平成２９年度

筑波大学大学院博士課程
システム情報工学研究科
コンピュータサイエンス専攻
博士前期課程（一般入学試験　８月期）
試験問題　基礎科目（数学，情報基礎）

Mathematics／Fundamentals of Computer Science (CS)

[注意事項][Instructions]

1. 試験開始の合図があるまで，問題の中を見てはいけません．また，筆記用具を手に持ってはいけません．
   Do NOT open this booklet before the examination starts. In addition, do not have a pen in hand before the examination starts.
2. 試験開始の合図のあとで，全ての解答用紙の定められた欄に，研究科，専攻，受験番号を記入すること．
   Fill in the designated spaces on each answer sheet with the name of the graduate school, name of your main field, and the examination number after the examination starts.
3. この問題は全部で 21 ページ（表紙を除く）です．1～10 ページは日本語版，11～21 ページは英語版です．
   This booklet consists of 21 pages, excluding the cover sheet. The Japanese version is shown on pages 1–10 and the English version on pages 11–21.
4. 解答用紙（罫線有り）を４枚，下書き用紙（白紙）を１枚配布します．
   You are given four answer sheets (ruled paper) and one draft sheet (white paper).
5. 問題は全部で 4 問（数学$\boxed{1}$，数学$\boxed{2}$，情報$\boxed{1}$，情報$\boxed{2}$）あります．このうち，以下の組み合わせで２問選択すること．数学$\boxed{1}$と情報$\boxed{1}$を同時に選択しないこと．各問題には小問が２問（(1)と(2)）あります．小問ごとに解答用紙を分けて記入すること．
   There are four problems (Math$\boxed{1}$, Math$\boxed{2}$, CS$\boxed{1}$, and CS$\boxed{2}$). Select two problems from the following combinations. Do NOT select Math$\boxed{1}$ and CS$\boxed{1}$ together. Each problem has two sub-problems (1) and (2). Write your answer to each sub-problem in a different answer sheet.

	解答用紙１枚目と２枚目 / First and second answer sheets	解答用紙３枚目と４枚目 / Third and fourth answer sheets
1	数学／Math$\boxed{1}$(1)(2)	数学／Math$\boxed{2}$(1)(2)
2	数学／Math$\boxed{1}$(1)(2)	情報／CS$\boxed{2}$(1)(2)
3	数学／Math$\boxed{2}$(1)(2)	情報／CS$\boxed{1}$(1)(2)
4	数学／Math$\boxed{2}$(1)(2)	情報／CS$\boxed{2}$(1)(2)
5	情報／CS$\boxed{1}$(1)(2)	情報／CS$\boxed{2}$(1)(2)

6. 解答用紙に解答を記述する際に，問題番号を必ず明記すること．
   When writing the answers, clearly label the problem number on each answer sheet.

平成２８年８月２５日

問題 数学 $\boxed{\mathbf{1}}$ **(1)** は，数学 $\boxed{\mathbf{1}}$ **(2)** とともに解答すること． 数学 $\boxed{\mathbf{1}}$ を選択した場合は，情報 $\boxed{\mathbf{1}}$ を選択しないこと． この解答用紙の先頭に，「数学 $\boxed{\mathbf{1}}$ **(1)**」と明記し，それ以外の問題を解答しないこと．

問題 数学 $\boxed{\mathbf{1}}$ **(1)**

正の整数を正の整数に対応付ける関数 **fib** を次の式で定義する．

$$\mathbf{fib}(n) = \begin{cases} 1 & n \leq 2 \text{ のとき} \\ \mathbf{fib}(n-2) + \mathbf{fib}(n-1) & n > 2 \text{ のとき} \end{cases}$$

(a) $\mathbf{fib}(15)$ および $\mathbf{fib}(16)$ の値を求めよ．

(b) 0 以上の整数 a と正の整数 d に対して，a を d で割った余りを $a \ \mathbf{mod} \ d$ と表すとき，$(\mathbf{fib}(16))^2 \ \mathbf{mod} \ 10$ の値を求めよ．ただし，0 以上の整数 x, y と正の整数 d に対して，

$$(x \cdot y) \ \mathbf{mod} \ d = ((x \ \mathbf{mod} \ d) \cdot (y \ \mathbf{mod} \ d)) \ \mathbf{mod} \ d$$

が成立することを使ってよい．

(c) $\mathbf{fib}(256) \ \mathbf{mod} \ 10$ を求めよ．ただし，以下の公式がすべての $n \geq 2$ に対して成立することを使ってよい．

$$\mathbf{fib}(2n-1) = (\mathbf{fib}(n-1))^2 + (\mathbf{fib}(n))^2$$
$$\mathbf{fib}(2n) = 2 \cdot \mathbf{fib}(n-1) \cdot \mathbf{fib}(n) + (\mathbf{fib}(n))^2$$

(d) $\mathbf{fib}(2^{100}) \ \mathbf{mod} \ 10$ の値を求めよ．

問題 数学 $\boxed{\mathbf{1}}$ **(2)** は，数学 $\boxed{\mathbf{1}}$ **(1)** とともに解答すること． 数学 $\boxed{\mathbf{1}}$ を選択した場合は，情報 $\boxed{\mathbf{1}}$ を選択しないこと． この解答用紙の先頭に,「数学 $\boxed{\mathbf{1}}$ **(2)**」と明記し，それ以外の問題を解答しないこと．

