

修士課程 社会情報学専攻入学者選抜試験問題  
(情報学基礎)

Entrance Examination for Master's Program  
(Fundamentals of Informatics)

Department of Social Informatics

平成 25 年 2 月 12 日 10 : 00 ~ 12 : 00

February 12, 2013 10:00 - 12:00

【注意】

- ・ 試験開始の合図があるまで中を見てはいけない。
- ・ 問題用紙は表紙を含めて 6 枚である。試験開始後、枚数を確認し、落丁または印刷の不鮮明なものがあれば直ちに申し出ること。
- ・ 問題は 5 題である。このうち 3 題を選択し、解答しなさい。
- ・ 解答用紙の表紙に記載されている注意事項についても留意すること。
- ・ 問題 1 題につき、解答用紙 1 枚を使用すること。解答用紙は裏面を使用しても構わないが、使用する場合は裏面に継続することを明記すること。

NOTES

- ・ Do not open the pages before a call of starting.
- ・ This is the Question Booklet in 6 pages including this front cover.  
After the call of starting, check all pages are in order and notify proctors immediately if missing pages or unclear printings are found.
- ・ There are 5 questions. Choose and answer 3 questions in total.
- ・ Read carefully the notes on the front cover of the Answer Sheets, too.
- ・ Use one sheet for each question. If necessary, the reverse side may be used, stating "Over" at the end of the page.

【情報学基礎】問題番号：I

以下のプログラム Prog0 に関する設問に答えよ。

Answer the following questions related to the program Prog0.

```
1:  program Prog0;
2:  var
3:      data1, data2: real;
4:  begin
5:      readln(data1);
6:      readln(data2);
7:      data1 := data1 + data2;
8:      data1 := data1 - data2;
9:      writeln(data1);
10: end.
```

- (1) 以下の全ての語を用いながら、プログラム Prog0 の全ての行について詳しく説明せよ。

語：コンピュータのメモリ、データの型、メモリ内の記憶場所、変数

Explain the details of all the lines in the program Prog0. Note that all of the following terms must be used.

Terms: computer memory, data type, locations in memory, variable

- (2) プログラム Prog0 を実行し、data1 に対して 1000 を入力し、data2 に対して 100000 を入力したところ、このプログラムは「1.0000000000E+03」という表示を出力した。この出力が、何を意味しているのか詳しく説明せよ。

When the program Prog0 is executed with an input for data1 being 1000 and an input for data2 being 100000, the program shows “1.0000000000E+03” as the output. Give a detailed description what the output means.

- (3) プログラム Prog0 を実行し、data1 に対して 1000 を入力し、data2 に対して 10000000000000000000 を入力したところ、このプログラムは「0.0000000000E+00」という表示を出力した。このときのプログラムの動作を説明せよ。そして、この出力がなされた理由を明らかにせよ。

When the program Prog0 is executed with an input for data1 being 1000 and an input for data2 being 10000000000000000000, the program shows “0.0000000000E+00” as the output. Explain how the program works in this case, and clarify the reason why the output is shown.

【情報学基礎】問題番号：Ⅱ

次のそれぞれの集合について、可算 (countable) であるか非可算 (uncountable) であるか述べよ。また、その理由について十分な説明を与えよ。

For each of the following sets, decide whether the set is countable or uncountable. Give a full explanation in each case.

- (1) Pascal のプログラムの集合.

The set of Pascal programs.

- (2) 入力の一つの正の整数であり、出力が一つの正の整数である関数の集合.

The set of functions that input a positive integer and output a positive integer.

- (3) 2 つの自然数の組の集合.

The set of tuples of two natural numbers.

---

【情報学基礎】問題番号：III

以下の生成規則 R1, R2, R3, R4, R5 を用いて, 次の文

$$UI := X + Y * Z$$

を P88 マシンのアセンブリ言語に翻訳せよ. P88 マシンの命令セットは表 1 に示す.

