令和 8 年 4 月入学 一般入試 (第 III 期) General Entrance Examination(III) of April Admissions for the 2026 Academic Year
令和 8 年 4 月入学 外国人留学生特別入試 International Students Entrance Examination of April Admissions for the 2026 Academic Year
京都工芸繊維大学 大学院工芸科学研究科 博士前期課程（修士課程） 情報工学専攻 試験問題
Question booklet of Master's Program of Information Science, Graduate School of Science and Technology,
Kyoto Institute of Technology (KIT)

専門科目　Special Subjects

[注意事項 Cautions]

1. この問題冊子は合図があるまで中を開かないでください。
   Do not open this question booklet until permitted by the proctor.
2. 課題は以下の 3 題であり、3題とも必須です。落丁・乱丁および印刷の不鮮明な箇所などがあれば、手を挙げて監督者に知らせなさい。
   Answer all three subjects listed below. Raise your hand and inform the proctors of any missing pages, disarranged pages, unclear printing, etc.
   プログラミング　Programming
   ハードウェア　Hardware
   情報通信　Data communications
3. 配布物は、この問題冊子 1 部、解答用紙 3 枚、および下書き用紙 1 枚です。
   The proctors distribute this question booklet, three answer sheets, and a memo sheet.
4. 机の上には受験票以外に、次のものを置いてもよろしい。
   You can put the following goods in addition to your exam admission ticket.
   (a) 黒鉛筆とシャープペンシル Black pencils and mechanical pencils
   (b) プラスチック製の消しゴム Plastic erasers
   (c) 電動でない小型の鉛筆削り Small-sized non-electric pencil sharpeners
   (d) 秒針音がしない小型の時計（辞書、電卓、通信等の機能があるものは不可）Small-sized silent watches or clocks without any additional dictionary, calculator, communication, etc.
   (e) 眼鏡、ハンカチ、目薬、無地のマスク、ティッシュペーパー（袋又は箱から中身だけを取り出したもの） Glasses, handkerchiefs, eye drops, plain masks, tissues without package
   これら以外については監督者の了解を受けてください。
   Ask the proctors for permission to use any goods other than the above.
5. 解答用紙 3 枚すべての上欄指定枠内に、問題科目名 (例:「プログラミング」など)、志望専攻名、受験番号を忘れずに記入し、問題ごとに別々の解答用紙に解答してください。解答用紙の裏面に解答を書いても構いません。解答用紙と下書き用紙の追加配布はしません。
   Use a separate answer sheet for each subject part. Fill in the subject-part name, the major of Master's Program, and your examinee's number in the designated boxes on all two answer sheets. You can use both sides of the answer sheet. No additional sheet is available.
6. この問題冊子はバラしても構いません。
   You can unbind this booklet.
7. 試験終了後も退出の許可があるまで退室はできません。中途退室できません。
   Do not leave the room after the exam until permitted by the proctor. Also, you do not during the exam.
8. 問題冊子と下書き用紙は持ち帰ってください。
   Bring this question booklet and the memo sheet when you leave the room after the exam.

プログラミング [1/4]

問１　図１に示すＣ言語で記述されたプログラム１の空欄（あ）に実引数(argument)として図２に示す(a), (b), (c)をそれぞれ当てはめた場合、プログラム１が実行可能かどうかを答えよ。実行可能である場合は、標準出力(standard output)に出力される内容を示せ。

```
#include <stdio.h>
typedef struct{int x;} S;

int main(void){
    S a[2]={{1},{2}}, *p=a;
    printf("%d¥n", （あ） );
    return 0;
}
```