問題 数学 $\boxed{\mathbf{1}}$ **(2)**

---

n を正の整数とする．0 以上の整数 x と y で $x+y<n$ を満たすものの組 (x,y) をすべて集めた集合を $\mathbf{T}_n$ とする．また，$0 \leq z < \dfrac{n(n+1)}{2}$ を満たす整数 z をすべて集めた集合を $\mathbf{N}_n$ とする．後述する様々な関数 $f : \mathbf{T}_n \to \mathbf{N}_n$ に対して，f が単射であるかどうか，すなわち，次の条件 (*) が成立するかどうかを考える．

> (*) すべての $(x_1, y_1) \in \mathbf{T}_n$ と $(x_2, y_2) \in \mathbf{T}_n$ に対して，$f(x_1, y_1) = f(x_2, y_2)$ ならば，$x_1 = x_2$ かつ $y_1 = y_2$ が成立する．

このとき，以下の問に答えなさい．

(a) $n=2$ とし，$f(x,y) = 2x+y$ と定義するとき，$f(0,0)$，$f(0,1)$，$f(1,0)$ のそれぞれの値を計算しなさい．また，この f が単射であることを示しなさい．

(b) $n=4$ とし，$f(x,y) = \dfrac{(x+y)(x+y+1)}{2} + x$ と定義するとき，$f(0,3)$，$f(1,2)$，$f(2,1)$，$f(3,0)$ の値を計算しなさい．

(c) 0 以上のすべての整数 z_1 と z_2 に対して，$z_1 \geq z_2 + 1$ ならば以下の不等式が成立することを示しなさい．

$$\frac{z_1(z_1+1)}{2} \geq \frac{z_2(z_2+1)}{2} + z_2 + 1$$

(d) $n > 0$ に対して，次のように定義される $f : \mathbf{T}_n \to \mathbf{N}_n$ が単射であることを示しなさい．

$$f(x,y) = \frac{(x+y)(x+y+1)}{2} + x$$

問題 数学 $\boxed{\mathbf{2}}$ **(1)** は，数学 $\boxed{\mathbf{2}}$ **(2)** とともに解答すること．　この解答用紙の先頭に,「数学 $\boxed{\mathbf{2}}$ **(1)**」と明記し，それ以外の問題を解答しないこと．

問題 数学 $\boxed{\mathbf{2}}$ **(1)**

以下の問いに答えなさい．ただし行列 I，行列 E，および行列 O は次のように与えられるとする．

$$I = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix}, \qquad E = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad O = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

(a) I^2 を求めなさい．

(b) 行列 A が次のように与えられる．

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} E & I \\ I & E \end{pmatrix}$$

このとき，$A^2 = 2\begin{pmatrix} O & I \\ I & O \end{pmatrix}$ であることを示しなさい．

(c) A^4 を求めなさい．

(d) A^6 を求めなさい．

(e) A^8 を求めなさい．

(f) $A^{(2n)}$ を求めなさい．ただし n は 1 以上の整数とする．

問題 数学 $\boxed{2}$ (2) は，数学 $\boxed{2}$ **(1)** とともに解答すること． この解答用紙の先頭に,「数学 $\boxed{2}$ **(2)**」と明記し，それ以外の問題を解答しないこと．

問題 数学 $\boxed{2}$ (2)

ガウス積分は関数 e^{-x^2} の実数値全域での広義積分として以下のように定義される．

$$G = \int_{-\infty}^{+\infty} e^{-x^2} dx$$

次の問題に答えなさい．すべての問題で計算過程を示すこと．上記の G と問題 (a) の H は広義積分可能であると仮定してよい．

(a) $f(x, y) = e^{-(x^2+y^2)}$ とした場合，以下で定義される H が G^2 と等しいことを示しなさい．

$$H = \iint_{\mathbb{R}^2} f(x, y) dx dy = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) dx dy$$

(b) 極座標-直交座標変換のヤコビ行列式

$$J(r, \theta) = \begin{vmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \theta} \\ \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \theta} \end{vmatrix}$$

を求めなさい．ただし，極座標 (r, θ) から直交座標 (x, y) への変換は次のように与えられるものとする．

$$\begin{aligned} x &= r \cos \theta \\ y &= r \sin \theta \end{aligned}$$

(c) 極座標-直交座標変換を用いて H の値を計算せよ．問題 (a) で定義された $f(x, y)$ について以下の式が成り立つことを利用してよい．

$$\iint_{\mathbb{R}^2} f(x, y) dx dy = \iint_{\Omega} f(r \cos \theta, r \sin \theta) \left| J(r, \theta) \right| dr d\theta$$

ただし，Ω は変数変換後の積分範囲である．

(d) G の値を求めよ．

問題 情報 $\boxed{\mathbf{1}}$ **(1)** は，情報 $\boxed{\mathbf{1}}$ **(2)** とともに解答すること． 情報 $\boxed{\mathbf{1}}$ を選択した場合は，数学 $\boxed{\mathbf{1}}$ を選択しないこと． この解答用紙の先頭に，「情報 $\boxed{\mathbf{1}}$ **(1)**」と明記し，それ以外の問題を解答しないこと．

問題 情報 $\boxed{\mathbf{1}}$ (1)

正の整数を正の整数に対応付ける関数 **fib** を次の式で定義する．

$$\mathbf{fib}(n) = \begin{cases} 1 & n \leq 2 \ \text{のとき} \\ \mathbf{fib}(n-2) + \mathbf{fib}(n-1) & n > 2 \ \text{のとき} \end{cases}$$

