

名古屋工業大学

2023年度（令和5年度）

編入学者・転入学者選抜学力検査[問題]

－ 専門試験 －

(情報工学科)

試験日時 2022年6月24日（金）

10:00～12:00

●解答上の注意

- (1) 解答の際、解答用紙のホチキス止めを外してください。
- (2) 配布物は、問題冊子1冊、解答用紙3枚、計算用紙1枚です。
- (3) 解答は各問題番号に対応する解答用紙に解答してください。
- (4) 解答が解答用紙表面に書ききれない場合は、裏面に続いてもよいが、その場合は表面の下側が裏面の上側になるようにし、上側2/3のスペースに解答を収めてください。
- (5) 電卓は使用できません。
- (6) 試験終了後は問題用紙と計算用紙を持ち帰ってください。

問題 1 設問すべてについて解答すること。ただし、回路を図示する場合には、必要に応じて図 1-1 に示す論理記号を用いること。また、A の否定を \bar{A} で表す。

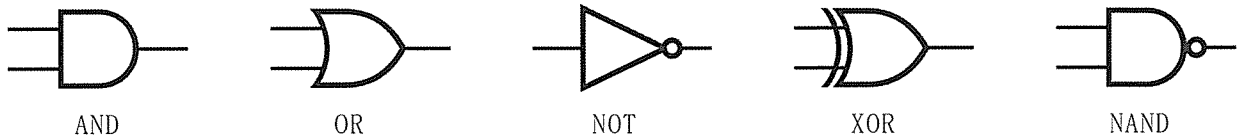


図 1-1：論理記号

I 次の (1)～(2) の問いについて答えよ。

(1) 2 の補数を用いて、以下の 8 ビット 2 進数の演算を行え。途中経過を明記すること。

(a) 01110000 - 01010101

(b) 00001111 - 10101010

(2) A, B, C を入力としたとき、 $F=(A+B) \cdot (B+C)$ を実現する論理回路を 3 個の NAND 素子のみを用いて図示せよ。

II 1 ビット全加算器を、入力 X, Y, 出力 Z, 桁上げ入力 Cin, 桁上げ出力 Cout を持つ図 1-2 で示すとする。次の (1)～(3) の問いについて答えよ。

(1) 1 ビット全加算器の真理値表を表 1-1 の形式で示せ。

(2) 1 ビット全加算器を 2 つ用いて 2 ビット全加算器を作成する。入力 X を下位ビットから X1, X2, 入力 Y を同じく Y1, Y2, 出力 Z を同じく Z1, Z2 として、2 ビット全加算器を図示せよ。

(3) (2) の回路を修正して、2 ビットの減算 $X-Y$ も実現したい。入力 sub を新たに加え、sub=0 の時に $X+Y$, sub=1 の時に $X-Y$ の結果を出力する回路を図示せよ。

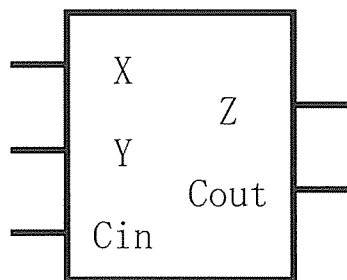


図 1-2：1 ビット全加算器

表 1-1：真理値表

X	Y	Cin	Z	Cout
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

III 南北方向と東西方向の道が交わる十字路に青と赤のみを示す信号機があるとする。この制御回路を以下の条件の下で作成する。

- 各信号機は常に青か赤のどちらかのみ点灯する。
- 向かい合う信号機は同じ色を示し，南北方向および東西方向の一方は青，他方は赤を示す。クリアランス時間（信号現示変更時に全て赤にする時間）はないとする。
- 青を示す方向にいる自動車は交差点を通過し，赤を示す方向にいる自動車は停止する。
- 赤を示す方向に自動車がいる場合，次のクロックでその方向が青になる。
- 赤を示す方向に自動車がない場合，次のクロックでも信号現示は変わらない。

回路の状態 Q を「南北方向が青（東西方向が赤）」「南北方向が赤（東西方向が青）」の 2 種類とし，順に 0, 1 と表現する。入力 N を南北方向に自動車がいる場合に 1，いない場合に 0 とし，入力 E を東西方向に自動車がいる場合に 1，いない場合に 0 とする。出力 A を「南北方向を青（東西方向を赤）にする」「南北方向を赤（東西方向を青）にする」とし，順に 0, 1 と表現する。次の (1)～(4) の問いについて答えよ。

