令和３年度　豊橋技術科学大学第３年次入学者選抜学力検査問題

専　門　科　目　（３：情報・知能工学）

注　意　事　項

1　試験開始の合図まで，この問題冊子と解答用紙を開いてはいけません。

2　問題冊子の枚数は表紙，草稿用紙を含めて７枚です。

3　問題冊子とは別に解答用紙が３枚あります。解答は用紙の裏面にまわってはいけません。

4　問題は３問あります。全問解答してください。

5　解答にかかる前に，すべての解答用紙の所定の箇所に受験番号を記入してください。

6　解答は必ず各問題別の解答用紙の所定の欄に記入してください。

7　落丁，乱丁，印刷不鮮明の箇所などがあれば，ただちに申し出てください。

8　問題冊子の余白は草稿用として使用しても構いません。

9　試験終了時刻まで退出してはいけません。

（草　稿　用　紙）

[1] 微分方程式を解くことに関して，以下の空欄 $\boxed{\quad ア \quad}$ ～ $\boxed{\quad セ \quad}$ に当てはまる数値または数式を解答用紙の該当する箇所に記述せよ。ただし，同じ記号の空欄には同じものが入る。

(1) y を t の関数とするとき，1次微分方程式 $\dfrac{dy}{dt} - 2t = 0$, $y(0) = 1$ の解は $y = \boxed{\quad ア \quad}$ である。

(2) y を t の関数とするとき，2次微分方程式 $\dfrac{d^2y}{dt^2} + 4y = 0$, $y(0) = 1$, $y'(0) = 4$ の解は $y = \cos 2t + \boxed{\quad イ \quad}$ である。

(3) y_1 と y_2 を t の関数とするとき，2元連立2次微分方程式 $\dfrac{d^2y_1}{dt^2} + 5y_1 - 2y_2 = 0$, $\dfrac{d^2y_2}{dt^2} - 2y_1 + 2y_2 = 0$, $y_1(0) = 1$, $y_2(0) = 2$, $y_1'(0) = -2\sqrt{6}$, $y_2'(0) = \sqrt{6}$ がある。この連立微分方程式は $\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$, 行列 $\boldsymbol{A} = \begin{bmatrix} -5 & 2 \\ 2 & -2 \end{bmatrix}$ とおくと，$\boldsymbol{y}''=\boldsymbol{A}\boldsymbol{y}$と書ける。この微分方程式を，行列 $\boldsymbol{A}$ がある行列 $\boldsymbol{P}$ を用いて $\boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$ と対角行列化できることを利用して解いていく。ただし，λ_1 と λ_2 は行列 $\boldsymbol{A}$ の固有値である。$\lambda_1 > \lambda_2$ とすると，$\lambda_1 = \boxed{\quad ウ \quad}$, $\lambda_2 = \boxed{\quad エ \quad}$ となる。λ_1 に対応する固有ベクトルの一つは $\begin{bmatrix} 1 \\ \boxed{\quad オ \quad} \end{bmatrix}$，$\lambda_2$ に対応する固有ベクトルの一つは $\begin{bmatrix} 1 \\ \boxed{\quad カ \quad} \end{bmatrix}$ となるので，$\boldsymbol{P} = \begin{bmatrix} 1 & 1 \\ \boxed{\quad オ \quad} & \boxed{\quad カ \quad} \end{bmatrix}$ を用いて行列 $\boldsymbol{A}$ を対角行列化できる。ここで，$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \boldsymbol{P}^{-1} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$ とおくと，

$$\frac{d^2}{dt^2}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \boldsymbol{P}^{-1}\frac{d^2}{dt^2}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \boldsymbol{P}^{-1}\boldsymbol{A}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

となる。つまり，$\dfrac{d^2x_1}{dt^2} = \boxed{\quad ウ \quad} \times x_1$, $\dfrac{d^2x_2}{dt^2} = \boxed{\quad エ \quad} \times x_2$ となる。この解を

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_1 \cos\left(\boxed{\quad キ \quad}\right) + b_1 \sin\left(\boxed{\quad キ \quad}\right) \\ a_2 \cos\left(\boxed{\quad ク \quad}\right) + b_2 \sin\left(\boxed{\quad ク \quad}\right) \end{bmatrix}$$

とおくと $\begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \boldsymbol{P}\begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix}$ および $\begin{bmatrix} y_1'(0) \\ y_2'(0) \end{bmatrix} = \boldsymbol{P}\begin{bmatrix} x_1'(0) \\ x_2'(0) \end{bmatrix}$ から $a_1 = \boxed{\quad ケ \quad}$, $b_1 = \boxed{\quad コ \quad}$, $a_2 = \boxed{\quad サ \quad}$, $b_2 = \boxed{\quad シ \quad}$ となる。

最終的に，この微分方程式の解は $\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \boxed{\quad ス \quad} \\ \boxed{\quad セ \quad} \end{bmatrix}$ となる。

