

令和4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11枚です。
- 3 問題冊子とは別に解答用紙が4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。
- 10 問題冊子は持ち帰ってください。

(草稿用紙)

[1] 砂糖が水に溶ける様子を観察する。ここで、砂糖の溶解速度（単位時間あたりの減少量）は、まだ溶けていない量に比例するものとする。いま、砂糖 100 g が最初の 10 秒間で 40 g 溶けたとする。以下の空欄  ~  に当てはまる数式，または正の整数値を解答用紙の該当する箇所に記述せよ。なお ,  に関しては，四捨五入して，整数で答えよ。 については，①から⑧の選択肢から1つ選べ。

(1) 時刻  $t$  で溶けていない砂糖の量を  $Q(t)$  とする。溶解速度は  $Q(t)$  の  $k$  倍 ( $k > 0$ ) とすると，

$$\frac{dQ(t)}{dt} = \text{ア}$$

微分方程式を解くと，定数  $A$  を用いて，

$$Q(t) = A \text{イ}$$

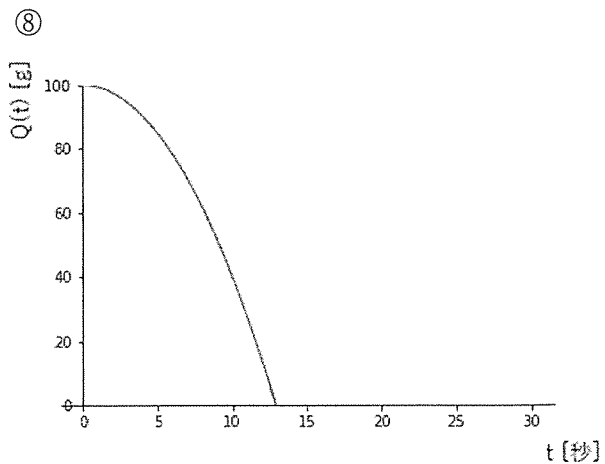
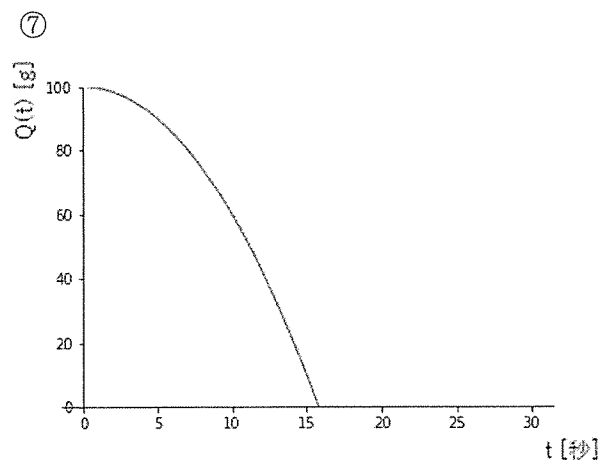
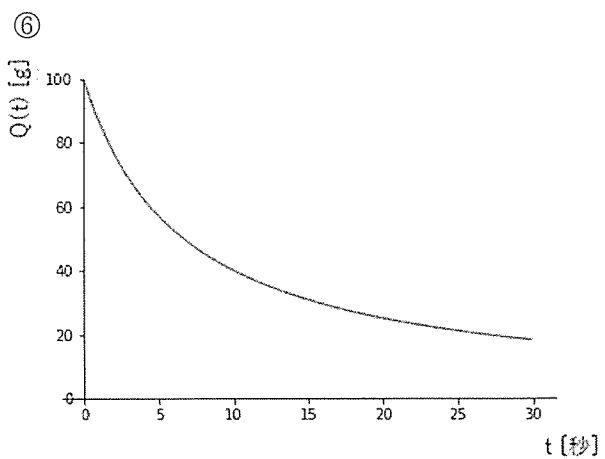
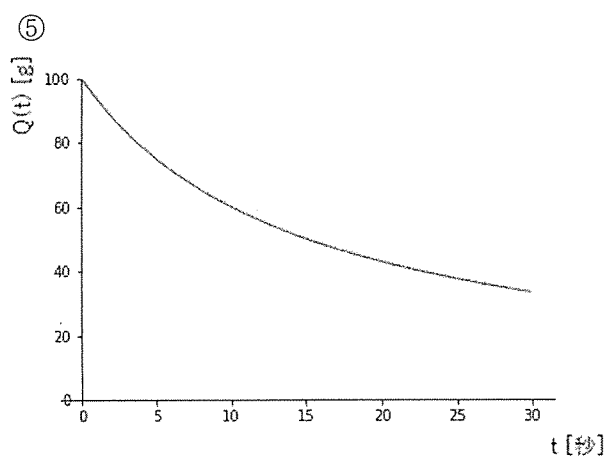
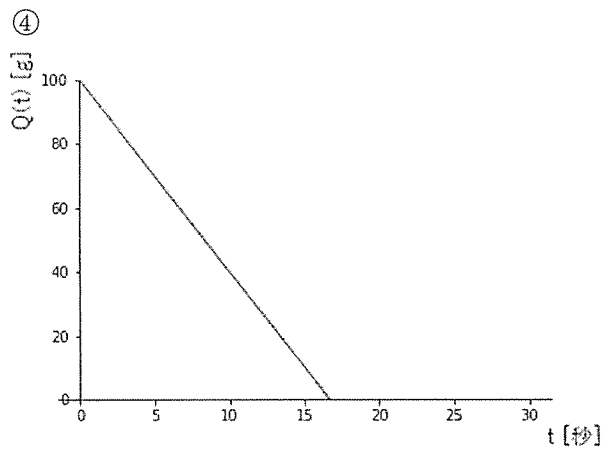
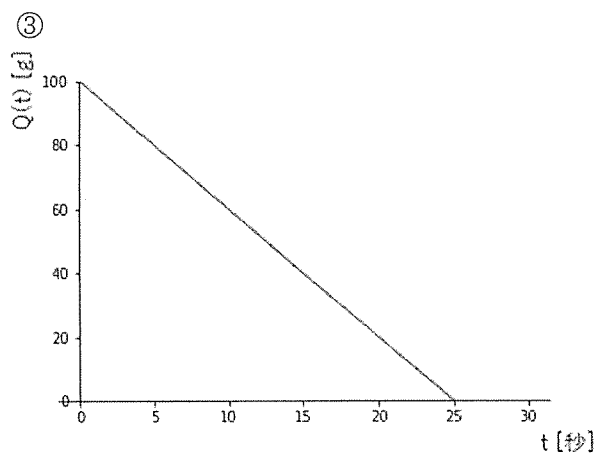
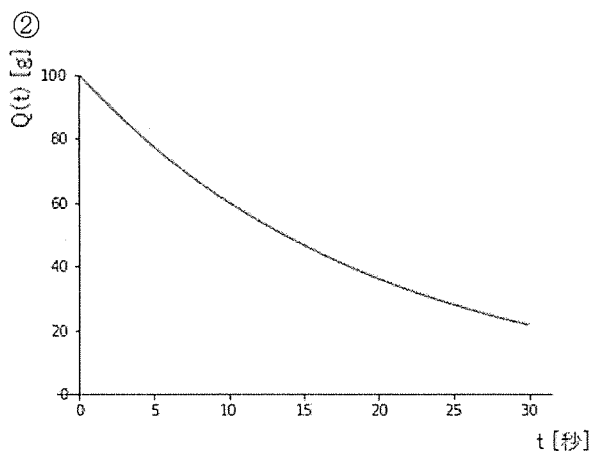
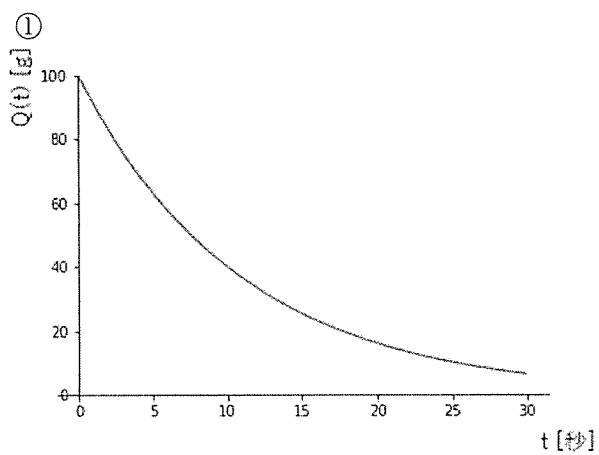
(2)  $t = 0$  秒において  $Q(0) = \text{ウ}$  g,  $t = 10$  秒において  $Q(10) = \text{エ}$  g であることから，

自然対数関数  $\ln$  を用いて， $k = \frac{\text{オ}}{\text{カ}} \ln \frac{\text{キ}}{\text{ク}}$  と求まる。