- (1) 状態を丸印，状態遷移を矢印で表現した状態遷移図を示せ。ただし，丸印の中に各状態を示し，矢印には状態遷移に伴う入出力を「入力/出力」の形で付記せよ。入力として N, E をこの順にまとめて 2 ビットで記し，出力として A を記せ。
- (2) 現在の状態 Q と入力 N, E から，出力 A と次状態 Q_{next} を示す状態遷移表を表 1-2 の形式で示せ。
- (3) 状態遷移表から出力 A と次状態 Q_{next} を示す論理式を求めよ。できるだけ簡単化せよ。
- (4) 図 1-3 に示す JK フリップフロップを用いて，この制御回路を図示せよ。

表 1-2：状態遷移表

Q	N	E	A	Q_{next}
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

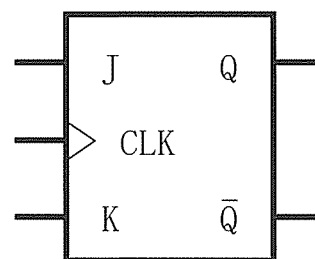


図 1-3：JK フリップフロップ

問題 2 設問すべてについて解答すること。

I 図 2-1 は，図 2-2 に示すグラフに対し，深さ優先探索を実行するプログラムであり，C 言語で書かれている。次の(1)，(2)の問いについて答えよ。

- (1) 図 2-1 の空欄 (ア) ~ (オ) に入る適切な記述を答えよ。なお，変数 a は図 2-2 のグラフを表現した隣接行列であり，行および列のインデックスがグラフの頂点番号を表している。
- (2) このプログラムが正しく作られていた場合に，プログラムを実行すると得られる標準出力への出力結果を答えよ。

行	コード
1	#include <stdio.h>
2	#define N (ア)
3	int a[(イ)][(イ)] =
4	{ {0,1,1,0,1},
5	{1,0,0,1,1},
6	{1,0,0,0,0},
7	{ (ウ) },
8	{1,1,0,1,0}};
9	int b[(イ)];
10	void tansaku(int i)
11	{
12	int j;
13	(エ) ;
14	for (j = 0; j < N; j++) {
15	if ((オ) && b[j] == 0) {
16	printf(" %d > %d , ", i, j);
17	tansaku(j);
18	}
19	}
20	}
21	void main(void)
22	{
23	int i;
24	for (i = 0; i < N; i++)
25	b[i] = 0;
26	tansaku(4);
27	printf("¥n");
28	}

図 2-1 : 深さ優先探索のプログラム

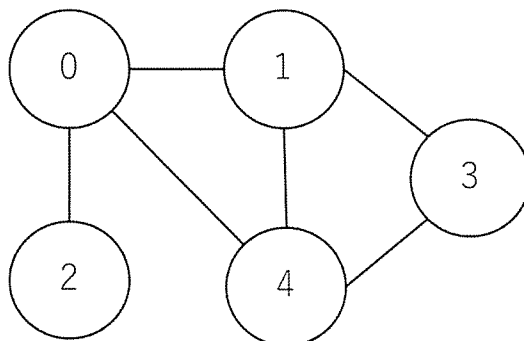


図 2-2 : 探索対象とするグラフ

II 図 2-3 は、ルートが最大となるヒープを作成するプログラムであり、C 言語で書かれている。次の(1), (2)の問いについて答えよ。

(1) 図 2-3 の空欄 (カ) ~ (ク) に入る適切な記述を答えよ。

(2) このプログラムが正しく作られていた場合に、プログラムを実行すると得られる標準出力への出力結果を答えよ。

行	コード
1	#include <stdio.h>
2	#define N 100
3	int heap[N];
4	int n = 0;
5	/* ヒープの表示 */
6	void print_heap(void) {
7	int i;
8	for (i = 1; i <= n; i++)
9	printf("%d ", heap[i]);
10	printf("\n");
11	}
12	/* ヒープに x を追加 */
13	void push_heap(int x) {
14	int i, j;
15	if (n < N - 1) {
16	n++;
17	heap[n] = x;
18	i = n;
19	j = (カ) ;
20	while (j > 0 && x > heap[j]) {
21	heap[i] = heap[j];
22	i = (キ) ;
23	j = (ク) ;
24	}
25	heap[i] = x;
26	}
27	}

