

令和 7 年 4 月入学 一般入試 (第 III 期) General Entrance Examination(III) of April Admissions for the 2025 Academic Year

令和 7 年 4 月入学 外国人留学生特別入試 International Students Entrance Examination of April Admissions for the 2025 Academic Year

京都工芸繊維大学 大学院工学科学研究科 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]

- 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.
- 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
- 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, two answer sheets, and a memo sheet.
- 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 (カッターナイフは可) Small-sized non-electric pencil sharpeners (including a small cutter)  
(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.
- 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.
- この問題冊子はバラしても構いません。  
You can unbind this booklet.
- 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.
- 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 [1/2]

問1 C言語で記述された Program 1 を実行して標準出力(standard output)に表示される内容を示せ。

Program 1

```
#include <stdio.h>
int main(void) {
    int a[] = {1,2,3,4,5}, *p = &a[1], i = 2;
    printf("%d\n", a[ a[ a[ a[ a[0] ] ] ] ] % 4);
    printf("%d\n", *a + *p);
    printf("%d\n", a[++i] > 3 ? a[++i] : a[i++]);
    return 0;
}
```

問2 C言語で記述された Program 2 は、文字列 (string) *s* が文字列 *t* を含むとき 1、含まないとき 0 を返す関数(function)である。  
□(あ)、□(い)、□(う) を埋めて関数を作成せよ。

Program 2

```
int includes(char *s, char *t) {
    char □(あ) = t;
    while(*s != '\0' && *p != '\0') {
        if(*s == *p) {
            s++; p++;
        } else {
            s += □(い); p = t;
        }
    }
    return □(う) ? 1 : 0;
}
```

問3 Java 言語で記述された Program 3 に関する問い(a)、(b)に答えよ。

Program 3

<pre>class A{     int a = 1;     int f() {return a;}     int f(int x) {return x;} } class B extends A{     int b = 2;     int f(int y) {return y+b;} }</pre>	<pre>class Main{     public static void main(String args[]) {         B a = new B();         int x = 3;         a.a = 4;         System.out.println(a.f());         System.out.println(a.f(x));     } }</pre>
--------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

- (a) Program 3 を実行して標準出力に表示される内容を示せ。  
(b) クラス(class) B の変数(variable) *b* の値を B 以外のクラスから変更できないようにするためには、クラス B のソースコード(source code)をどのように変更すればよいか、簡潔に説明せよ。

[次ページに続く]

## プログラミング [2/2]

問4 C言語で記述された Program 4 と Program 5 に関する問い(a)~(c)に答えよ。

Program 4

```
#include <stdio.h>
#define M 10
#define N 5

void bucket(int a[], int b[],
            int n){
    int i, place[M], j = 0;
    for(i = 0; i < M; i++){
        place[i] = 0;
    }
    for(i = 0; i < n; i++){
        place[ a[i]-1 ] = a[i];
    }
    for(i = 0; i < M; i++){
        if(place[i] > 0){
            b[j] = place[i];
            j++;
        }
    }
}

int main(void){
    int i, a[N] = {4,10,7,6,3};
    int b[N] = {0,0,0,0,0};
    bucket(a, b, N);
    for(i = 0; i < N; i++){
        printf("%d ",b[i]);
    }
    printf("\n");
    return 0;
}
```

Program 5

```
#include <stdio.h>
#define M 10
#define N 5

void counting(int a[], int b[],
              int n){
    int i, place[M];
    for(i = 0; i < M; i++){
        place[i] = 0;
    }
    for(i = 0; i < n; i++){
        place[ a[i]-1 ]++;
    }
    for(i = 0; i < (え) ; i++){
        place[i+1] += place[i];
    }
    for(i = 0; i < n; i++){
        b[ (お) [ a[i]-1 ] ] = a[i];
    }
}

int main(void){
    int i, a[N] = {4,10,7,6,3};
    int b[N] = {0,0,0,0,0};
    counting(a, b, N);
    for(i = 0; i < N; i++){
        printf("%d ",b[i]);
    }
    printf("\n");
    return 0;
}
```

- (a) Program 4 を実行して標準出力に表示される内容を示せ。
- (b) Program 5 を実行して標準出力に表示される内容が、Program 4 を実行して標準出力に表示される内容と同じとなるように (え)、 (お) を埋めて、プログラムを完成させよ。
- (c) main 関数の配列(array) a は要素(element)の個数を5とし、各要素は1から10までの整数値で初期化しなければならない。Program 4 と Program 5 の初期値は4,10,7,6,3となっている。ところが初期値が同じでも、Program 4 と、完成させた Program 5 とで、実行したときの標準出力に表示される内容が異なることがある。異なる内容が表示されるのは初期値がどのようになっているときか、説明せよ。

# 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. ただし, 真理値表中の「\*」は dont care (don't care) を表す.

$a$	$b$	$c$	$d$	$f$
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	*
1	0	1	0	*
1	0	1	1	0
1	1	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)  $f(a, b, c, d)$  に対する論理式 (logic expression) の最小積和形 (minimal sum-of-products form) を示しなさい.
- (c)  $f(a, b, c, d)$  に対する論理式の最小和積形 (minimal product-of-sums form) を示しなさい.
- (d) 設問(c)の最小和積形をもとに,  $f(a, b, c, d)$  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 NOR ゲート (NOR 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 プロセッサにおけるマシン命令 (instruction) の処理について, 設問(a)(b)に答えなさい.