図１　プログラム１

```
(a) *p++.x
(b) p++->x
(c) (*p).x
```

図２

問２　図３に示すＣ言語で記述されたプログラム２は、実数(real number)x と非負整数(non-negative integer)n が与えられたとき、x の n 乗を計算する関数 power を定義し、利用するものである。プログラム２の空欄（あ）～（く）に適切なコードを補い、プログラムを完成せよ。

```
#include <stdio.h>

double power(double x, int n){
    if(n==0) return （あ）;
    if(n%2 == 0){
        double half=power(（い）,（う）);
        return （え）*（お）;
    }else{
        return （か）*power(（き）,（く）);
    }
}

int main(void){
    double x;
    int n;
    printf("Enter a real number and non-negative integer.¥n");
    scanf("%lf %d", &x, &n);

    printf("%lf¥n", power(x, n));
    return 0;
}
```

図３　プログラム２

［次ページに続く］

プログラミング [2/4]

問３　無向グラフ(undirected graph)G=(V,E)を考える。
頂点集合(vertex set)はV={a, b, c, d, e, f}、重み付き辺集合(weighted edge set)をE={(a,b):4,(a,c):2,(b,e):3,(c,d):2,(c,f):4,(d,e):3,(d,f):1,(e,f):1}とする。ここで、重み付き辺(weighted edge)(a,b):4は、頂点(vertex)aと頂点bが接続され、その辺の重み(頂点aと頂点bの距離)が4であることを示す。
この時、頂点aから各頂点までの最短経路(shortest path)を求めるプログラムを、図４、図５、図６に示す。これらはいずれもＣ言語で記述されたものである。

・図４:指定された無向グラフの各辺を格納する関数addedge()と、すべての頂点についての最短経路を表示する関数printpath()を含むプログラムである。
・図５:最小の値(minimum value)が根(root)にある完全二分木(complete binary tree)のデータ構造(data structure)である最小ヒープ(minimum heap)を構築・操作するための関数群である。
・図６:ダイクストラ(Dijkstra)法により最短経路を求めるための関数 Dijkstra()である。

以下の問い(１), (２)に答えよ。

(１)図５および図６の空欄 (あ) ～ (き) に適切なコードを補い、プログラムを完成せよ。
(２)(１)で完成させたプログラムに対して、以下の変更を全て施した後に実行するとどのような結果が生じるか。また、その理由を説明せよ。
・変数の宣言(a)、条件文(b), (c), (d)を削除する。
・重み付き辺集合の(c,f)の重みを4から-4に変更する。

[次ページに続く]

プログラミング [3/4]

```
#include <stdio.h>
#define N 6
#define MAXM 32
#define HCAP 128
#define INF 10000000

typedef struct { int to, w, next; } Edge;
typedef struct { int d, v; } Item;
Edge E[MAXM];
Item H[1+HCAP];
int head[N], distv[N], prevv[N], EC=0, hsz=0;
int NEGATIVEEDGE=0;  //(a)

void addedge(int u, int v, int w){
    if(w<0){  //(b)
        NEGATIVEEDGE=1;
        return;
    }
    E[EC]=(Edge){ v, w, head[u] };
    head[u]=EC++;
}
void printpath(int v){
    if(prevv[v]!=-1) { printpath(prevv[v]); printf("->"); }
    printf("%c", 'a'+v);
}

int main(void){
    for(int i=0; i<N; ++i) head[i]=-1;

    addedge(0,1,4); addedge(1,0,4);
    addedge(0,2,2); addedge(2,0,2);
    addedge(1,4,3); addedge(4,1,3);
    addedge(2,3,2); addedge(3,2,2);
    addedge(2,5,4); addedge(5,2,4);
    addedge(3,4,3); addedge(4,3,3);
    addedge(3,5,1); addedge(5,3,1);
    addedge(4,5,1); addedge(5,4,1);

    dijkstra(0);

    if(NEGATIVEEDGE){  //(c)
        printf("Negative edge was present.¥n");
        return 0;
    }
    for(int v=0; v<N; ++v){
        printf("dist(a->%c)=%d : ", 'a'+v, distv[v]);
        printpath(v);
        printf("¥n");
    }
    return 0;
}
```

図4　プログラム3

[次ページに続く]