(a) $\mathbf{fib}(15)$ および $\mathbf{fib}(16)$ の値を求めよ．

(b) 0 以上の整数 a と正の整数 d に対して，a を d で割った余りを $a \ \mathbf{mod} \ d$ と表すとき，$(\mathbf{fib}(16))^2 \ \mathbf{mod} \ 10$ の値を求めよ．ただし，0 以上の整数 x, y と正の整数 d に対して，

$$(x \cdot y) \ \mathbf{mod} \ d = ((x \ \mathbf{mod} \ d) \cdot (y \ \mathbf{mod} \ d)) \ \mathbf{mod} \ d$$

が成立することを使ってよい．

(c) $\mathbf{fib}(256) \ \mathbf{mod} \ 10$ を求めよ．ただし，以下の公式がすべての $n \geq 2$ に対して成立することを使ってよい．

$$\begin{aligned} \mathbf{fib}(2n-1) &= (\mathbf{fib}(n-1))^2 + (\mathbf{fib}(n))^2 \\ \mathbf{fib}(2n) &= 2 \cdot \mathbf{fib}(n-1) \cdot \mathbf{fib}(n) + (\mathbf{fib}(n))^2 \end{aligned}$$

(d) $\mathbf{fib}(2^{100}) \ \mathbf{mod} \ 10$ の値を求めよ．

問題 情報 $\boxed{1}$ **(2)** は，情報 $\boxed{1}$ (1) とともに解答すること． 情報 $\boxed{1}$ を選択した場合は，数学 $\boxed{1}$ を選択しないこと． この解答用紙の先頭に,「情報 $\boxed{1}$ **(2)**」と明記し，それ以外の問題を解答しないこと.

問題 情報 $\boxed{1}$ **(2)**

原子命題 P と Q，含意記号 $\Rightarrow$ (ならば)，および，括弧だけで構成される論理式について考える.

論理式 A と B が同値であるとは，原子命題 P と Q のそれぞれに，T (真) または F (偽) を割り当てた時の A と B の真理値が，いつでも一致することである.

(a) 論理式 $(P \Rightarrow Q) \Rightarrow (Q \Rightarrow P)$ の真理値表を示せ.

(b) 論理式 $(P \Rightarrow Q) \Rightarrow Q$ と論理式 $(Q \Rightarrow P) \Rightarrow P$ が同値であることを示せ.

(c) $S = \{P, Q, P \Rightarrow P, P \Rightarrow Q, Q \Rightarrow P\}$ とする．この集合 S の任意の要素 X に対して，論理式 $P \Rightarrow X$ は，集合 S のいずれかの要素と同値であることを示せ.

(d) 集合 S に属する要素 X, Y から論理式 $X \Rightarrow Y$ を作る．論理式 $X \Rightarrow Y$ が論理式 $(P \Rightarrow Q) \Rightarrow Q$ でも論理式 $(Q \Rightarrow P) \Rightarrow P$ でもないとき， $X \Rightarrow Y$ は，S のいずれかの要素と同値となることを示せ.

問題 情報 $\boxed{2}$ (1) は，情報 $\boxed{2}$ (2) とともに解答すること．　この解答用紙の先頭に，「情報 $\boxed{2}$ (1)」と明記し，それ以外の問題を解答しないこと．

問題 情報 $\boxed{2}$ (1)

以下の関数により正の整数の追加と取り出しができる一種の優先度キュー (priority queue) を C 言語で構造体 pq として実装することを考える．

- `struct pq *newq()`

  新たに優先度キューを作成して初期化し，そのキューへのポインタを返す．

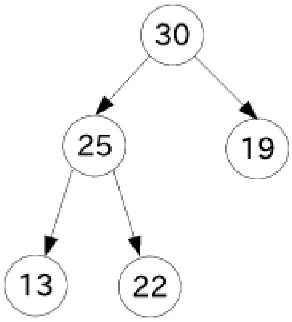
- `void putq(struct pq *q, int v)`

  正の整数 v を優先度キュー q に追加する．

- `int getq(struct pq *q)`

  優先度キュー q に格納されている整数のうち最も値が大きいものを一つキューから取り出し，その値を返す．ただし，キューが空の場合には，この関数は -1 を返す．

ここではこの優先度キューを，以下のような完全 2 分木のデータ構造を用いて実現する．



上図の各ノード (○) の中に書かれている数値が，優先度キューに格納される数値である．この完全 2 分木には，以下のような条件を満足するように数値が格納されている．

- 各ノードが持つ数値は，そのノードのいずれの子ノードが持つ数値より小さくない．

(a) 上記の条件で n 個の正の整数が格納された完全 2 分木を考える．この n 個の正の整数を，その根から幅優先で走査した順に配列の要素 a[1]～a[n] に格納したとする．例えば，上図の完全 2 分木であれば，数値は 30, 25, 19, 13, 22 の順に配列の要素 a[1]～a[5] に格納される．

以下の空欄 $\boxed{\text{(A)}}$ ～ $\boxed{\text{(C)}}$ を埋めて文章を完成させなさい．

・整数 i ($1 \leq \mathtt{i} < \frac{n}{2}$) に対して，a[i] の子ノードの数値は $\boxed{\text{(A)}}$ と $\boxed{\text{(B)}}$ に格納される．