[2] 以下の文章および図中にある空欄 $\boxed{\text{ア}}$ ～ $\boxed{\text{シ}}$ に入れるのに適当なもの一つを図5にある選択肢のうちから選び，その番号((1)～(50))で答えよ。同じ番号を何度使用してもよい。正しい選択肢が複数ある場合はそのうち一つを答えればよい。

また，空欄 $\boxed{\text{A}}$ ～ $\boxed{\text{C}}$ は，その値を求めよ。

素数を求めるプログラムについて考える。$2, 3, 5, \cdots$ と，2からはじめて，最初の100個の素数を求めていくことにする。できるだけ効率のよいプログラムをつくることを目標に，まず基本的な処理手順を作成し，それを改良していくことにする。

図1に，最初に考えた基本的な処理手順を示す。ここで，変数 k はそれまでに得られた素数の個数，変数 x は素数の候補である。なお，以下の処理手順において，記号 ← は ← の左側の変数に ← の右側の定数，変数，あるいは式の計算結果の値を代入することを意味する。

```
(1-1)   k ← 0
(1-2)   x ← 2
(1-3)   k [ ア ] 100 である間，以下を繰り返す
(1-4)       もし x が素数であるならば
                x を印字する
                k ← [ イ ]
(1-5)       x ← x + 1
(1-6)   (1-3) の繰り返しはここまで
```

図 1: 最初に考えた処理手順

図1の(1-4)の部分で行う素数であるかどうかの判定を詳しく書いたのが，図2である。なお，mod は割り算の余りを求める演算子で，例えば，(7 mod 3) は 1 である。

```
(2-1)   f ← 2
(2-2)   ( x mod f ) [ ウ ] 0 である間，以下を繰り返す
(2-3)       f ← f + 1
(2-4)   (2-2) の繰り返しはここまで
(2-5)   もし f [ エ ] x であるならば，x は素数である
          そうでなければ，x は素数でない
```

図 2: 素数判定の処理手順

このプログラムの改良のために，図 2 の実行効率を考えてみよう。図 2 の処理手順では，変数 x がどのような値であっても (2-2) 以降の繰り返し処理による素数判定を実行している。特に，x が素数であった場合，(2-3) の代入処理を［オ］回実行することになる。しかし，2 以外の正の偶数は素数ではないので，繰り返し処理による素数判定は変数 x が奇数である場合にのみ行えばよい。また，x が奇数かつ f が偶数ならば (2-2) の条件は必ず成り立つので，変数 f が奇数の場合のみこの条件を調べればよい。このことを利用して，図 2 の処理手順を改良したのが，図 3 である。

```
(3-1)   もし x = 2 であるならば，x は素数である
          そうでなければ，(3-2) 以降を実行する
(3-2)   もし x mod [ カ ] = 0 であるならば，x は素数でない
          そうでなければ，(3-3) 以降を実行する
(3-3)   f ← [ キ ]
(3-4)   ( x mod f ) [ ウ ] 0 である間，以下を繰り返す
(3-5)       f ← f + 2
(3-6)   (3-4) の繰り返しはここまで
(3-7)   もし f [ エ ] x であるならば，x は素数である
          そうでなければ，x は素数でない
```

図 3: 改良した素数判定の処理手順

図 2 から図 3 への改良で，どれほど効率がよくなるか，ひとつの実行例で考えてみよう。変数 x として，素数である 83 が与えられていたとする。(2-3) の代入処理は［A］回行われるのに対し，(3-5) の代入処理は［B］回行われることになり，およそ［ク］倍の効率向上が期待できる。

このプログラムのさらなる改良を考えてみよう。図 3 の処理手順では，変数 x が素数であった場合，(3-5) の代入処理を［ケ］/ 2 回実行することになる。ところで，変数 x が素数でない，すなわち，x に 1 と x 以外の約数がある場合，その約数のうちの一つは必ず x の平方根に等しいかより小さい値となる。このことを利用して，図 2 の処理手順を改良したのが，図 4 である。

```
(4-1)   もし x = 2 であるならば，x は素数である
          そうでなければ，(4-2) 以降を実行する
(4-2)   もし x mod [ カ ] = 0 であるならば，x は素数でない
          そうでなければ，(4-3) 以降を実行する
(4-3)   f ← [ キ ]
(4-4)   (( f^2 [ コ ] x ) かつ ( (x mod f) [ ウ ] 0 )) である間，以下を繰り返す
(4-5)       f ← f + 2
(4-6)   (4-4) の繰り返しはここまで
(4-7)   もし f^2 [ サ ] x であるならば，x は素数である
          そうでなければ，x は素数でない
```

図 4: さらに改良した素数判定の処理手順

図 3 から図 4 への改良で，どれほど効率がよくなるか，先の実行例で考えてみよう。変数 x として，素数である 83 が与えられていたとする。(4-5) の代入処理は $\boxed{\text{C}}$ 回行われることになる。したがって，図 3 と比べておよそ $\boxed{\text{シ}}$ 倍，また図 2 と比べるとおよそ $\boxed{\text{ク}}$ × $\boxed{\text{シ}}$ 倍，の効率向上が期待できる。ただし，実際の処理においては，(4-4) における判定処理が (3-4) のそれに比べ複雑になっていることを差し引いて考えないといけない。

