     do i = 1,ndeg
266         s1 = s1 + f(i)
267     end do
268
269     si_ex = s1
270
271     return
272
273     end subroutine fouexp
274
275 パラメータデータファイル pl_para2_w.dat
276
277 'density of the mantle (kg/m**3)'
278 3300.0d0
279 'specific heat (J/kg)'
280 1200.0d0
281 'thermal diffusivity (m^2/s)'
282 1.0d-6
283 'age of plate (Ma)'
284 040.0d0
285 'surface temperature (deg. C)'

```

```

286  0.0d0
287  'mantle temperature (deg. C)'
288  1300.0d0
289  'number of depth interval'
290  150
291  'plate thickness (km)'
292  300.0d0
293  'output file name'
294  'plate_A040_L300.data'

```

295

296 1次元熱伝導方程式の数値解を求めるプログラムである。入力されたインターバルで深さを  
 297 変えながら温度を計算する。プログラムは2つの部分に分けられており、フーリエ級数を含む  
 298 級数を計算する部分をサブルーチン、その他をメインルーチンにしている。それぞれを1つの  
 299 ファイルにしている。

300

301 主プログラム(メインルーチン)

302 program plcm: program 文。プログラム名を宣言する。

303 implicit none: 宣言文。変数をすべて明示的に宣言することを宣言している。

304 real(8):: rho,...: 宣言文。rho 以下が8バイト(倍精度)実数変数であることを宣言。

305 integer:: mz,...: 宣言文。mz 以下が4バイト整数であることを宣言。

306 character(40):: apara: 宣言文。apara が40バイト文字変数(ASCII文字40字)であることを  
 307 宣言。

308 real(8),allocatable:: Tmp(:): 宣言文。Tmp が倍精度実数であるとともに、動的割り付け配  
 309 列変数であることを宣言。

310 open ( 10, file = 'pl\_para2\_w.dat' ): 補助出入力(open)文。物理パラメータが書かれたフ  
 311 ァイル pl\_para2\_w.dat を開いて装置番号10番に関連づける。

312 read (10,\*) apara: 入力文。ファイルの1行目を文字列として apara に読み込む。文字列は  
 313 クォーテーションマークで囲わねばならない。この read 文が終わると、改行して次の行へ移  
 314 動する。

315 read (10,\*) rho: 入力文。ファイルの2行目の数値を倍精度実数変数 rho に読み込む。

316 write (6,\*) apara, ' = ', rho: 出力文。apara の値と“=”と rho の値を装置番号6番の装置(標  
 317 準出力装置:画面)に1行に出力する。

318 t = age \* 1.0d6 \* 365.2422d0 \* 24.0d0 \* 3600.0d0: 代入文。Ma (百万年)を秒に直す

319  $dz = hpl / mz$  : 代入文. 計算点の間隔数  $mz$  から深さの間隔  $dz$  を計算する. 計算点のインデ  
320 ックス  $j$  は  $0$  から始まるので, 計算点は全部  $mz+1$  では個ある.  
321 `allocate ( Tmp( 0:mz ) )` : 割り付け文. `Tmp` の配列の範囲が  $0$  から  $mz$  であるとして, メモリ  
322 の領域を `Tmp` 用に  $(mz+1) \times 8$  バイト分割り付ける.  
323  $ndeg = mz / 2$  : 代入文. 離散フーリエ級数の次数の最大値を  $mz/2$  に設定する.  
324 `Tmp(0) = T0` : 代入文. 表面の点の温度を代入している.  
325 `Tmp(mz) = TM` : 代入文. マントルあるいはプレート底の点の温度を代入している.  
326 `do j = 1,mz-1 ~ end do` : `do` ループ. 計算点のインデックスには  $j$  を使用しているが, ループ  
327 のインデックスとしてもこれを使用している. 深さを変化させながら温度を計算する.  
