次の問８は必須問題です。必ず解答してください。

問８　次のアルゴリズムの説明及びプログラムを読んで，設問に答えよ。

方程式の解の一つを求めるアルゴリズムである。任意に定めた解の予測値から始めて，計算を繰り返しながらその値を真の値に近づけていく。この方法は，ニュートン法と呼ばれる。

〔アルゴリズム１の説明〕

３次方程式 $a_3x^3+a_2x^2+a_1x+a_0=0$ の解の一つを，次の手順で求める。

(1)　解の予測値 x，係数 a_3，a_2，a_1，a_0 を読み込む。

(2)　$3\times a_3$ の値を b_2 に，$2\times a_2$ の値を b_1 に，$1\times a_1$ の値を b_0 に，それぞれ求める。

(3)　次の①～④の処理を一定の回数繰り返す。

①　$a_3x^3+a_2x^2+a_1x+a_0$ の値を求め，これを f とする。

②　$b_2x^2+b_1x+b_0$ の値を求め，これを d とする。

③　x，f，d の値を印字する。

④　$x-\dfrac{f}{d}$ の値（解の一つにより近い値となる）を求め，これを新たな x とする。

プログラム１は，このアルゴリズム１を実装したものである。

〔アルゴリズム２の説明〕

アルゴリズム１を一般化して，n 次方程式 $a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0=0$ の解の一つを，次の手順で求める。

なお，方程式によっては解が求められない場合がある。

(1)　次数 n，解の予測値 x，係数 a_n，a_{n-1}，…，a_1，a_0 を読み込む。

(2)　$n\times a_n$ の値を b_{n-1} に，$(n-1)\times a_{n-1}$ の値を b_{n-2} に，…，$2\times a_2$ の値を b_1 に，$1\times a_1$ の値を b_0 に，それぞれ求める。

(3)　次の①～④の処理を一定の回数繰り返す。

①　$a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0$ の値を求め，これを f とする。

②　$b_{n-1}x^{n-1}+b_{n-2}x^{n-2}+\cdots+b_1x+b_0$ の値を求め，これを d とする。

③　x，f，d の値を印字する。

④　$x-\dfrac{f}{d}$ の値（解の一つにより近い値となる）を求め，これを新たな x とする。

プログラム２は，このアルゴリズム２を実装したものである。

〔プログラム1〕

（行番号）

```
○主プログラム: プログラム1
○整数型: i
○実数型: d, f, x
○実数型: a3, a2, a1, a0, b2, b1, b0

・read(x, a3, a2, a1, a0)    /* x, a3〜a0 の値を読み込む。 */
・b2 ← 3.0 × a3
・b1 ← 2.0 × a2
・b0 ← a1
■ i: 1, i ≦ 100, 1          /* 繰返し回数は 100 回とする。 */
│ ・f ← ((a3 × x + a2) × x + a1) × x + a0
│ ・d ←  (b2 × x + b1) × x + b0
│ ・print(x, f, d)           /* x, f, d の値を印字する。 */
│ ・x ← x − f ÷ d
■
/* プログラム1の終わり */
```

〔プログラム2〕

（行番号）

```
○主プログラム: プログラム2
○整数型: i, k, n             /* 1 ≦ n ≦ 9 とする。 */
○実数型: d, f, x
○実数型: a[10], b[10]        /* 添字は 0 から始まる。 */

・read(n, x)                 /* n, x の値を読み込む。 */
■ k: n, k ≧ 0, −1           /* k を n, n−1, …, 0 として繰り返し, */
│ ・read(a[k])               /*  係数 a[k]の値を順に読み込む。 */
■
┌ - - - - - - - - - - - - ┐
     手順(2)の処理
└ - - - - - - - - - - - - ┘
■ i: 1, i ≦ 100, 1          /* 繰返し回数は 100 回とする。 */
│ ┌ - - - - - - - - - - - - ┐
│    手順(3)の①と②の処理
│ └ - - - - - - - - - - - - ┘
│ ・print(x, f, d)           /* x, f, d の値を印字する。 */
│ ・x ← x − f ÷ d
■
/* プログラム2の終わり */
```

設問　次の記述中の[　　　]に入れる正しい答えを，解答群の中から選べ。

(1)　解の予測値 $x=2.5$，係数 $a_3=1$，$a_2=-3$，$a_1=-1$，$a_0=3$ を与えて，3次方程式 $x^3-3x^2-x+3=0$ の解の一つを求める（解は3，1，−1）。プログラム1をある処理系で実行した結果，図1に示すとおり解の一つである $x=3$ が近似的に得られた。

（行番号）	x	f	d
1	2.500000	−2.625000	2.750000
2	3.454545	4.969947	14.07438
3	3.101425	8.741682(−1)	9.247965
4	3.006900	5.548452(−2)	8.082941
5	3.000035	2.833717(−4)	8.000425
6	3.000000	7.527369(−9)	8.000000
7	3.000000	0.000000	8.000000
8	3.000000	0.000000	8.000000

