[1] 課題内容

ファイル(cabinet)から歴代首相の在籍期間を入力し、長期政権順に並び替えディスプレイに表示するプログラムを作成せよ。表示時には入力データの内容に加えて通算期間(日)も同時に出力すること

[2] プログラム

cabinetプログラム

```
--------------------------------------------------------------------------------
/***************************************
  Program    : cabinet.c
  Student-ID : 045713C
  Author     : OSHIRO,Kazuya
  Date       : 04/07/13
  Comment    : 歴代首相の在籍期間と通算日数
***************************************/

#include <stdio.h>

#define FILE_NAME "cabinet"      /* FILE_NAMEを文字列cabinetを定義する */


struct date{                     /* dateの構造体 */
   int year;                     /* 整数型のyearという変数 */
   int month;                    /* 整数型のmonthという変数 */
   int day;                      /* 整数型のdayという変数 */
};

struct cabinet{                  /* cabinetの構造体 */
   char name[20];         /* char型のnameサイズは20 */
   struct date term[2];          /* date構造体の1次元配列 */
   int days;                     /* 整数型のdaysという変数 */
};

int period (struct date x, struct date y);
   /* period関数の宣言、引数に構造体dateをとる */
int dll (struct date x);            /* dll関数の宣言 */
int uruu (int y);                   /* uruu関数の宣言 */
void sort (struct cabinet x[], int n);      /* sort関数の宣言 */

main()
{
   struct cabinet sc[50]; /* 構造体を50含む配列を宣言 */
   FILE *fp;                     /* ファイル変数fpの宣言 */
   int i, n = 0;          /* 整数型の変数iとnを宣言 */
   char buf[256], *buf_ptr;      /* char型の配列bufとポインタbuf_ptrを宣言 */

   fp = fopen (FILE_NAME,"r");   /* FILE_NAME(cabinet)を開く */
   printf ("INPUT FILE NAME : %s--------------------------¥n",FILE_NAME);
   while (fgets(buf, 256, fp) != NULL){
     /* 文字配列bufに最大256文字、をfp(つまりはcabinet)から1列、読み込む */
     /* その値がNULLつまり読み込む文字列がなくなったときwhile文を抜ける */
        for (buf_ptr = buf; *buf_ptr; buf_ptr++)
            printf("%c", *buf_ptr);
```

```
        /* buf_ptrをbufの先頭アドレスに初期化、buf_ptrの値がNULLになるまで */
    /* buf_ptrのアドレスを一つずらし(インクリメント)、その後配列の値を */
    /* 出力する。 */
         }
    puts ("-------------------------------------------------");

    fclose (fp);                    /* fpつまりcabinetを閉じる */

    fp = fopen (FILE_NAME,"r");
    while (fgets (buf, 256, fp) != NULL){
        sscanf (buf, "%s%d%d%d%d%d%d",
                 sc[n].name,
                 &sc[n].term[0].year,&sc[n].term[0].month,&sc[n].term[0].day,
                 &sc[n].term[1].year,&sc[n].term[1].month,&sc[n].term[1].day);
        /* bufから指定した変数を指定した書式で読み込む */
        sc[n].days = period (sc[n].term[0],sc[n].term[1]);
        /* sn[n].daysは通算日数を表している。period関数の値をとる。 */
        n++;
    }

    sort (sc, n);           /* sort関数を呼び出す */
    for (i=0; i<n; i++) {  /* iがn以上になるまでインクリメントしプリントする */
      printf("%-20s %4d/%2d/%2d - %4d/%2d/%2d %5d¥n",
               sc[i].name,
               sc[i].term[0].year, sc[i].term[0].month, sc[i].term[0].day,
               sc[i].term[1].year, sc[i].term[1].month, sc[i].term[1].day,
               sc[i].days);
    }
    puts("-------------------------------------------------");
    fclose(fp);
}


int period (struct date x, struct date y) /* period関数の定義 */
{
    int k, d = 0;

    for (k = x.year; k < y.year; k++){
        if (uruu(k))
              d += 366;             /* uruu関数の値が真のときdに366を足す */
        else
              d += 365;             /* 偽のとき365を足す */
    }
    return (d + dll(y) - dll(x) + 1);
    /* 通算日数を表す、その値を呼び出し元へ返す */
}

int dll (struct date x)             /* dll関数の定義 */
{
    int i, sum;
    int month[] = {0,31,28,31,30,31,30,31,31,30,31,30,31};
    /* この配列は月の日数を表している */
    if (uruu(x.year))
         month[2] = 29;             /* uruu関数が真のときmonth[2]の値を29とする */
    for (sum = 0, i= 1; i < x.month; i++)
         sum += month[i]; /* それぞれの月の日数を足していく */

    return(sum + x.day);
    /* sum + x.day でその年の初めからdate xの日までの通算日数が出る。その値を返す */
```

```
}

int uruu (int y)          /* uruu関数の定義 */
{
  return(((y % 4 == 0) && (y % 100 != 0)) || (y % 400 == 0));
  /* 閏年の計算、4の倍数で100の倍数でないものまたは400で割れる数のとき真(1)を返す */
}

void sort (struct cabinet x[], int n) /* sort関数の定義 */
{
   struct cabinet w;              /* 構造体の変数wを宣言 */
   int i, j;                      /* 整数型の変数iとjを宣言 */
   for (i=1; i<n; i++){
       w = x[i];          /* wにx[i]の値を保持しておく */
       for(j = i - 1; j >= 0 && w.days > x[j].days; j--)
           x[j + 1] = x[j];   /* x[j+1]、つまり次の配列にx[j]をコピーする */

       x[j + 1] = w;
       /* コピーにしたことによって作り出したx[j+1]にwを代入する */
   }
}
```