・配列 a[1]～a[n] で最も大きな数値は $\boxed{\text{(C)}}$ に格納される．

(b) 以下のプログラムは上記の (a) に示した配列 a を用いて構造体 pq と関数 newq，putq，getq を実装したものである．ただし、a[0] には定数 INT_MAX の値を代入している．この定数 INT_MAX の値は優先度キューに追加されるどの整数よりも大きな値であるものとする．また，以下のプログラムで定数 SIZE の値は十分に大きく，優先度キューに格納される整数の個数がこの値を超えることはないものとする．

$\boxed{\text{(D)}}$ ～ $\boxed{\text{(F)}}$ を埋めてこれらの関数を完成させなさい．

```
1  struct pq {
2    int n;
3    int a[SIZE];
4  };
5
6  struct pq *newq()
7  {
8    struct pq *q = malloc(sizeof(struct pq));
9    q->n = 0;
10   q->a[0] = INT_MAX;
11   return q;
12 }
13
14 void putq(struct pq *q, int v)
15 {
16   int i = ++(q->n);
17   while (q->a[i/2] <= v) {
18     q->a[i] = q->a[   (D)   ];
19     i = i / 2;
20   }
21   q->a[i] = v;
22 }
23
24 int getq(struct pq *q)
25 {
26   int i = 1, j, v, w;
27   if (q->n == 0) {
28     return    (E)   ;
29   }
30   v = q->a[1];
31   w = q->a[(q->n)--];
32   while (i <= (q->n) / 2) {
33     j = 2 * i;
34
35     if (j < q->n && q->a[j] < q->a[j+1]) j++;
36
37     if (w >= q->a[j]) {
38       break;
39     } else {
40          (F)   ;
41       i = j;
42     }
43   }
44   q->a[i] = w;
45   return v;
46 }
```

(c) n 個の整数が格納されている優先度キュー q を引数として関数 getq を呼び出したときの最悪の場合の漸近的な時間計算量を以下から選びなさい．

(i) 時間計算量は $O(1)$ である．

(ii) 時間計算量は $O(\log n)$ であり，$O(1)$ ではない．

(iii) 時間計算量は $O(n)$ であり，$O(\log n)$ ではない．

(iv) 時間計算量は $O(n \log n)$ であり，$O(n)$ ではない．

(v) 時間計算量は $O(n^2)$ であり，$O(n \log n)$ ではない．

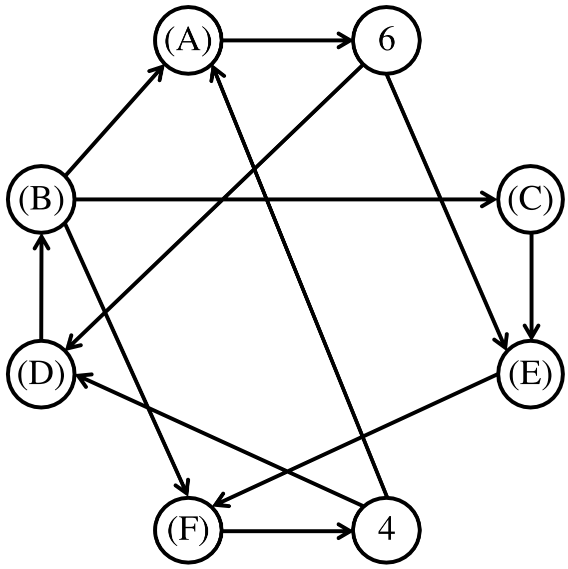
(d) プログラムの 10 行目では，q->a[0] に INT_MAX が代入されている．もし INT_MAX の代わりに q->a[0] に 0 が代入されたとするとどのような問題が生じるか，説明しなさい．

問題 情報 $\boxed{2}$ (2) は，情報 $\boxed{2}$ (1) とともに解答すること． この解答用紙の先頭に，「情報 $\boxed{2}$ (2)」と明記し，それ以外の問題を解答しないこと．

問題 情報 $\boxed{2}$ (2)

次ページに掲載されている C 言語で記述されたプログラムリストについての以下の設問に答えなさい．

(a) プログラムリストの 3 行目から 12 行目にある 2 次元配列 `adj_mat` は，頂点 0, 1, ..., 7 の 8 つの頂点を持つ重みなし有向グラフを示す隣接行列である．グラフの頂点番号は配列の添え字に対応し，`adj_mat[i][j]` が 1 のとき，頂点 `i` から頂点 `j` への辺が存在することを表している．下図は，この `adj_mat` で表現されているグラフを図示したものである．(A) から (F) までに入る適切な整数を答えよ．



(b) プログラムリストの 14 行目から定義されている関数 `calc_dists` は，`adj_mat` で定義されたグラフの指定された頂点 (`origin`) から，グラフ上のすべての頂点への距離を計算する．ここで，頂点 `i` から頂点 `j` までの距離とは，`i` から `j` へ到達するために通らなくてはならない辺の数の最小値である．(a) のグラフにおいて，`origin` を 0 としたときの，頂点 0, 1, ..., 7 への距離を示せ．

(c) プログラムリストの 20 行目から 28 行目では，`adj_mat` を隣接リストの形に変換している．この処理を実行させた後に配列 `adj_list` および `adj_index` に代入されている値を書け．但し，値が初期化されていない要素は書かないこと．