プログラミング [4/4]

```
int less(Item a, Item b){ return  (あ) ; }
void hswap(int i, int j){ Item t=H[i]; H[i]=H[j]; H[j]=t;}
void push(Item x){
    H[++hsz]=x;
    for(int i=hsz; i>1 && (い) ; i/=2) hswap(i,i/2);
}

Item pop(void){
    Item top=H[1];
    H[1]=H[hsz--];
    for(int i=1; ;){
        int L=i*2, R=L+1, s=i;
        if (L<=hsz && less(H[L], H[s])) s=L;
        if (R<=hsz && (う) ) s=R;
        if (s==i) break;
        hswap(i, s); i=s;
    }
    return top;
}
```

図５　最小ヒープを求める関数群のプログラム

```
void dijkstra(int src){
    if(NEGATIVEEDGE) return;  //(d)
    for(int i=0; i<N; ++i){
        distv[i]=INF;
        prevv[i]=-1;
    }
    distv[src]=0;
    push((Item){ 0, src });
    while(hsz){
        Item it=pop();
        int d=it.d, u=it.v;
        if ( (え) ) continue;
        for(int ei=head[u]; ei!=-1; ei=E[ei].next){
            int v=E[ei].to, w=E[ei].w;
            if (distv[v]> (お) + w){
                distv[v]= (か) +w;
                prevv[v]=u;
                push((Item){ (き) , v });
            }
        }
     }
}
```

図６　関数 Dijkstra()のプログラム

ハードウェア

問1 右の真理値表（truth table）に対応する 4 変数の論理関数（logic function）$f(a,b,c,d)$について，設問(a)～(d)に答えなさい．

a	b	c	d	f
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

(a) $f(a,b,c,d)$のカルノー図（Karnaugh map）を示しなさい．ただし，abを行，cdを列として示すこと．

(b) $f(a,b,c,d)$の最小積和形（minimal sum-of-products form）をすべて求めなさい．なお，その個数も明示すること．

(c) $f(a,b,c,d)$ の最小和積形（minimal product-of-sums form）をすべて求めなさい．なお，その個数も明示すること．

(d) 設問(b)の最小積和形の 1 つを選び，それをもとに，$f(a,b,c,d)$を実現する組み合わせ回路（combinatorial circuit）を 2 入力または 3 入力の NAND ゲート（NAND gate）のみを用いて構成し，その回路図（schematic circuit diagram）を示しなさい．

問2 コンピュータ内部での整数の表現と演算について，設問(a)～(d)に答えなさい．

(a) 符号なし整数（unsigned integer）を n ビットで表現する場合の整数の最大値と最小値を答えなさい．

(b) 負の数を 2 の補数（2's complement）で表す場合，n ビットで表現できる整数の最大値と最小値を答えなさい．

(c) 負の数を 2 の補数で表す場合と 1 の補数（1's complement）で表す場合について，整数の加算手続きはどのように異なるか説明しなさい．

(d) 負の数を 2 の補数で表すものとする．整数が正負のいずれであっても，それをkビットだけ左シフト演算することは，それによってオーバフローが生じない限り，元の数を 2^k倍することに等しい．ここで，nビットの負の数の場合にこれが正しいことを説明しなさい．

情報通信 [1/2]

問 1

入力アルファベット (Input Alphabet) $A = \{0, 1\}$ ならびに出力アルファベット (Output Alphabet) $B = \{0, 1, e\}$ を有する，下図の 2 元消失通信路 (Binary Erasure Channel, BEC) を考える．