(3) 最初の 30 秒間に砂糖が溶けた量は  g である。

(4) 砂糖が 64 g 溶けるのは，最初から  秒後である。

(5)  $Q(t)$  の図として正しいのは  である。





(2.6) (2.2)の繰り返しはここまで

- (2) ここでプログラム2の実行効率を考えてみよう。プログラム2の処理手順では、 $i$ を1から $x$ まで処理を繰り返して約数を探している。ここで、 $x$ が $i$ で割り切れる場合、 $i$ が約数であるとともに  も約数であることに着目する。すると、約数を探す際に $x$ まで確認する必要はなく、 を超えない最大の整数までで十分である。そこでプログラム2を改良したのがプログラム3である。

プログラム3:  $x$ の約数をすべて見つけてxDivListに格納するプログラム

- (3.1)  $i \leftarrow 1$   
 (3.2)  $i \leq$   である間、以下を繰り返す  
 (3.3)     もし  $(x \% i) = 0$  ならば  
 (3.4)         xDivListに $i$ と  を追加する  
 (3.5)      $i \leftarrow i+1$   
 (3.6) (3.2)の繰り返しはここまで

この改良でどれだけ効率が良くなるか具体的な実行例で考えてみよう。変数 $x$ として1024が与えられていたとする。(2.3)の処理は  回行われるのに対して、(3.3)の処理は  回行われることになる。処理の回数は  $1 /$   になる。このとき、プログラム3終了時にxDivListの要素数は  になる。

- (3) 次に、さらに計算量を抑えて最大公約数を得るために、「ユークリッドの互除法」というアルゴリズムを用いることにする。ユークリッドの互除法を次に示す。ここで $a \geq b$ とする。

ユークリッドの互除法

$a \% b = 0$  のときに最大公約数は $b$ である。  
 $a \% b = r_1$  ( $r_1 \neq 0$ ) のとき、 $a, b$ の最大公約数は  $b, r_1$ の最大公約数と等しい。  
 $b \% r_1 = r_2$  ( $r_2 \neq 0$ ) のとき、 $b, r_1$ の最大公約数は  $r_1, r_2$ の最大公約数と等しい。  
 $r_1 \% r_2 = r_3$  ( $r_3 \neq 0$ ) のとき、 $r_1, r_2$ の最大公約数は  $r_2, r_3$ の最大公約数と等しい。  
 ⋮  
 ⋮  
 ⋮  
 $r_{n-1} \% r_n = 0$  のとき、最大公約数は $r_n$ となる。

ユークリッドの互除法で最大公約数が求まることを証明する。  
 もし  $a \% b = 0$  の場合、 $a \geq b$  であるから最大公約数は $b$ である。

$a \% b \neq 0$  の場合を考える。 $a \% b$  を  $r_1$  とおく。 $a, b$ の最大公約数を $m$ とし、 $b, r_1$ の最大公約数を $M$ とする。 $a$ と $b$ は $m$ および整数 $a', b'$ を用いて以下のように書け

る。

$$a = ma' \quad \text{式 (1)}$$

$$b = mb' \quad \text{式 (2)}$$

ここで、 $a$ を $b$ で割った際の余りが $r_1$ であることから、 $a$ は整数 $k$ を用いて以下のように書ける。

$$a = kb + r_1 \quad \text{式 (3)}$$

式 (3) に式 (1), 式 (2) を代入すると

$$ma' = kmb' + r_1$$

移項して $m$ でまとめると以下となる。

$$m(a' - kb') = r_1$$

ここで $a' - kb'$  は整数であることから、 $r_1$ は $m$ で割り切れることになる。つまり $a, b, r_1$ は $m$ で割り切れる。このとき $M$ は $b, r_1$ の最大公約数であり、 $m$ は $b, r_1$ の公約数であることから、

$$M \boxed{\text{エ}} m \quad \text{式 (4)}$$

という関係が成り立つ。

次に、 $b, r_1$ の最大公約数が $M$ であることから、 $b, r_1$ は整数 $b'', r_1''$ を用いて以下のように書ける。

$$b = Mb'' \quad \text{式 (5)}$$

$$r_1 = Mr_1'' \quad \text{式 (6)}$$

式 (5), 式 (6) を式 (3) に代入すると

$$a = kMb'' + Mr_1''$$

$M$ でまとめると以下となる。

$$a = M(kb'' + r_1'')$$

ここで $kb'' + r_1''$ は整数であることから、 $a$ は $M$ で割り切れることになる。つまり $a, b, r_1$ は $M$ で割り切れる。このとき $m$ は $a, b$ の最大公約数であり、 $M$ は $a, b$ の公約数であることから、