(d) プログラムリスト 30 行目から 52 行目では，配列 `adj_list` および `adj_index` を用いて，`origin` から頂点 `i`（i=0, 1, ..., 7）までの距離を `origin` からの幅優先探索で求め，`dist_vec[i]` に格納している．但し，`origin` から頂点 `i` に到達できない場合は，`dist_vec[i]` の値は `UNREACH` とする．`(G)` から `(M)` に入るべき式を入れて，プログラムを完成させよ．

```
#define UNREACH -1
#define N_VERT 8
const int adj_mat[N_VERT][N_VERT] = {
  {0, 1, 0, 1, 0, 1, 0, 0},
  {0, 0, 1, 0, 0, 0, 0, 0},
  {0, 0, 0, 1, 0, 0, 0, 0},
  {0, 0, 0, 0, 1, 0, 0, 0},
  {0, 0, 0, 0, 0, 1, 0, 1},
  {0, 0, 0, 0, 0, 0, 1, 0},
  {0, 0, 1, 0, 0, 0, 0, 1},
  {1, 0, 0, 0, 0, 0, 0, 0},
};

void calc_dists(const int origin, int dist_vec[]) {
  int adj_list[N_VERT * N_VERT], adj_index[N_VERT + 1];
  int array1[N_VERT], array2[N_VERT];
  int *curr = array1, *next = array2, *tmp;
  int i, j, index = 0, dist = 1, len_curr = 1, len_next = 0;

  for (i = 0; i < N_VERT; i++) {
    adj_index[i] = index;
    for (j = 0; j < N_VERT; j++) {
      if (adj_mat[i][j] == 1) {
        adj_list[index++] = j;
      }
    }
  }
  adj_index[N_VERT] = index;

  for (i = 0; i < N_VERT; i++) {
    dist_vec[i] = UNREACH;
  }

  curr[0] = origin;
  dist_vec[origin] = 0;
  while (len_curr > 0) {
    for (i = 0; i < len_curr; i++) {
      for (j = [    (G)    ]; j < [    (H)    ]; j++) {
        if ([    (I)    ] == [    (J)    ]) {
          [    (K)    ] = dist;
          [    (L)    ] = [    (M)    ];
          len_next++;
        }
      }
    }
    tmp = next;
    next = curr;
    curr = tmp;
    len_curr = len_next;
    len_next = 0;
    dist++;
  }
}
```

Select **Problem Math** $\boxed{\mathbf{1}}$ **(1)** with **Math** $\boxed{\mathbf{1}}$ **(2)**. If you select **Math** $\boxed{\mathbf{1}}$, then do not select **CS** $\boxed{\mathbf{1}}$. Clearly label the answer sheet at the top of the page as "**Math** $\boxed{\mathbf{1}}$ **(1)**" and do not write answers to other problems on the answer sheet.

Problem Math $\boxed{\mathbf{1}}$ **(1)**

We define the function **fib** which maps a positive integer to a positive integer as follows:

$$\mathbf{fib}(n) = \begin{cases} 1 & \text{if } n \leq 2 \\ \mathbf{fib}(n-2) + \mathbf{fib}(n-1) & \text{if } n > 2 \end{cases}$$

(a) Compute the values of $\mathbf{fib}(15)$ and $\mathbf{fib}(16)$.

(b) Let a **mod** d be the remainder after division of a by d for a non-negative integer a and a positive integer d. Find the value of $(\mathbf{fib}(16))^2 \ \mathbf{mod}\ 10$.

You may use the fact that, for non-negative integers x and y and a positive integer d, the following equation holds.

$$(x \cdot y) \ \mathbf{mod}\ d = ((x \ \mathbf{mod}\ d) \cdot (y \ \mathbf{mod}\ d)) \ \mathbf{mod}\ d$$

(c) Find the value of $\mathbf{fib}(256) \ \mathbf{mod}\ 10$.

You may use the following formulae for any $n \geq 2$:

$$\begin{aligned} \mathbf{fib}(2n-1) &= (\mathbf{fib}(n-1))^2 + (\mathbf{fib}(n))^2 \\ \mathbf{fib}(2n) &= 2 \cdot \mathbf{fib}(n-1) \cdot \mathbf{fib}(n) + (\mathbf{fib}(n))^2 \end{aligned}$$

(d) Find the value of $\mathbf{fib}(2^{100}) \ \mathbf{mod}\ 10$.

Select **Problem Math** $\boxed{1}$ **(2)** with **Math** $\boxed{1}$ **(1)**. If you select **Math** $\boxed{1}$, then do not select **CS** $\boxed{1}$. Clearly label the answer sheet at the top of the page as "**Math** $\boxed{1}$ **(2)**" and do not write answers to other problems on the answer sheet.

Problem Math $\boxed{1}$ **(2)**

Suppose n is a positive integer. Let $\mathbf{T}_n$ be the set of all pairs (x, y) such that x and y are non-negative integers and $x + y < n$, and $\mathbf{N}_n$ be the set of all integers z such that $0 \leq z < \dfrac{n(n+1)}{2}$.

We shall consider, for each f defined below, whether the function $f : \mathbf{T}_n \to \mathbf{N}_n$ is injective, namely, whether the following condition (*) holds.

(*) For any $(x_1, y_1) \in \mathbf{T}_n$ and any $(x_2, y_2) \in \mathbf{T}_n$, if $f(x_1, y_1) = f(x_2, y_2)$, then $x_1 = x_2$ and $y_1 = y_2$ hold.