Use the production rules R1, R2, R3, R4 and R5 shown below to translate the following statement into the assembly language of P88 Machine. The instructions of P88 Machine are shown in Table 1.

$$UI := X + Y * Z$$

構文(Syntax)	意味(Semantics)
R1: $\langle i \rangle_j \rightarrow w$	$M(\langle i \rangle_j) = w$
R2: $\langle e \rangle_i \rightarrow \langle i \rangle_j$	$M(\langle e \rangle_i) = M(\langle i \rangle_j)$ $code(\langle e \rangle_i) = \text{nothing}$
R3: $\langle s \rangle_k \rightarrow \langle i \rangle_j := \langle e \rangle_i$	$code(\langle s \rangle_k) = code(\langle e \rangle_i)$ COPY AX, M( $\langle e \rangle_i$ ) COPY M( $\langle i \rangle_j$ ), AX
R4: $\langle e \rangle_i \rightarrow (\langle e \rangle_j + \langle e \rangle_k)$	$M(\langle e \rangle_i) = \text{createnam}$ $code(\langle e \rangle_i) = code(\langle e \rangle_j)$ code( $\langle e \rangle_k$ ) COPY AX, M( $\langle e \rangle_j$ ) ADD AX, M( $\langle e \rangle_k$ ) COPY M( $\langle e \rangle_i$ ), AX
R5: $\langle e \rangle_i \rightarrow (\langle e \rangle_j * \langle e \rangle_k)$	$M(\langle e \rangle_i) = \text{createnam}$ $code(\langle e \rangle_i) = code(\langle e \rangle_j)$ code( $\langle e \rangle_k$ ) COPY AX, M( $\langle e \rangle_j$ ) MUL AX, M( $\langle e \rangle_k$ ) COPY M( $\langle e \rangle_i$ ), AX

ただし,  $\langle i \rangle_j, \langle e \rangle_i, \langle e \rangle_j, \langle e \rangle_k, \langle s \rangle_k$  は文法変数である.  $w$  は識別子の文字列である.  $M$  は意味関数である.

where,  $\langle i \rangle_j, \langle e \rangle_i, \langle e \rangle_j, \langle e \rangle_k$ , and  $\langle s \rangle_k$  are grammar variables.

$w$  is some identifier string. The notation  $M$  stands for the concept of "meaning".

$\langle s \rangle_k \rightarrow$

$$M(\langle e \rangle_i) = M(\langle i \rangle_j)$$

$$code(\langle e \rangle_i) = \text{nothing}$$

表 1 命令セット

TABLE 1: Instruction Set

命令 Instruction	フォーマット Format	アクション Action
Copy from mem	COPY AX, mem	AX:=mem
Copy to mem	COPY mem, AX	mem:=AX
Add	ADD AX, mem	AX:=AX+mem
Subtract	SUB AX, mem	AX:=AX-mem
Multiply	MUL AX, mem	AX:=AX*mem
Divide	DIV AX, mem	AX:=AX div mem
Compare	CMP AX, mem	if AX < mem then CF:=B else CF:=NB
Jump	JMP lab1	Go to the instruction with label lab1
Jump if not below	JNB lab1	Go to the instruction with label lab1 if CF=NB. Otherwise go to next instruction
Jump if below	JB lab1	Go to the instruction with label lab1 if CF=B. Otherwise go to next instruction
Input	IN AX	Input an integer into register AX
Output	OUT AX	Output an integer from register AX

COPY AX, Y

MUL AX, Z

COPY CN2, AX

COPY AX, X

ADD AX, CN2

COPY CN1, AX

COPY AX, CN1

COPY U1, CN1

**【情報学基礎】問題番号：IV**

計算困難な問題は、大きな入力に対する計算時間の急激な増加によって多項式時間内の求解が困難な問題である。膨大な数のプロセッサによる並列処理は、計算困難な問題を解く方法として挙げることができるが、汎用的な解法とは成り得ない。その理由を説明せよ。

A problem is called intractable if it is hard to solve for large inputs within polynomial time due to the rapid increase of the execution time. Although parallel computation using an enormous number of processors is one potential solution, it is not a versatile solution. Explain its reason.

---

**【情報学基礎】問題番号：V**

以下の事柄から4つを選び、説明せよ。

- ・ 決定木
- ・ コンパイラ
- ・ タイムシェアリング
- ・ オペレーティングシステム
- ・ NP 完全問題
- ・ エキスパートシステム

Choose four terms from the following list and explain them.

- Decision trees
  - Compilers
  - Time sharing
  - Operating systems
  - NP-complete problems
  - Expert systems
-