---

[3] 実行結果

cabinet result

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
[Kazuya-OSHIRO:~/prog1/report/#7] j04013% ./cabinet2
INPUT FILE NAME : cabinet---------------------------
J-Koizumi 2001 4 26 2004 7 15
Y-Mori 2000 4 5 2001 4 26
K-Obuchi 1998 7 30 2000 4 5
R-Hashimoto 1996 1 11 1998 7 30
T-Murayama 1994 6 30 1996 1 11
T-Hata 1994 4 28 1994 6 30
M-Hosokamwa 1993 8 9 1994 4 28
K-Miyazawa 1991 11 5 1993 8 5
T-Kaifu 1989 8 10 1991 11 5
S-Uno 1989 6 2 1989 8 9
N-Takeshita 1987 11 16 1989 6 2
Y-Nakasone 1982 11 27 1987 11 6
Z-Suzuki 1980 7 17 1982 11 27
M-Ohira 1978 12 7 1980 6 12
T-Fukuda 1976 12 24 1978 12 7
T-Miki 1974 12 9 1976 12 14
K-Tanaka 1972 7 7 1974 12 9
E-Satoh 1964 11 9 1972 7 7
-------------------------------------------------
E-Satoh              1964/11/ 9 - 1972/ 7/ 7  2798
Y-Nakasone           1982/11/27 - 1987/11/ 6  1806
J-Koizumi            2001/ 4/26 - 2004/ 7/15  1177
R-Hashimoto          1996/ 1/11 - 1998/ 7/30   932
K-Tanaka             1972/ 7/ 7 - 1974/12/ 9   886
Z-Suzuki             1980/ 7/17 - 1982/11/27   864
T-Kaifu              1989/ 8/10 - 1991/11/ 5   818
T-Miki               1974/12/ 9 - 1976/12/14   737
T-Fukuda             1976/12/24 - 1978/12/ 7   714
K-Miyazawa           1991/11/ 5 - 1993/ 8/ 5   640
```

```
K-Obuchi              1998/ 7/30 - 2000/ 4/ 5   616
N-Takeshita           1987/11/16 - 1989/ 6/ 2   565
T-Murayama            1994/ 6/30 - 1996/ 1/11   561
M-Ohira               1978/12/ 7 - 1980/ 6/12   554
Y-Mori                2000/ 4/ 5 - 2001/ 4/26   387
M-Hosokamwa           1993/ 8/ 9 - 1994/ 4/28   263
S-Uno                 1989/ 6/ 2 - 1989/ 8/ 9    69
T-Hata                1994/ 4/28 - 1994/ 6/30    64
--------------------------------------------------
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