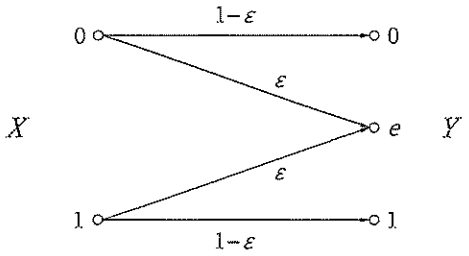


図: 2 元消失通信路

ここで，X と Y はそれぞれ入力記号 (Input Symbol) $x \in A$ と出力記号 (Output Symbol) $y \in B$ の確率変数 (Random Variable)，出力記号 $e \in B$ は 0, 1 のどちらにも判定されなかった消失記号 (Erasure Symbol)，入力記号 $x \in A$ から消失記号 $e \in B$ への遷移確率 $\varepsilon = P_{Y|X}(e|x)$ は消失確率 (Erasure Probability) である．例えば，$\varepsilon = 0.1$ のときに，長さ 1,000 の 2 元系列を送信すれば，受信側では平均して 100 個の記号を消失することになる．

入力記号 $1 \in A$ の生起確率 (Occurrence Probability) を $P_X(1) = p$ としたときに，次の 4 つの情報量 (Information Content) を p と ε を用いた式で表しなさい．

(a) X のエントロピー (Entropy) $H(X)$
(b) Y に対する X の条件付エントロピー (Conditional Entropy) $H(X|Y)$
(c) X と Y の相互情報量 (Mutual Information) $I(X;Y)$
(d) 2 元消失通信路の通信路容量 (Channel Capacity) C_0

問 2

インターネット上での通信相手が正規のサーバであることを証明する基盤となるサーバ証明書 (server certificate) について，以下の問いに答えよ．

(a) 近年，認証局/ブラウザフォーラム (CA/Browser Forum) において，サーバ証明書の有効期限 (validity period) を短縮する議論が行われている．2020 年 9 月以前のサーバ証明書の有効期限は 825 日であったが，これを徐々に短縮して 2029 年 3 月 15 日以降は有効期限を 47 日とすることがアナウンスされている．サーバ証明書の有効期限を短縮する理由として何が考えられるか，情報セキュリティの観点から答えよ．

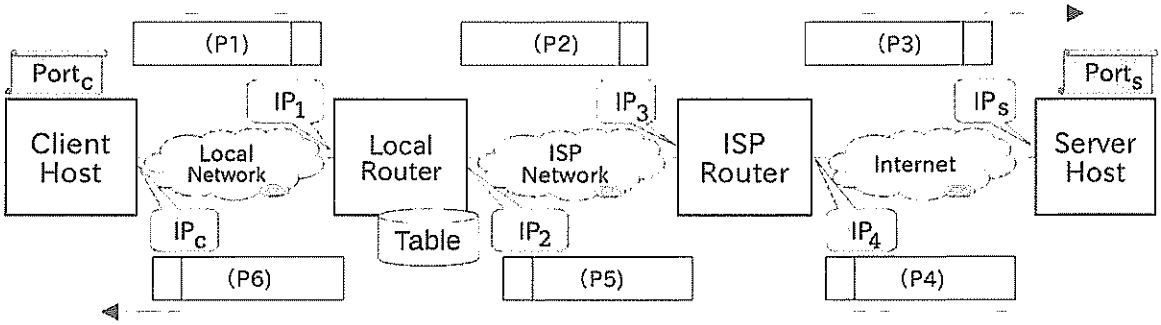
(b) サーバ証明書の有効期限短縮に伴い，サーバ証明書を自動更新する仕組みが不可欠になる．証明書の更新期限が近づくと，サーバは認証局 (certificate authority) に対して新しいサーバ証明書の発行を要求する．ここで，認証局は受け取った更新要求が正規の DNS ドメイン管理者 (あるいは Web サイト管理者) からの要求であることを自動的に検証したい．どのような検証方法が考えられるか，認証局・要求元サーバとの間でやり取りされる情報と合わせて説明せよ．

[次ページに続く]

情報通信 [2/2]