- (a) 命令実行サイクル (instruction cycle) は複数のステージで構成される. それぞれのステージとその順序について説明しなさい.
- (b) 命令レベルの並列処理アーキテクチャには, VLIW (Very Long Instruction Word), スーパスカラ (superscalar), スーパーパイプライン (super-pipeline), がある. それぞれの特徴を説明し, それらの処理性能の優劣について論じなさい.

## 情報通信 [1/2]

### 問 1

店頭での代金支払いにおいてキャッシュレス (cashless) 決済が広まりつつあり、支払い用の QR コードをカメラで読み込む方式 (QR コード決済方式, QR code payment) と支払いカードと決済端末の間に近距離無線通信 (NFC) を行う方式 (タッチ決済方式, contactless payment) が広く使われている。(注) QR コードは, (株) デンソーウェーブの登録商標です.)

- (a) QR コード決済方式で QR コードに決済金額の情報を持たせることのメリットとデメリットを述べよ。
- (b) タッチ決済対応のクレジットカードでは, カードが偽造されたものでないことを確かめるための通信がカードと決済端末の間で行われる。この時に利用できる暗号技術を挙げ, カード・決済端末間の通信にどのように使うか説明せよ。

### 問 2

TCP/IP を用いた通信に関する以下の文章について, 以下の [あ]~[つ] に当てはまる適切な語句を語群から選んで答えよ。また, [A] に当てはまる適切な対策を文章で答えよ。

TCP/IP を用いた通信では, [あ] 層のプロトコル (Protocol) として [い] と [う] が広く使われている。このうち, [い] を主に使用するインターネット (Internet) 上の [え] 層のプロトコルとして, FTP や [お], [か] などが, [う] を主に使用する [え] 層のプロトコルとして [き], [く] などが知られている。

[い] は [け] 型であるため, 通信を開始する為には [け] を確立する必要がある。接続元から [こ] フラグを含むパケットを送り, 接続先から [こ] と [さ] フラグを含むパケットを送り, それに対する [さ] フラグを含むパケットを送るという [し] を実行するために双方向の [す] のやり取りが必須となる。一方, [う] は [せ] 型であるため, [け] を確立する必要が無く [そ] のやり取りが素早く実施でき [た] 性の高い通信に適している。

ここで, 送信元のアドレス偽装 (Address Spoofing) を行った [す] を送りつける攻撃について考える。[い] の場合, [け] を確立するための [こ] フラグを含むパケットを送信元のアドレスを偽装して送ることで [こ] と [さ] フラグを含むパケットを偽装したアドレス宛に集中的に送らせる攻撃や, 通信を行っていると思われるアドレスを偽装して [ち] フラグを含むパケットを大量に送りつけることで, [け] を強制切断させるような攻撃が可能と考えられる。一方, [う] の場合, [け] の確立が不要であるため, 単純に偽装されたアドレスを送信先として応答を返してしまうことになる。そのため, インターネット上に該当プロトコル用のサーバ (Server) が多数あることや, 問合せ [そ] に対する応答 [そ] のサイズが大きい, という条件を満たす [き] や [く] のプロトコルは, リフレクター攻撃 (Reflector Attacks) と呼ばれる [つ] 攻撃に使われることが多くなっている。

このような送信元のアドレス偽装をしづらくするために, 各組織のネットワークの出入り口のルータで, [A] というような対策が用いられている。

#### [語群]

3-way handshake, ACK, DNS, DoS, FIN, HTTP, ICMP, IP, MAC, NAT, NTP, RST, SMTP, SYN, TCP, UDP, URG, アプリケーション (Application), 輻輳ウィンドウ (Congestion Window) コネクション (Connection), コネクションレス (Connectionless), データリンク (Data link), フレーム (Frame), 高信頼性 (High Reliability), メッセージ (Message), ネットワーク (Network), パケット (Packet), リアルタイム (Realtime), 受信ウィンドウ (Receive Window), セグメント (Segment), トランスポート (Transport), ウィルス (Virus)

[次ページに続く]

## 情報通信 [2/2]

### 問 3

次のパリティ検査行列 (parity check matrix)  $H$  で定義されるハミング符号 (Hamming code) を考える。

$$H = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

- (a) この符号の生成行列 (generator matrix)  $G$  を求めよ。
- (b) 情報ベクトル (information vector)  $u = (0, 1, 1, 0)$  に対する符号語 (codeword)  $v$  を求めよ。
- (c) (b) で求めた符号語 (codeword)  $v$  に対して誤りベクトル (error vector)  $e = (0, 1, 0, 0, 0, 0, 1)$  で表される誤りが生じた場合の受信語 (received word)  $r$  を求めよ。
- (d) (c) で求めた受信語 (received word)  $r$  を最小距離復号法 (minimum distance decoding) で復号して得られる推定符号語 (estimated codeword)  $\hat{v}$  を求めよ。

[以上]