[4] 考察

a. 関数の説明

＊＊＊ファイル関数について＊＊＊

ファイル関数はファイルを扱うために使う関数である。ファイル関数が使用する構造体および関数の宣言は標準インクルードファイル<stdio.h>に格納されている。ファイル変数の宣言は次のようになる。

```
FILE *file-variable;
```

ファイルは使う前にオープンしなければならない。ファイルをオープンするにはfopenを使う。その形式は次のようになる。

```
file-variable = fopen(name,mode);
```

ここでのfile-variableはファイル変数をnameは開くファイル名を示している。modeはファイルを開くときに読み込みなのか書き込みなのかなどを指定する。今回のプログラムでは"r"が使われており、これは読み込みようとしてファイルを開くことを表している。ちなみにファイルが見つからない場合NULL(エラー)を返す。
この他にmodeには
"w"----------書き込みモードでファイルを開く。
ファイルがない場合新しく作成し、存在している場合その内容を破壊する。
"a"----------追加モードでファイルを開く。
ファイルがない場合は作成する。
"r+"----------既存ファイルを対象に、読み込み/書き込みの両方のモードで開く。
ファイルがない場合はエラーを返す。
"w+"----------ファイルを作成し、読み込み/書き込みの両方のモードで開く
ファイルが存在している場合はその内容を破壊する。
"a+"----------読み書き/書き込みモードの両方のモードで開く。
ファイルが存在する場合は追加、ない場合は作成する。
などがある。

＊＊＊fgets関数について＊＊＊

前回の使ったときはキーボードからの読み込みだったが今回はファイルからの読み込みになったので軽く考察を。
fgets()の一般的な形式は

```
fgets (name, size, file);
```

nameは読み出した文字列を格納する文字列配列。sizeは文字列のサイズ、fgetsは一行を読み出すか（size-1）文字の数の文字を読み込む。fileは入力元になるファイル。ここをstdinに変えるとキーボードからの入力(標準入力)になる。

＊＊＊sscanf関数について＊＊＊
標準関数sscanf()は、文字列を指定した書式に従って、指定した変数に読み込む関数である。
一般的なsscanf()の形式は

```
sscanf (line, format, &variable1,&variable2,...)
```

のようになっている。lineは読み込む文字列、formatは%dや%sのような書式指定。それにあわせてvariable1やvariable2などに値が代入されることになる。

b. 例題プログラムの間違い

今回の難所だったプログラムの間違い。
いくつか自分で在任通算日数をだしてみることで閏年の計算がおかしいことに気づきました。
それと友達の助言により間違いが見つかりました。
今回の重大な間違いは各月の日数をきめる配列にある。

```
static int month[] = {0,31,28,31,30,31,30,31,31,30,31,30,31};
```

この初めのstatic、これにより閏年が条件のif文により一度month[2] = 29が実行されるとその後に呼び出される値はすべて先ほど保持された29となってしまうのです。よってstaticをつけずにおけば問題は解決というわけです。重大だけどとても気づきにくい問題でした。

c.構造体について

今回のプログラムは構造体が大きな役割を果たしているので前回のレポートでも書いたがいくつか付け加えて書いておきます。
構造体の各フィールドにアクセスするための構文

```
viriable.field
```

例えば今回の場合さらに構造体を配列とおいてsn[n].dayやsn[n].yearなどのように表記している。さらに今回の構造体は少し複雑なので図をいれてみました。

d. プログラムの流れ

＊＊＊dll関数編＊＊＊

この関数は任期の初めの年月日、あるいは終わりの年月日を引数にとり、そのそれぞれの年の初めから任期開始の日、任期終了の日までの日数を表す。初めに整数型の変数iとsumを宣言する。このiは月を表し、sumは月の合計日になる。配列month[]は各月の日数を表しておりmonth[0]は0、month[1]は31という風になっている。month[2]は特別でuruu関数にてその年が閏年ならばmonth[2]は29になる。for文によりiが引数の月を過ぎるまで各月の日数がsumに足されていく。for文をぬけreturnで値を返すのだがちゃんとした日数を出すためにsumに引数の任期初日、もしくは任期最終日を足した値を返す。