問 3

IPv4 のままでインターネットの接続機器を増やす仕組みとして，NAPT (Network Address and Port Translation) がある．また，RFC1918 で定義されている Address Allocation for Private Internets (通称プライベートアドレス) があり，いくつかの IPv4 アドレスブロックは，グローバルなインターネット空間では直接使用しないことになっている．一方，いくつかのインターネットサービスプロバイダ (Internet Service Provider, ISP) では，自社内のネットワークと他者のネットワークとの間で NAPT を行う，CGN (Carrier Grade NAT) や LSN (Large Scale NAT) と呼ばれる技術も使われるようになってきている．このとき，ISP への加入者が NAPT 機器を用いて ISP に接続し，ISP からインターネットに接続するところでも NAPT が行われる．



図は，この状況を，アドレス IP_c を持つ Client Host がポート $Port_c$ を用い，アドレス IP_s を持つ Server Host のポート $Port_s$ と通信しようとしていることを図示している．(P1)〜(P6) はパケットを表しており，Client Host から Server Host へパケットは (P1), (P2), (P3) と流れ，Server Host から Client Host への応答パケットは (P4), (P5), (P6) と流れることを示している．一般に，パケットは，(送信元アドレス，送信元ポート，宛先アドレス，宛先ポート) の 4 項組で識別される．

(a) NAPT の動きについて，図の例示に基づいて Local Router 上のみで行われる場合を想定し，(P1)〜(P6) のそれぞれのパケットの 4 項組がどうなるかを説明せよ．なお，説明にあたって，図に存在しないコンポーネントが必要であれば適宜追加して説明しても良い．

(b) 加入者の Local Network で 192.168.0.0/24 を利用しており，ISP 側より IP_2 として 192.168.0.10/16 が付与された場合に，起こりうる問題について簡単に説明せよ．

(c) (b) のような問題が生じないようにするためには，どのような方法が考えられるかについて，アイディアを説明し，そのアイディアでは (b) のような問題が起こらないことを説明せよ．

解答例及び出題意図

プログラミング

解答(100)
問1 (20 = 4 x 5)
（a）実行不可
（b）実行可
（c）実行可
標準出力(standard output)に出力される内容
1
1

問2(24 = 4 x 6, 10=2 x 5)

あ:1.0

い：x

う：n/2　5

え：half

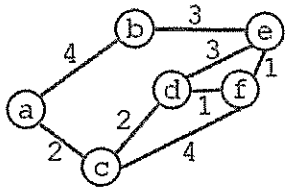
お：half

か：x

き：x

く：n-1　5

問3



(1)(20=4×5, 10=5×2)

あ:a.d<b.d

い：less(H[i], H[i/2])　5

う：less(H[R],H[s])　5

え：d!=distv[u]

お：d

か：d

き：distv[v]
(2) (16=2×8)
発生する結果： プログラムが無限ループまたは異常終了する。
理由：
(c,f) の重みを -4 にすると負閉路が生じ、ダイクストラ法の前提（非負辺）が崩れるため、距離計算が収束しない。

ハードウェア

問1

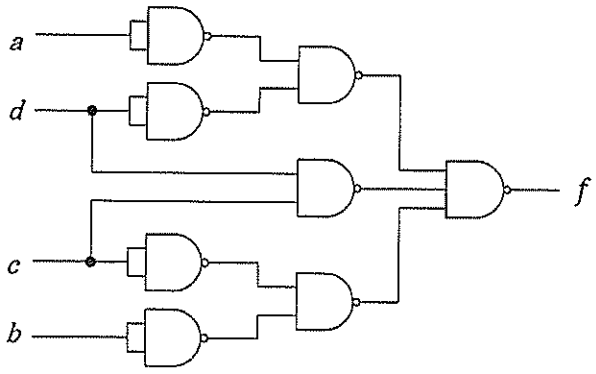
(a)

ab＼cd	00	01	11	10
00	1	1	1	1
01	1	0	1	1
11	0	0	1	0
10	1	1	1	0

(b) 1個 $f(a,b,c,d) = \bar{a}\bar{d} + \bar{b}\bar{c} + cd$

(c) 2個 $f(a,b,c,d) = (\bar{b} + c + \bar{d})(\bar{a} + \bar{c} + d)(\bar{a} + \bar{b} + c)$
$f(a,b,c,d) = (\bar{b} + c + \bar{d})(\bar{a} + \bar{c} + d)(\bar{a} + \bar{b} + d)$

(d)



問2

(a) 最大値は$2^n - 1$，最小値は0.