Answer the following questions.

(a) Let $n = 2$ and define f by $f(x, y) = 2x + y$. Compute the values of $f(0, 0)$, $f(0, 1)$, and $f(1, 0)$. Show that this f is injective.

(b) Suppose $n = 4$, and define f by $f(x, y) = \dfrac{(x+y)(x+y+1)}{2} + x$. Compute the values of $f(0, 3)$, $f(1, 2)$, $f(2, 1)$, and $f(3, 0)$.

(c) Show that, for any non-negative integers z_1 and z_2, if $z_1 \geq z_2 + 1$ holds, then the following inequality holds:

$$\frac{z_1(z_1+1)}{2} \geq \frac{z_2(z_2+1)}{2} + z_2 + 1$$

(d) Suppose $n > 0$. Show that the function $f : \mathbf{T}_n \to \mathbf{N}_n$ defined below is injective.

$$f(x, y) = \frac{(x+y)(x+y+1)}{2} + x$$

Select **Problem Math** $\boxed{\mathbf{2}}$ **(1)** with **Math** $\boxed{\mathbf{2}}$ **(2)**. Clearly label the answer sheet at the top of the page as "**Math** $\boxed{\mathbf{2}}$ **(1)**" and do not write answers to other problems on the answer sheet.

Problem Math $\boxed{\mathbf{2}}$ **(1)**

Answer the following questions when I, E, and O are given as follows.

$$I = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix}, \qquad E = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad O = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

(a) Find I^2.

(b) Let us assume that A is given as follows.

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} E & I \\ I & E \end{pmatrix}$$

Show that $A^2 = 2\begin{pmatrix} O & I \\ I & O \end{pmatrix}$.

(c) Find A^4.

(d) Find A^6.

(e) Find A^8.

(f) Find $A^{(2n)}$ when n is an integer that is larger than or equal to 1.

Select **Problem Math** $\boxed{\mathbf{2}}$ **(2)** with **Math** $\boxed{\mathbf{2}}$ **(1)**. Clearly label the answer sheet at the top of the page as "**Math** $\boxed{\mathbf{2}}$ **(2)**" and do not write answers to other problems on the answer sheet.

Problem Math $\boxed{\mathbf{2}}$ **(2)**

The Gaussian integral is defined as the improper integral of the function e^{-x^2} over all real values, that is

$$G = \int_{-\infty}^{+\infty} e^{-x^2} dx.$$

Answer the following questions. Show your steps in all answers of the questions. You may assume that the improper integrals of G above and of H (in question (a)) converge.

(a) If $f(x, y) = e^{-(x^2+y^2)}$, show that H defined by the following is equal to G^2.

$$H = \iint_{\mathbb{R}^2} f(x, y) dx dy = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) dx dy$$

(b) Compute the Jacobian determinant,

$$J(r, \theta) = \begin{vmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \theta} \\ \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \theta} \end{vmatrix},$$

for the polar-Cartesian transformation from polar coordinates (r, θ) to Cartesian coordinates (x, y), where the transformation is given by the following functions:

$$\begin{aligned} x &= r \cos \theta, \\ y &= r \sin \theta. \end{aligned}$$

(c) Compute the value of H using polar-Cartesian transformation. You can use the following equation related to $f(x, y)$ defined by question (a).

$$\iint_{\mathbb{R}^2} f(x, y) dx dy = \iint_{\Omega} f(r \cos \theta, r \sin \theta) \left| J(r, \theta) \right| dr d\theta,$$

where Ω is the integral range after the transformation of variables.

(d) Compute the value of G.

Select **Problem CS** $\boxed{1}$ **(1)** with **CS** $\boxed{1}$ **(2)**. If you select **CS** $\boxed{1}$, then do not select **Math** $\boxed{1}$. Clearly label the answer sheet at the top of the page as "**CS** $\boxed{1}$ **(1)**" and do not write answers to other problems on the answer sheet.

Problem CS $\boxed{1}$ **(1)**

We define the function **fib** which maps a positive integer to a positive integer as follows:

$$\mathbf{fib}(n) = \begin{cases} 1 & \text{if } n \leq 2 \\ \mathbf{fib}(n-2) + \mathbf{fib}(n-1) & \text{if } n > 2 \end{cases}$$

(a) Compute the values of $\mathbf{fib}(15)$ and $\mathbf{fib}(16)$.

(b) Let a **mod** d be the remainder after division of a by d for a non-negative integer a and a positive integer d. Find the value of $(\mathbf{fib}(16))^2$ **mod** 10.

You may use the fact that, for non-negative integers x and y and a positive integer d, the following equation holds.

$$(x \cdot y) \mathbf{\ mod\ } d = ((x \mathbf{\ mod\ } d) \cdot (y \mathbf{\ mod\ } d)) \mathbf{\ mod\ } d$$

(c) Find the value of $\mathbf{fib}(256)$ **mod** 10.

You may use the following formulae for any $n \geq 2$:

$$\begin{aligned} \mathbf{fib}(2n-1) &= (\mathbf{fib}(n-1))^2 + (\mathbf{fib}(n))^2 \\ \mathbf{fib}(2n) &= 2 \cdot \mathbf{fib}(n-1) \cdot \mathbf{fib}(n) + (\mathbf{fib}(n))^2 \end{aligned}$$

(d) Find the value of $\mathbf{fib}(2^{100})$ **mod** 10.