$$M \boxed{\text{オ}} m \quad \text{式 (7)}$$

という関係が成り立つ。したがって、式 (4), 式 (7) を同時に満足するのは

$$m \boxed{\text{カ}} M$$

のときのみである。つまり  $a, b$  の最大公約数  $m$  は  $b, r_1$  の最大公約数  $M$  と等しい。

次に,  $b \% r_1 = r_2$  とおく。  $b, r_1$  の最大公約数と,  $r_1, r_2$  の最大公約数は上記と同じ論理によって等しくなる。同様に  $r_1 \% r_2 = r_3$  とおくと,  $r_1, r_2$  の最大公約数は,  $r_2, r_3$  の最大公約数と等しくなる。

ここでもし  $r_{n-1} \% r_n = 0$  となった場合,  $r_{n-1} > r_n$  であるから最大公約数は  $r_n$  となる。

$r_{n-1}, r_n$  の最大公約数  $r_n$  は,  $r_{n-2}, r_{n-1}$  の最大公約数と等しく, 上記の論理を逆にたどると,  $r_1, r_2$  の最大公約数と等しく,  $b, r_1$  の最大公約数と等しく, 最終的に  $a, b$  の最大公約数と等しいことになる。

以上で証明終了する。

(4) ユークリッドの互除法を用いて最大公約数を求めるプログラムは以下となる。

プログラム 4: ユークリッドの互除法で最大公約数を求めるプログラム

- (4.1)  $r \leftarrow a \% b$   
 (4.2) もし  $r = 0$  ならば  
 (4.3)      $\boxed{\text{キ}}$  を表示して処理を終了する  
 (4.4) そうでないとき,  $a \leftarrow \boxed{\text{ク}}$ ,  $b \leftarrow \boxed{\text{ケ}}$  として (4.1) に戻る

プログラム 4 への改良でどれほど効率が良くなるか, 具体的な実行例で考えてみよう。変数として  $a = 512, b = 200$  が与えられたときに, プログラム 1 およびプログラム 3 を用いた場合と比較する。プログラム 3 では (3.3) の処理は  $a$  に対して  $\boxed{\text{E}}$  回,  $b$  に対して  $\boxed{\text{F}}$  回行われる。プログラム 1 では (1.10) の処理は  $\boxed{\text{G}}$  回行われる。一方でプログラム 4 において (4.2) の処理は  $\boxed{\text{H}}$  回実行される。プログラム 4 で大幅に効率が良くなっていることが分かる。



## 選択肢

|                |                |            |               |
|----------------|----------------|------------|---------------|
| ① 1            | ② 2            | ③ 3        | ④ 0           |
| ⑤ <            | ⑥ >            | ⑦ $\leq$   | ⑧ $\geq$      |
| ⑨ %            | ⑩ =            | ⑪ a        | ⑫ b           |
| ⑬ i            | ⑭ r            | ⑮ aDivList | ⑯ bDivList    |
| ⑰ aDivList(ia) | ⑱ bDivList(ia) | ⑲ ia       | ⑳ ib          |
| ㉑ km           | ㉒ kb           | ㉓ $r_1$    | ㉔ $r_2$       |
| ㉕ $r_3$        | ㉖ $r_n$        | ㉗ GCD      | ㉘ $x/i$       |
| ㉙ $\sqrt{x}$   | ㉚ x            | ㉛ $x/2$    | ㉜ $\log_2(x)$ |

[3] 以下の各問いに答えよ。ただし，2値変数  $x_1, x_2$  に対して，論理積を  $x_1x_2$ ，論理和を  $x_1+x_2$ ， $x_1$  の否定を  $\bar{x}_1$  で表す。真理値表およびカルノー図においては，真の値を1，偽の値を0，ドントケアを  $x$  で表す。ここで，全ての変数（否定でもよい）を各論理積項に含み，それらが論理和で結ばれている形式を主加法標準形と呼び，論理関数  $F(x_1, \dots, x_n)$  に対する双対関数を  $\overline{F(\bar{x}_1, \dots, \bar{x}_n)}$  と定義し， $F^d(x_1, \dots, x_n)$  で表すものとする。