(b) 最大値は$2^{n-1} - 1$，最小値は-2^{n-1}.

(c) 負の数を2の補数で表す場合は，符号ビットも含めて加算し，最上位桁からの繰り上げは無視する．これに対して，負の数を1の補数で表す場合は，最上位桁から繰り上げが生じると加算結果の最下位に1を加算する．

(d) nビットの正の整数Aに対して，$-A$は2の補数を用いて$2^n - A$で表される．これをkビット左にシフトすると，$(2^n - A) \times 2^k = 2^{n+k} - A \cdot 2^k = (2^{n+k} - 2^n) + (2^n - A \cdot 2^k)$．ここで$2^{n+k} - 2^n$は$n+1$ビット目より上位のビットが表す値で，元の$n$ビットには$2^n - A \cdot 2^k$が残る．これは$A \cdot 2^k$の補数であり，$-A \cdot 2^k$を表している．つまり，$-A$を$2^k$倍したことになる．

情報通信

問1

(解答)

(a) $H(X) = -(1-p)\log_2(1-p) - p\log_2 p$

(b) $H(X|Y) = \varepsilon H(X) = \varepsilon(-(1-p)\log_2(1-p) - p\log_2 p)$

(c) $I(X;Y) = H(X) - H(X|Y) = (1-\varepsilon)H(X) = (1-\varepsilon)(-(1-p)\log_2(1-p) - p\log_2 p)$

(d) $C_0 = \max_{0\leq p\leq 1} I(X;Y) = 1-\varepsilon$

問2

(出題意図)

(a) いくら通信路が暗号化されていても、通信相手が正規の相手であると証明できなければ安全ではない。サーバ証明書の有効期間を短くすることの情報セキュリティ上の効果を問う設問である。サーバ証明書の秘密鍵が漏洩する、サーバ証明書が不正に発行されるなどサーバ証明書が危殆化する事態が生じた場合でも、サーバ証明書の有効期間を短くすることで被害を受ける期間を短くすることができる。また、古くなった暗号化アルゴリズムの変更や鍵長の変更を促すことにもつながり、セキュリティ水準の向上が期待される。

(b) 「自身が正規のＤＮＳドメイン・Webサイト管理者であること」をどのように示すかを問う設問である。第三者が正規の管理者であることを勝手に名乗れるような脆弱な方式であってはならない。サーバ証明書の自動更新手続はACME(Automatic Certificate Management Environment)プロトコルとして標準化されており、このプロトコルに相当する内容が書かれていればよい。認証局が検証すべき内容は2つあり、1つは「正規のサーバ証明書更新要求であること」、もう1つは「正規のDNSドメイン・Webサイト管理者であること」である。正規の更新要求であることは公開鍵を用いた電子署名により検証できる。正規のDNSドメイン・Webサイト管理者であることの検証は、認証局からの「チャレンジ」を通じて行われる。チャレンジとして、認証局から送られてきたトークン（ランダムな文字列）をＤＮＳあるいはWebサイトの特定の場所に登録する方式がよく用いられる。登録が完了したことを認証局に通知すると、認証局はトークンが登録されたことを外部からのアクセスにより確認し、これによって正規の管理者であることを検証する。

問3

(出題の意図)

(a) インターネットのパケット転送についての基本理解があり，それを説明できる能力を有しているかどうか．

(b) また，そこで生じうる問題への気づきと解決提案に思い至る思考力を有しているかどうかを問うている．