Select **Problem CS** $\boxed{\mathbf{1}}$ **(2)** with **CS** $\boxed{\mathbf{1}}$ **(1)**. If you select **CS** $\boxed{\mathbf{1}}$, then do not select **Math** $\boxed{\mathbf{1}}$. Clearly label the answer sheet at the top of the page as "**CS** $\boxed{\mathbf{1}}$ **(2)**" and do not write answers to other problems on the answer sheet.

Problem CS $\boxed{\mathbf{1}}$ **(2)**

Consider the logical formulae built from the atomic formulae P and Q, implication sign $\Rightarrow$, and parentheses.

We say that the formulae A and B are equivalent if and only if their truth values agree for every assignment of the atomic formulae P and Q to T (true) or F (false).

(a) Describe the truth table of the formula $(P \Rightarrow Q) \Rightarrow (Q \Rightarrow P)$.

(b) Show that the formulae $(P \Rightarrow Q) \Rightarrow Q$ and $(Q \Rightarrow P) \Rightarrow P$ are equivalent.

(c) Let $S = \{P, Q, P \Rightarrow P, P \Rightarrow Q, Q \Rightarrow P\}$. Show that, for any X in the set S, the formula $P \Rightarrow X$ is equivalent to some element of S.

(d) For elements X and Y in the set S, we build the formula $X \Rightarrow Y$. Show that, if the formula $X \Rightarrow Y$ is neither $(P \Rightarrow Q) \Rightarrow Q$ or $(Q \Rightarrow P) \Rightarrow P$, $X \Rightarrow Y$ is equivalent to some element in S.

Select **Problem CS** $\boxed{\mathbf{2}}$ **(1)** with **CS** $\boxed{\mathbf{2}}$ **(2)**. Clearly label the answer sheet at the top of the page as "**CS** $\boxed{\mathbf{2}}$ **(1)**" and do not write answers to other problems on the answer sheet.

Problem CS $\boxed{\mathbf{2}}$ (1)

Let us consider a kind of priority queues in which we can store and get positive integers using the following functions. We implement the queue as `struct pq` written in the C language.

- `struct pq *newq()`

  Creates a new priority queue, initializes it, and returns a pointer to the queue.

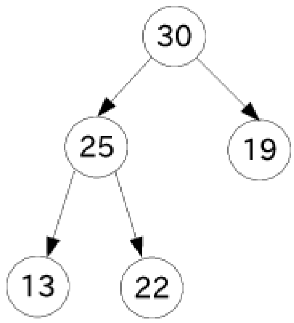
- `void putq(struct pq *q, int v)`

  Stores the positive integer value `v` into the priority queue `q`.

- `int getq(struct pq *q)`

  Removes one of the largest integer values stored in the priority queue `q`, and returns the value. This function returns -1 when the queue is empty.

In the following, we implement this priority queue with a complete binary tree as illustrated below.



In the above figure, numbers in the nodes (○) are the values stored in the priority queue. In this complete binary tree, values are stored as the following condition is satisfied.

- The value stored in each node is either greater than or equal to any value stored in its child nodes.

(a) Let us consider that n positive integers are stored in the complete binary tree under the above condition. Suppose that we store the n positive integers into elements of array `a[1]` to `a[n]` in the order as we traverse in the breadth first traversal order. For example, for the complete binary tree in the above figure, we store the values into elements of array `a[1]` to `a[5]` in the order of 30, 25, 19, 13, and 22.

Fill in the blanks $\boxed{\text{(A)}}$ to $\boxed{\text{(C)}}$ in the following descriptions.

- For the integer `i` ($1 \leq$ `i` $< \frac{n}{2}$), the values of the node `a[i]`'s child nodes are stored in $\boxed{\text{(A)}}$ and $\boxed{\text{(B)}}$.
- The largest value in the array `a[1]`～`a[n]` is stored in $\boxed{\text{(C)}}$.

(b) The following program implements the structure `pq` and the functions `newq`, `putq`, and `getq` with the array `a` explained in (a). Note that the value of the constant `INT_MAX` is assigned to `a[0]`. You may assume that the value of the constant `INT_MAX` is greater than any integers stored in the priority queue. Also you may assume that `SIZE` is a sufficiently large constant, and that the number of integers stored in the priority queue never exceeds the value of `SIZE`.

Fill in the blanks $\boxed{\text{(D)}}$ to $\boxed{\text{(F)}}$ to complete the functions.

```
struct pq {
  int n;
  int a[SIZE];
};

struct pq *newq()
{
  struct pq *q = malloc(sizeof(struct pq));
  q->n = 0;
  q->a[0] = INT_MAX;
  return q;
}

void putq(struct pq *q, int v)
{
  int i = ++(q->n);
  while (q->a[i/2] <= v) {
    q->a[i] = q->a[    (D)    ];
    i = i / 2;
  }
  q->a[i] = v;
}

int getq(struct pq *q)
{
  int i = 1, j, v, w;
  if (q->n == 0) {
    return     (E)    ;
  }
  v = q->a[1];
  w = q->a[(q->n)--];
  while (i <= (q->n) / 2) {
    j = 2 * i;

    if (j < q->n && q->a[j] < q->a[j+1]) j++;

    if (w >= q->a[j]) {
      break;
    } else {
          (F)    ;
      i = j;
    }
  }
  q->a[i] = w;
  return v;
}
```

(c) Choose the correct option describing the asymptotic time complexity in the worst case when the function `getq` is called with a priority queue `q` in which n integers are stored.