行	コード
28	/* メイン関数 */
29	void main(void) {
30	int i;
31	/* 配列の初期化 */
32	for (i = 0; i < N; i++)
33	heap[i] = 0;
34	/* ヒープの作成 */
35	push_heap(5);
36	push_heap(3);
37	push_heap(6);
38	push_heap(1);
39	push_heap(9);
40	push_heap(7);
41	/* ヒープの表示 */
42	print_heap();
43	}

図 2-3 : ヒープを作成するプログラム

Ⅲ 図 2-4 は C++ で書かれたプログラムである。次の文章を読み、(1)～(3)の問いについて答えよ。

- (1) 図 2-4 のプログラムでは、複数の果物の合計金額を計算している。果物には葡萄とレモンの 2 種類があり、1 個当たりの金額の計算方法が異なる。そのため、各果物の金額の計算方法とその計算に必要な変数を定義するクラスを Grape, Lemon として作成している。また、両クラスを統一的に扱うために、両クラスの基底クラスとして Fruit を定義している。Grape や Lemon は、Fruit のメンバを (ケ) しているため、Grape や Lemon では定義されていない変数 (コ) を利用することができる。また、Fruit は関数 getPrice を持つが、getPrice は中身の機能が実装されていない純粋仮想関数であるため、Fruit はインスタンスの生成ができない。このようなインスタンスの生成ができないクラスは、(サ) クラスと呼ばれる。Grape や Lemon では異なる計算内容の getPrice が実装されている。このように基底クラスの関数を派生クラスで独自に定義し直すことを関数の (シ) と呼ぶ。

35 行目では、Fruit のポインタを格納する配列を、new 演算子を用いて動的に確保している。new 演算子はメモリ領域を確保した後、そのポインタを返す。38～40 行目では Grape や Lemon のインスタンスを生成し、それらのポインタを配列に格納している。C++ では、基底クラスのポインタ変数に派生クラスのポインタを格納できる。また、今回のように基底クラスのポインタが派生クラスのインスタンスを指している状態で、基底クラスの仮想関数を呼び出した場合には、派生先の関数が呼び出される。これにより、45 行目では、果物の種類に応じた計算方法が自動的に切り替わるようになっている。

上記 (ケ)～(シ) に入る適切な語句を以下の (A)～(J) から選び、記号で答えよ。

(A) 表象, (B) 抽象, (C) 継承, (D) 生成, (E) 構造体,
(F) オーバーライド, (G) オーバーロード, (H) カプセル化,
(I) weight, (J) grade

- (2) 図 2-4 の空欄 (ス)～(チ) に入る適切な記述を答えよ。
- (3) このプログラムが正しく作られていた場合に、プログラムを実行すると得られる標準出力への出力結果を答えよ。

行	コード	行	コード
1	#include <stdio.h>	30	// メイン関数
2		31	int main() {
3	class Fruit {	32	int N = 3;
4	protected:	33	
5	int grade;	34	// 動的配列の作成
6	public:	35	Fruit** obj = new (セ) [N];
7	virtual int getPrice(void) = 0;	36	
8	};	37	// インスタンスの生成と格納
9		38	obj[0] = new (ソ) (100,2);
10	class Grape : public Fruit {	39	obj[1] = new (タ) (120,1);
11	int weight;	40	obj[2] = new (チ) (10);
12	public:	41	
13	Grape(int w, int g) {	42	// 合計金額の計算と表示
14	weight = w; grade = g;	43	int price = 0;
15	}	44	for (int i = 0; i < N; i++)
16	int (ス) (void){	45	price += obj[i]->getPrice();
17	return weight * grade;	46	printf("%d¥n", price);
18	}	47	
19	};	48	// メモリの開放
20		49	for (int i = 0; i < N; i++)
21	class Lemon : public Fruit {	50	delete obj[i];
22	public:	51	delete[] obj;
23	Lemon(int g) {	52	}
24	grade = g;		
25	}		
26	int (ズ) (void){		
27	return 10 * grade;		
28	}		
29	};		

