Cプログラミング

大レポート

課題3

指名：　　　樋口陽祐

所属：　基幹理工学部

学籍番号：1w1523146

出題日：　　1月6日

提出日： 　1月19日

担当者：青柳滋己先生

課題内容：

簡易整数電卓の作成

考察：

昨年のACM-ICPC国際大学対抗プログラミングコンテストの国内予選のProblem Cで、特別な式を計算する計算機を作成する問題があった[1]。特別な式とは通常の計算式と違い、加算演算子と乗算演算子、そしてピリオドのみを並べたものとなっている。この問題を解いた経験から、本課題も容易に取り組むことができた。演算子、記号には優先順位が有り、入力された式を左から順に計算しても正しい計算結果とならない。よって、式を分割し、優先順位の高い演算子を含む式を計算していく必要がある。本プログラムでは、これを再帰処理によって行うことにした。

解法：

split関数は、文字列strをn文字目でstr1とstr2に分ける。poww関数は累乗（a^b）を計算した値を返す。fact関数は階乗(a!) を計算した値を返す。check関数は、strのn文字目が括弧（“（　）”）内に存在しているかどうかを調べ、ある場合は1ない場合は0を返す。solve関数では、入力された式の計算をする。47~50行目では、文字列strの文字数を数える。51~64行目では、strに + または – があるかどうかを調べ、括弧外にある場合は演算子の前後で文字列を分け、それぞれの文字列に対し再度solve関数を呼び出し、足したまたは引いた結果を返す。また、 - が文字列の最初であった場合、2文字目の整数に-1をかけて返す。66~75行目では、strに \* または / があるかどうかを調べ、括弧外にある場合は演算子の前後で文字列を分け、それぞれの文字列に対し、再度solve関数を呼び出し、掛けたまたは割った結果を返す。77~83行目では、strに ^ があるかどうか調べ、括弧外にある場合は演算子の前後で文字列を分け、それぞれの文字列に対し再度solve関数を呼び出し、それぞれの結果の累乗を返す。84~90行目では、strに ! があるかどうか調べ、括弧外にある場合は演算子直前の文字に対し再度solve関数を呼び出し、それに対する階乗を返す。92~104行目では、strに ( が存在するか調べ、ある場合、) がある場所を調べ、括弧内の文字列に対し再度solve関数を呼び出す。106~109行目では、strを数字に変換する。main関数では、文字列の入力を行い、最初の整数が負である場合、最初の文字に0を加える操作を行う。そして、文字列strを引数としたsolve関数の結果を出力する。

感想：

課題４ができなかったら課題３をやると決めていたが、どちらともやることにした。計算の仕方よりかは、文字列の扱い方を考えるのに時間をかけた。前期のCプログラミング入門では、今回のような「楽しい」問題がなかったため物足りなかったが、後期では大レポートなどで良い経験を積めてよかった。自分の傾向として、私の書くプログラムは整理整頓されていなく、だらだらと長く書いてしまうため、改善してゆきたい。

参考文献

C/C++リファレンス

<<http://www.cppll.jp/cppreference/>>

C言語関数辞典

　<<http://www.c-tipsref.com/>>

[1]icpc国内予選 2015の過去問

<http://icpc.iisf.or.jp/past-icpc/domestic2015/contest/all_ja.html#section_C>

四則演算の優先順位

<http://www.geisya.or.jp/~mwm48961/math/operator1.htm>