(i) The time complexity is $O(1)$.

(ii) The time complexity is $O(\log n)$, and is not $O(1)$.

(iii) The time complexity is $O(n)$, and is not $O(\log n)$.

(iv) The time complexity is $O(n \log n)$, and is not $O(n)$.

(v) The time complexity is $O(n^2)$, and is not $O(n \log n)$.

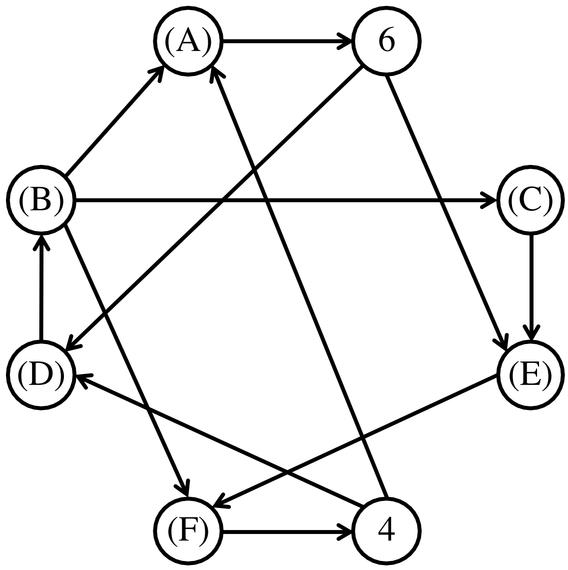
(d) On the line number 10 of the program, `INT_MAX` is assigned to `q->a[0]`. Explain what problem(s) may arise if 0 is assigned to `q->a[0]` instead of `INT_MAX`.

Select **Problem CS** $\boxed{2}$ **(2)** with **CS** $\boxed{2}$ **(1)**. Clearly label the answer sheet at the top of the page as "**CS** $\boxed{2}$ **(2)**" and do not write answers to other problems on the answer sheet.

Problem CS $\boxed{2}$ **(2)**

Let us consider the program list written in C language in the following page. Answer the following questions.

(a) The 2-dimentional array `adj_mat` in the program list between lines 3 and 12 is an unweighted directed graph with eight vertexes labeled as 0, 1, ..., 7. The vertex number corresponds to the array's index. When `adj_mat[i][j]` is 1, there exists an edge from vertex `i` to vertex `j`. The following diagram depicts the graph represented by `adj_mat`. Find the appropriate labels of vertexes for (A) to (F).



(b) The function `calc_dists` defined from line 14 in the program list calculates the distances from a given vertex (`origin`) to every vertex in the graph defined by `adj_mat`. The distance from vertex `i` to vertex `j` is the minimum number of edges required to traverse from `i` to `j`. When `origin` is 0 in the graph in (a), show all distances to vertexes 0, 1, ..., 7.

(c) The lines 20 to 28 in the program list converts `adj_mat` into an adjacency list. Show the contents of the arrays `adj_list` and `adj_index` after the process is executed. You do not show any contents that are not initialized.

(d) The lines 30 to 52 in the program list calculates the distances from `origin` to all vertexes `i` (`i`=0, 1, ..., 7) by the breadth-first search and stores them into `dist_vec[i]`, where `dist_vec[i]` is `UNREACH` if the vertex `i` is unreachable from `origin`. Fill the blanks `(G)` to `(M)` with appropriate expressions to complete the program.

```c
#define UNREACH -1
#define N_VERT 8
const int adj_mat[N_VERT][N_VERT] = {
  {0, 1, 0, 1, 0, 1, 0, 0},
  {0, 0, 1, 0, 0, 0, 0, 0},
  {0, 0, 0, 1, 0, 0, 0, 0},
  {0, 0, 0, 0, 1, 0, 0, 0},
  {0, 0, 0, 0, 0, 1, 0, 1},
  {0, 0, 0, 0, 0, 0, 1, 0},
  {0, 0, 1, 0, 0, 0, 0, 1},
  {1, 0, 0, 0, 0, 0, 0, 0},
};

void calc_dists(const int origin, int dist_vec[]) {
  int adj_list[N_VERT * N_VERT], adj_index[N_VERT + 1];
  int array1[N_VERT], array2[N_VERT];
  int *curr = array1, *next = array2, *tmp;
  int i, j, index = 0, dist = 1, len_curr = 1, len_next = 0;

  for (i = 0; i < N_VERT; i++) {
    adj_index[i] = index;
    for (j = 0; j < N_VERT; j++) {
      if (adj_mat[i][j] == 1) {
        adj_list[index++] = j;
      }
    }
  }
  adj_index[N_VERT] = index;

  for (i = 0; i < N_VERT; i++) {
    dist_vec[i] = UNREACH;
  }

  curr[0] = origin;
  dist_vec[origin] = 0;
  while (len_curr > 0) {
    for (i = 0; i < len_curr; i++) {
      for (j = [ (G) ]; j < [ (H) ]; j++) {
        if ([ (I) ] == [ (J) ]) {
          [ (K) ] = dist;
          [ (L) ] = [ (M) ];
          len_next++;
        }
      }
    }
    tmp = next;
    next = curr;
    curr = tmp;
    len_curr = len_next;
    len_next = 0;
    dist++;
  }
}
```