図 2-4 : C++で書かれたプログラム

問題 3 設問すべてについて解答すること。

なお、計算が必要な解答は特に指示のない限り小数の形ではなく最も簡約化した分数または対数で示すこと。ここで簡約化とは、分数に関しては既約形、対数に関しては分数を含まない最も簡単な形（例： $\log_2 6 = 1 + \log_2 3$, $\log_2 \left(\frac{1+x}{3}\right) = \log_2(1+x) - \log_2 3$ ）に変形することを指す。

I 以下の分布で定められる定常マルコフ情報源について考える。

$$P(X_t = 0|X_{t-1} = 0) = \frac{1}{4}, \quad P(X_t = 2|X_{t-1} = 0) = \frac{3}{4}$$

$$P(X_t = 0|X_{t-1} = 1) = \frac{1}{2}, \quad P(X_t = 1|X_{t-1} = 1) = \frac{1}{2}$$

$$P(X_t = 1|X_{t-1} = 2) = \frac{2}{3}, \quad P(X_t = 2|X_{t-1} = 2) = \frac{1}{3}$$

このとき、次の(1)～(3)の問いについて答えよ。

- (1) 上記のマルコフ情報源の状態遷移図（シャノン線図）を下に示す図 3-1 の例にならって示せ。この例では、丸で囲まれた文字は出力シンボルを示し、各シンボルを結ぶ矢印の上にかかれた数字は遷移確率を表している。

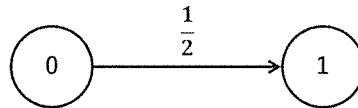


図 3-1：シャノン線図の例

- (2) 上記のマルコフ情報源について、状態遷移行列を示せ。
(3) 上記のマルコフ情報源における定常分布を求めよ。

II 情報アルファベットが $\{a, b, c, d, e\}$ である情報源 S があり、それらの生起確率が以下であるとする。

$$P(a) = 0.21, \quad P(b) = 0.25, \quad P(c) = 0.14, \quad P(d) = 0.1, \quad P(e) = 0.3$$

このとき、次の(1)～(3)の問いについて答えよ。ただし、解答には答えだけでなく、必ず導出過程を含めること。

- (1) これらの記号を二元符号 $\{0,1\}$ によりハフマン符号化し、その導出に用いるハフマン木とともに示せ。
(2) これらの記号を三元符号 $\{0,1,2\}$ によりハフマン符号化し、その導出に用いるハフマン木とともに示せ。
(3) (1), (2) それぞれの符号化について、平均符号長を求め、小数の形で小数第 2 位まで示せ。小数第 3 位以下がある場合は切り捨てること。

Ⅲ 次の文章を読み、(1)～(6)の問いについて答えよ。

いま、いびつな形をした形状の異なる3枚のコインA, B, Cが手元にある。これらのコインを投げた場合、表・裏の出る確率はコインごとに異なり、また、必ずしも等しくならない。ここで、はじめにコインAを投げ、表が出た場合にはコインBを、裏が出た場合にはコインCを投げるという試行を十分に多い回数行った。このとき、2回目にどちらのコインが投げられたか、また、表・裏のどちらが出たかのそれぞれについて回数を数えた。その結果、2回目に表が出る割合は50%であった。ここで、2回目に表が出た場合についてはコインBが投げられた割合が75%であった。一方で、2回目に裏が出た場合はコインCが投げられた割合が75%であった。1回目のコイン投げで表・裏のどちらが出たかを表す変数を X_1 (裏:0, 表:1) とし、2回目の結果を X_2 (裏:0, 表:1) とする。このとき、以下の問いに答えよ。

(1) エントロピー $H(X_2)$ を求めよ。

(2) 同時確率 $P(X_1 = 0, X_2 = 0)$, $P(X_1 = 0, X_2 = 1)$, $P(X_1 = 1, X_2 = 0)$, $P(X_1 = 1, X_2 = 1)$ をそれぞれ求めよ。

(3) 確率 $P(X_1 = 0)$, $P(X_1 = 1)$ をそれぞれ求めよ。

(4) エントロピー $H(X_1)$ を求めよ。

(5) 条件付エントロピー $H(X_1|X_2)$ を求めよ。

(6) 2回目のコイン投げの結果から得られる相互情報量を求めよ。