(1)	1	(2)	2	(3)	3	(4)	4	(5)	5
(6)	6	(7)	7	(8)	8	(9)	9	(10)	10
(11)	20	(12)	30	(13)	100	(14)	200	(15)	300
(16)	1000	(17)	2000	(18)	3000	(19)	10000	(20)	20000
(21)	30000	(22)	0	(23)	=	(24)	≠	(25)	<
(26)	>	(27)	≦	(28)	≧	(29)	x	(30)	x+1
(31)	x+2	(32)	x+3	(33)	x-1	(34)	x-2	(35)	x-3
(36)	k	(37)	k+1	(38)	k+2	(39)	k+3	(40)	k-1
(41)	k-2	(42)	k-3	(43)	f	(44)	f+1	(45)	f+2
(46)	f+3	(47)	f-1	(48)	f-2	(49)	f-3	(50)	f^2

図 5: 空欄を埋めるための選択肢

［3］　以下の問いに答えよ。ただし，解答では，論理積には「·」，論理和には「＋」，変数 A の否定には「$\overline{A}$」を用い，真理値表およびカルノー図では真の値は 1，偽の値は 0，ドントケア（出力が 0 でも 1 でも良い）は×を用いて答えよ。また，各ゲート素子は図 1 のように表すとする。

2 入力 AND 素子　2 入力 OR 素子　2 入力 NAND 素子　2 入力 NOR 素子

図 1：ゲート素子

(1)　図 2 に示した NAND 素子で構成される次の論理回路アとイと等価である論理回路を，図 3 に示す (a)〜(f) からそれぞれ一つずつ選べ。

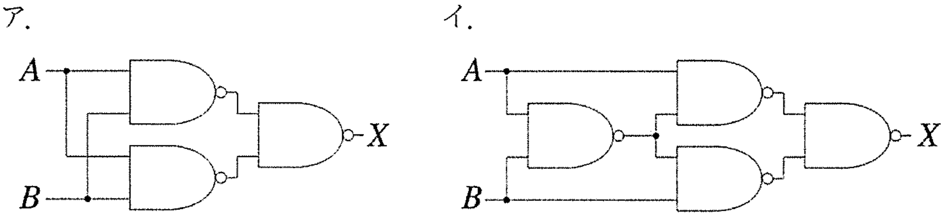

図 2：NAND 素子で構成された論理回路

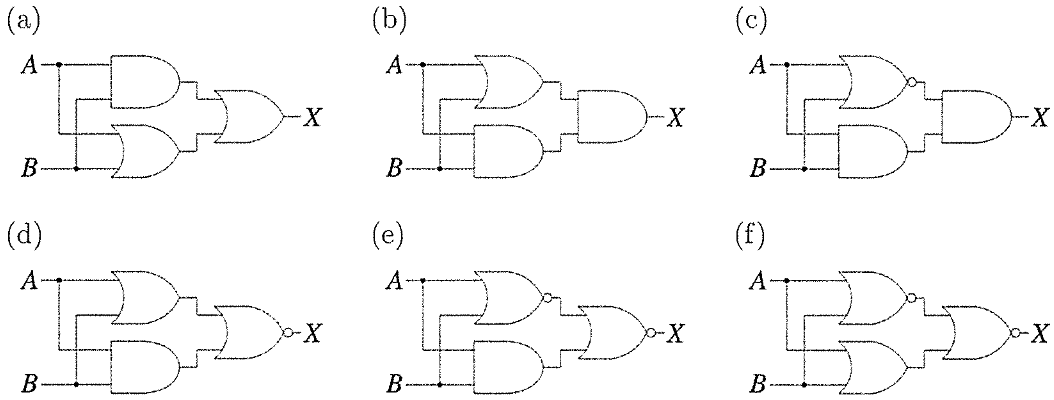

図 3：選択肢

(2)　それぞれ 1 ビットからなる四つの信号 A, B, C, D を入力変数とし，三つの信号 X, Y, Z を出力する回路がある。入力信号は，A が最上位ビット，B が第 2 位ビット，C が第 3 位ビット，D が第 4 位ビット（最下位ビット）を表し，0〜15 の整数値を 2 進数表現で表すものとする（例えば，$ABCD = 1010$ であれば，10 進数の 10 を表す）。出力信号は入力値を 5 で割った時の余りを，入力信号と同様に X を最上位ビット，Y を第 2 位ビット，Z を第 3 位ビット（最下位ビット）とする 2 進数で表すとする。ただし，余りが 4 のときは最上位ビットの X が 1 となり，下位ビットの Y, Z の値はドントケアであるとする。以下の問いに答えよ。

ア.　解答用紙の真理値表を埋めよ。

イ.　解答用紙の出力 Y, Z のそれぞれのカルノー図を埋めよ。

ウ.　出力 Y と Z の論理式を最小積和形で答えよ。ただし，最小積和形とは，複数個の論理積項が論理和で結ばれている論理式において，論理積と論理和の総数が最小になる形式である。