

基礎プログラミング演習 II 教材

■ どのようにしてプログラムを書くか

やみくもに作業してはいけません。「プログラミング」つまり「どのようにプログラムを組むか」ということ自体を感じてください。

□ ハンドトレスを厭わない

実際にプログラムを走らせる前に、(コンピュータを使わず) 手作業でプログラムの動きを追ってみましょう。#3 で秒数を指定すると何時何分何秒に相当するか計算するプログラムがありました。その課題をプログラミングしているとき、例えばある秒数を仮定して、手計算でそれが何時何分何秒になるか計算し、書いたプログラムが実際に期待通りの結果を出すか確認する、といった作業をきちんとやっていますか？

適当に思いついた秒数を入れてみて、何時何分何秒と表示された時点で動いたと思っていませんか？それが正しい時分秒であるか分からないが、どうやら動いたらしい、では全く駄目です。

□ テストランをする

プログラムが完成する前にでも、完成後でも、とにかく何度も実行してそれが正しく動作する(ハンドトレスの結果と一致する)ことを確認しましょう。

どこかの秒数(12345 秒でも、36000 秒でも)が何時何分何秒と計算して確認し、+10 秒したら確かにそのようになるか、+100 秒したら 1 分 40 秒進んだ結果が出るか、確認すべきです。

□ テストプログラムを作るのは当たり前

今回、受講生は「時分秒から秒数を求める」プログラムを既に持っています。課題 2. はその逆を求めるものなので、これを利用してより適切なテストランができるはずです。

プログラムの動作を確認するために、テストプログラムを別に作成することは普通のことです。テストデータを作るためのプログラムを何本も作るのは当然のことです。

提出することがない(評価の対象とならない)プログラムを作ることを避けて、最小の作業量で要求に対応しようとするのはよほど慣れたプログラマがやることで、学ぶ過程にある人がやって良いことではありません。どうすれば正しいプログラムを作ることが出来るか試行錯誤することに価値があります。

(「正しいプログラムの開発法」は無い(工学的に確立されていない)のですから)

□ 美意識をもって書く

同じ時分秒を出すために幾つもの計算法と表記が考えられると思います。秒を求めるのに

$12345\%3600\%60$ と書いた人は、

$12345 \% 3600 \% 60$ と書いた方がより見やすい(誤読がない)かもしれない、

$(12345 \% 3600) \% 60$ と書いた方がプログラマの意図が汲みやすいかもしれない

(あるいはその逆)といったことを考えるべきです(*)。分を求めるのに

$12345 / 60 \% 60$ と

$12345 \% 3600 / 60$ と、

どちらが分かりやすい(プログラマの意図が見える)でしょう。

プログラムはどのようにでも書けます。つまりどれがより良いプログラムなのか検討する必要があります。そして「簡潔で、無駄が無く、しかも分かりやすい」は時としてトレードオフの関係にあります。プログラマにはどこにバランスポイントを置くかがセンスとして求められるのです。

(* そもそも 3600 の剰余に対して 60 の剰余を取り直すことは無駄なのですが。)