(1) 次の主加法標準形で表されている論理関数について，以下の各問いに答えよ。

$$F(x_1, x_2, x_3) = \bar{x}_1x_2x_3 + x_1\bar{x}_2x_3 + x_1x_2\bar{x}_3 + x_1x_2x_3$$

- ア．解答用紙の真理値表の空欄を埋めよ。
- イ．関数  $F(x_1, x_2, x_3)$  を単純化せよ。
- ウ．関数  $F(x_1, x_2, x_3)$  の双対関数  $F^d(x_1, x_2, x_3)$  を主加法標準形で表せ。

(2) 次の真理値表で表される論理関数について，以下の各問いに答えよ。

| $f_{k-1}$ | $a_k$ | $b_k$ | $f_k$ |
|-----------|-------|-------|-------|
| 0         | 0     | 0     | 0     |
| 0         | 0     | 1     | 1     |
| 0         | 1     | 0     | 0     |
| 0         | 1     | 1     | 0     |
| 1         | 0     | 0     | 1     |
| 1         | 0     | 1     | 1     |
| 1         | 1     | 0     | 0     |
| 1         | 1     | 1     | 1     |

- ア．3つの入力  $f_{k-1}, a_k, b_k$  に対する出力  $f_k$  について，解答用紙のカルノー図の空欄を埋めよ。
- イ．問(2)アのカルノー図を使って単純化を行い，その単純化した式を示せ。

(3) 次の文章は，問(1)および問(2)の論理関数についての説明である。空欄  ~  に対する適切なものを選択肢から一つ選べ。

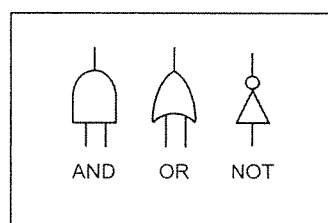
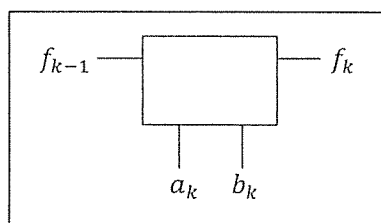
問(1)の論理関数は、3個の1ビットの数の  を行う関数であり、 となる値を出力する。単純化した式より、この関数は  個の2入力ANDゲートと、 個の3入力ORゲートで構成できることがわかる。この関数の双対関数は、元の関数の形と  形となり、このような関数のことを  と呼ぶ。

問(2)の論理関数は、 桁の2進数の  を行う関数である。 $f_{k-1} = 0$ かつ であるとき、出力 $f_k$ は0となり、 $f_{k-1} = 0$ かつ であるとき、出力 $f_k$ は1となる。

選択肢

0, 1, 2, 3, 4, 5, 6, 7, ドントケア,  
 加算, 減算, 乗算, 除算, 比較演算, 一致演算, 多数決演算,  
 同じ, 違う, 反対, 否定, 多数, 少数, 高い, 低い,  
 合成関数, 導関数, 自己双対関数, 一次関数, 自己反双対関数,  
 下位桁, 上位桁, 繰り上がり, 繰り下がり, 比較結果,  
 $a_k = b_k, a_k \neq b_k, a_k < b_k, a_k \leq b_k, a_k > b_k, a_k \geq b_k$

(4) 問(2)の論理関数に対する回路を複数組み合わせさせて、二つの正の $n$ 桁の2進データ  $A = a_{n-1} \dots a_0$  と  $B = b_{n-1} \dots b_0$  に対する  $n$  桁の演算回路を構成することを考える。ここで入力  $f_{k-1}$  には、一つ下の桁の出力  $f_{k-1}$  をそのまま入力するものとする。以下の各問いに答えよ。ただし、最下位桁を除く  $k (\neq 0)$  に対する回路を以下の左のブロック図で表すとする。



- ア. 最下位桁となる  $f_0$  の論理式の単純化した式を示せ。ただし、 $f_{-1} = 0$  とする。
- イ. 問(4)アの結果と上図の回路を組み合わせ、4桁の演算回路を構成し、その回路図を示せ。ただし、上図左のブロック図のほか、上図右の2入力AND, 2入力OR, NOTを使ってよい。
- ウ. 次の  $A$  と  $B$  の組み合わせに対する出力  $f_3$  を解答用紙の表に埋めよ。
  - ①  $A=0101$  と  $B=0101$
  - ②  $A=1010$  と  $B=1011$
  - ③  $A=1101$  と  $B=1001$