328  $z = dz * dfloat( j )$  : 代入文. インデックス  $j$  に対応した深さを計算している.  $dz$  は深さの  
329 間隔である.  
330 `call fourexp( si_ex, z, t, D, hpl, ndeg )` : `call` 文. 級数部分を計算するサブルーチンを  
331 呼び出す. ( )の中はサブルーチンとの間でローカル変数の値を受け渡すための引数が入る. こ  
332 こでは `si_ex, z, t, D, hpl, ndeg` の6つである. `si_ex` 以外は前に定義されている. つまり,  
333 この値をサブルーチンに値を引き継ぐ. `si_ex` は前に定義されていない変数で, サブルーチン  
334 `fourexp` で計算した結果が返って来る.  
335  $Tmp(j) = T0 + ( TM-T0 ) * ( z/hpl + si_ex )$  : 代入文. 定常解と級数の結果を足し合わせ  
336 て温度を計算する.  
337 `open ( 20, file = fname )` : `open` 文. 出力先ファイルを開く. “`read (10,*) fname`”で読み  
338 込んだファイル名のファイルを新規作成する, あるいは開く.  
339 `do j = 0,mz ~ end do` : `do` ループ. 結果をファイルに書くための `do` ループである.  
340 `write (20,*) zk, Tmp( j )` : 出力文. `zk` と `Tmp( j )` の値をファイルを装置番号 `20` のファイ  
341 ルに1行として書き込む.  
342 `close (10)` : 補助出入力(`close`)文. ファイルに EOF(end of file)を書き, ファイルを閉じる.  
343  
344 副プログラム(サブルーチン)  
345 `subroutine fourexp( si_ex, z, t, D, hpl, ndeg )` : `subroutine` 文. サブルーチンの始ま  
346 りを表す. ( )の中は引数が入る. 引数は `call` 文と数や型が合っていないと行かない. それぞ  
347 れの変数はプログラム単位ごとに定義されている. コールされたときには, 引数の値は, 値を  
348 メインルーチンからもらっている. リターンのときは逆に, メインルーチンへ変数の値を返し  
349 ている. コールされたときにはメインルーチン側で `z, t, D, hpl, ndeg` の値は決まっている  
350 が, `si_ex` は不定である. サブルーチンで計算した結果が `si_ex` に代入される.

351 real(8):: z, t : 宣言文. 変数はプログラム単位ごとに決まる(ローカル変数)なので, プログ  
 352 ラム単位ごとに宣言する必要がある.  
 353 pi = 4.0d0 \* atan( 1.0d0 ) : 代入文. 円周率を計算する  
 354 f(i) = 2.0d0 / ( dfloat(i) \* pi )...& : 代入文. 級数項を計算する. 最後の&は次の行へ継  
 355 続することを表す.  
 356 & \* sin( dfloat(i) )...& : 継続行. 先頭の&は省略できるが継続行であることをわかりやす  
 357 くするために書いている. この行も最後に&があり, 継の行へ継続する.  
 358 return : return 文. サブルーチンから呼び出したプログラムへ復帰する. 引数の所に入ってい  
 359 る変数の値がすべて呼び出したプログラムへ引き渡される (つまり, サブルーチンで変えてし  
 360 まった値はコールした側でも変化する). これによって, メインルーチンで si\_ex が確定した値  
 361 となる.  
 362 end subroutine fourexp : end subroutine 文. 副プログラムの終わりを表す.  
 363  
 364 パラメータファイル  
 365 奇数行はその下の値の意味を表すコメント行, 偶数行がデータとなっている.  
 366 上から, マントルの密度(kg m<sup>-3</sup>), 定圧比熱(J kg<sup>-1</sup>), 熱拡散率(m<sup>2</sup> s<sup>-1</sup>), プレートの年代(Ma),  
 367 地表温度(°C), マントルの温度(°C), 計算点の間隔数(個), 計算する最大の深さ(km), 出力ファ  
 368 イル名, である.  
 369