令和７年度 応用プログラミング試験問題

出題者: 平居
試験時間: 50 分
注意事項: 応用プログラミング講義資料、過去に自分が作成したコードの閲覧、筆記用具の持ち込みは可。その他 web サイトの検索や AI の使用は不可。学生証を通路側に置くこと。答案 (exam1.c、exam2.c、main.c、multi.c、Makefile) は yutaka.hirai@koeki-u.ac.jp へメール添付で提出する。メールは Koeki Mail を用い、件名は「応用プログラミング試験解答 [学生番号]」とし、メール本文に氏名と学籍番号を記載すること。試験終了後に答案提出時間を設ける。答案の再提出は不可とする。完成したコードはコンパイルして実行できることを確認してから提出すること。部分点を与えるので、途中のファイルも提出すること。

以下のプログラムを C 言語を用いて作成せよ。出力時は文末に必ず改行のエスケープシーケンスを入れること。

第 1 問　「応用プログラミングを受講しました。単位ください。私が解けた問題数：」と表示させた後にユーザーに 0 から 3 の整数値を入力させ、0 が入力された場合は「残念！また来年受けてね。」、1 が入力された場合は「まずまずかな、単位はあげよう。」、2 が入力された場合は「よくできました。君には優を与えよう。」、3 が入力された場合は、「素晴らしい！来年は TA をお願いしたいな。」、それ以外の数字が入力された場合は「0 から 3 を入力してね。」と表示するプログラム exam1.c を作成せよ (30 点)。

第 2 問　以下の表は架空の学生 3 名の情報リテラシー及びデータリテラシーの点数である。

氏名	情報リテラシー	データリテラシー
Taro	84	64
Hanako	91	82
Takashi	73	79

この点数から、各科目の GP (グレード・ポイント) と各学生の GPA (グレード・ポイント・アベレージ) を計算し、左から順に学生名、情報リテラシーの GP 、データリテラシーの GP、GPA を記したファイル gpa.txt を出力するプログラム exam2.c を作成せよ。GP は整数型、GPA は浮動小数点数型小数点第一位までを出力せよ。ただし、GP は以下のように定義する。

点数	GP
90 点以上	4
80 点以上 90 点未満	3
70 点以上 80 点未満	2
60 点以上 70 点未満	1
60 点未満	0

尚、GPA の定義は以下の通りである。簡単のため、各科目の単位数は 2 とする。

$$\mathrm{GPA} = \frac{(\text{各科目の GP} \times \text{単位数}) \text{ の合計}}{\text{単位数の合計}}.$$

(30 点)。

第 3 問　変数 x、y について以下の微分方程式を 2 次のルンゲ・クッタ法という近似解法で数値的に解く。

$$\frac{\mathrm{d}y}{\mathrm{d}x} = xy.$$

(1) ディレクトリ exam3 を作成し、その下に double 型の変数 x と y の掛け算をした値を返す関数 multi を記述したファイル multi.c を作成せよ (10 点)。

(2) 微分方程式を解くためには、初期値を設定する必要がある。ここでは、初期値を $x = 0.0$、$y = 4.0$ とする。main 関数を収めたファイル main.c をディレクトリ exam3 内に作成し、変数 x と変数 y を宣言してそれぞれ 0.0 と 4.0 で初期化せよ (5 点)。

(3) 微分方程式を数値的に解くには、ある範囲を複数回分割して近似解を求める。ここでは、x が 0 から 3 までの範囲を 100 分割する。等間隔で分割する場合、x の初期値を x_0 とした場合、i ステップ目の x の値、x_i は次のようになる。

$$x_i = x_0 + ih,$$

ただし、h は定数で、今回の場合は $h = (3.0 - 0.0)/100 = 0.03$ になる。i ステップ目の y_i の値が分かっている時、$i+1$ ステップ目の y の値、y_{i+1} を求めるには次のようにする。まず、x_i で解となる関数の接線をステップの半分、すなわち $h/2$ まで引く。次に、もう 1 回微分方程式の右辺を評価し、元の点 (x_i, y_i) から再度直線を引く。これにより、y_{i+1} を求める。式に書くと以下のようになる。

$$\begin{aligned} k_1 &= y_i + \frac{h}{2} f(x_i, y_i), \\ y_{i+1} &= y_i + hf\left(x_i + \frac{h}{2}, k_1\right), \\ x_{i+1} &= x_i + h. \end{aligned}$$

ただし、$f(x, y)$ は微分方程式の右辺であり、(1) で定義した関数に対応する。これを 100 回の繰り返し処理を行う for ループの中に記述し、x と y の値を逐次画面に出力するプログラムを main.c 内に作成せよ (20 点)。

(4) main.c と multi.c の依存関係を記述した Makefile を作成し、make でコンパイルして実行せよ (5 点)。

— 以上 —