注1　数値の後の(−k)は，$\times10^{-k}$ を示す。例えば，5.548452(−2)は，5.548452×10^{-2}，すなわち0.05548452を表す。

注2　表示は有効数字7けた（8けた目を四捨五入）

図1　プログラム1の印字結果

この印字結果の行番号6，7の x の値（網掛けの部分）はいずれも3.000000である。行番号6，7を印字した時点で変数 x に保持されていた実際の値をそれぞれ x_6，x_7 で表すと，[　a　]。

なお，この処理系では，実数型は2進数の浮動小数点形式であって，有効けた数は10進数で十数けた程度であることが分かっている。

(2)　プログラム2では，係数 a_k（$k:n$，$n-1$，…，1，0）の値を配列aの要素a[k]に，b_k（$k:n-1$，$n-2$，…，1，0）の値を配列bの要素b[k]に，それぞれ図2のように格納している。

要素番号	0	1	2	…	n−1	n	…
配列a	a_0	a_1	a_2	…	a_{n-1}	a_n	…

要素番号	0	1	…	n−3	n−2	n−1	…
配列b	$1\times a_1$	$2\times a_2$	…	$(n-2)\times a_{n-2}$	$(n-1)\times a_{n-1}$	$n\times a_n$	…

図2　係数 a_k，b_k の値の格納

プログラム2の行番号9～11は，アルゴリズム2の手順(2)の処理である。この部分のプログラムは，次のようになる。

〔プログラム2の一部〕

（行番号）

9 ■ k: n, k ≧ 1, −1　/* kをn, n−1, …, 1として繰り返す。 */

10 │ ・$\boxed{\text{ b }}$

11 ■

また，行番号13～18は，アルゴリズム2の手順(3)の①と②の処理である。プログラム1では，例えば f の値 $a_3x^3+a_2x^2+a_1x+a_0$ を求める式を，

f ← ((a3 × x + a2) × x + a1) × x + a0

と変形して，演算回数を減らす工夫をしている。この部分にも同様の工夫をすると，プログラムは次のようになる。

〔プログラム2の一部〕

（行番号）

13 ・f ← a[n] × x + a[n−1]

14 ・d ← $\boxed{\text{ c }}$

15 ■ k: n−2, k ≧ 0, −1　/* kをn−2, n−1, …, 0として繰り返す。*/

16 │ ・f ← f × x + a[k]

17 │ ・d ← d × x + $\boxed{\text{ d }}$

18 ■

(3) 次数 $n=4$，係数 $a_4=1$，$a_3=-8$，$a_2=24$，$a_1=-32$，$a_0=16$ として，4次方程式 $x^4-8x^3+24x^2-32x+16=0$ の解を求める（4個の解がすべて2）。解の予測値を $x=2.00001$ として，ある処理系でプログラム2を実行したところ，図3に示すとおりの印字結果となった。

(行番号)	x	f	d
1	2.000010	−3.552714(−15)	3.996803(−15)
2	2.888899	6.243232(−1)	2.809423
3	2.666674	1.975398(−1)	1.185225
4	2.500006	6.250281(−2)	5.000169(−1)
5	2.375004	1.977628(−2)	2.109446(−1)

図3　プログラム2の印字結果

この印字結果の行番号2では，x の値（網掛けの部分）が解である2から遠ざかってしまっている。その原因を調べるため，f を求める式に実際の数値を当てはめて，

$$\underbrace{(((1.0\times 2.00001-8.0)\times 2.00001+24.0)\times 2.00001-32.0)\times 2.00001}_{\text{(A)}}+16.0$$

として，(A) の部分の中間結果を印字するプログラムを作り，同じ処理系で実行した。印字結果は −16.00000 であり，正確な値 −15.99999999999999999999 と有効数字7けたで一致した。しかし，行番号1で印字された f の値は，正確な値である 10^{-20}（印字の表記では 1.000000(−20)）とは異なっている。

これらのことから判断して，(A) の部分では演算の過程で $\boxed{\quad e \quad}$ が徐々に累積し，(A) の計算結果に16.0を加算するときに，けた落ちが発生したと考えられる。

aに関する解答群

ア　$x_6=x_7$ である　　　イ　$x_6\neq x_7$ である

ウ　$x_6=x_7$ とも $x_6\neq x_7$ ともいえない

bに関する解答群

ア　b[k−1] ← (k−1) × a[k]　　　イ　b[k−1] ← k × a[k]

ウ　b[k] ← k × a[k+1]　　　エ　b[k] ← (k+1) × a[k+1] c，dに関する解答群

ア　b[k−1]	イ　b[k]
ウ　b[k+1]	エ　b[n−1]
オ　b[n−1] × x	カ　b[n−1] × x ＋ b[n−2]

eに関する解答群

ア　けたあふれ	イ　けた落ち
ウ　指数下位けたあふれ	エ　丸め誤差