＊＊＊period関数編＊＊＊

引数に構造体の値を二つとる。struct date xは任期の初め、struct date yは任期終わりの年月日を示している。kは始めにx.yearつまりは任期の初めの年を表しておりこの年がy.year、任期終わりの年でなければkを一年ずらす。そしてif文の条件式に入るここではuruu関数を呼び出してその年が閏年であるかないかの真偽を出す。それにより閏年ならdに366を足し、閏年でない年の場合には365を足す。この作業をkがy.yearになるまで繰り返すことにより、年単位での通算日数ができてくる。

　最後にreturnで値を返すのだが`(d + dll(y) - dll(x) + 1)x` とかかれている。この値は下の図を見ればわかるようにperiodのfor文で数えすぎた分のdll(x)を引き、足りない部分のdll(y)を足す。そしてこの値には任期初日が含まれていないため1を足す。

＊＊＊uruu関数編＊＊＊

この関数は引数に整数型の値をとる。これは「年」のデータを表しており、この年が閏年かどうかを判断してその真偽を呼び出し元へ返す。閏年の条件は4の倍数の年でかつ100の倍数でない年、もしくは400の倍数の年。これをかくと`((y % 4 == 0) && (y % 100 != 0)) || (y % 400 == 0)`となる。

＊＊＊sort関数編＊＊＊

この関数はソート済みの集合体の中からソートする値を適切な位置に挿入していくことを行う。まず、引数に構造体の値と整数型の値をとる。そして構造体の変数wと整数型のi,jを宣言する。for文ではiがn未満なら1を足す(インクリメント)、wにx[i]の値を代入することで目的の値を保持する働きがある。さらにその中のfor文にてjをiの一つ前の値に初期化、jは0以上でかつw.days(x[i]の時の任期通算日数)がx[j].days(x[j]の時の任期通算期間)より大きいときjから1を引き(デクリメント)、x[j+1]にx[j]をコピーする。これを繰り返すことにより対象の値が入る場所を決める。for文をぬけてx[j+1]にwの値を代入する。初めのfor文が偽になるつまりiがnになるまでこれらの作業を繰り返すことですべてを昇順にソートすることができる。

＊＊＊main関数(全体)編＊＊＊
この前に考察したすべての関数を使いmain関数は動かすことになる。まず始めに構造体の配列sc[],ファイル変数fp,整数型の変数i,n,char型の配列buf[]とポインタbuf_ptrを宣言する。cabinetファイルをfopenで開く。printfで文字列を表示する。while文とfor文でファイル内の文字を表示していく。そしてファイルを閉じる。
　その後、再びcabinetを開き、while文とsscanfを使うことでbufにある値を構造体の配列に読み込んでいく。次に、sc[n].days(通算日数)にperiod関数を呼び出した値を代入していく。最後のn++で配列を次に移していくような形になる。
　その後にsort関数を呼び出す。そのことで構造体配列に通算日数の昇順にそれぞれの値が入ることになる。それからfor文とprintf()を使ってソートされた文字列と数字をそれぞれ出力していく。

[5] 感想

今回のプログラミング、色々あってかなり遅くに始めたんですがそれが大失敗でした。こんなに難しいとは・・・。プログラムを初めから追っていくと途中でこんがらがる。さらに問題点を見つけるのもかなり地道な作業になって時間がかかりました。それでもって考察も尋常じゃなく時間がかかりました。多少疲れました今回は。まぁ、とりあえずあと残りは後一つ、最後だけにこれまでで一番難しそうです。心して取りかかろうと思います。

参考文献および使用コンパイラ

C実践プログラミング 第3版　[Steve Oualline 著]
DACCHO'S Software Factory
[http://homepage1.nifty.com/daccho/index.htm]
先生のPDFファイルおよび授業ノート
gcc - GNU project C and C++ compiler