

数理解析・計算機数学 I

2004年度前期

学期末レポート問題

2004年7月28日

1 レポート問題について

問題は「カテゴリA」と「カテゴリB」に分類されている。「カテゴリA」はアルゴリズムに関する問題などであり、「カテゴリB」はC言語によるプログラムの作成である。

2 評価の方法について

講義の評価はこの問題集にあるレポートの結果のみを元に行う。全ての問題で、大学院生と学部生とは同一の採点基準とする。

3 レポートの提出方法

それぞれのカテゴリごとに提出方法が異なる。また、大学院生と学部生それぞれに対して推奨する提出方法が異なる。ここに書かれた注意を守っていないレポートに関しては採点の対象としない。

3.1 全てに共通な注意

各問題とも、アルゴリズム等を本で調べるのはかまわないが、プログラムは自分で考えること。

他人のレポートやプログラムのコピー、またはそのマイナーな改変を提出してはならない。そのようなことがあった場合は、提出者だけでなく、コピーをさせた人についても、すべてのレポートを評価対象としない。上記のような疑問点が生じた場合には個別に口頭試問を行うことがある。

3.2 提出期限

提出期限は2004年8月19日(木)とする。提出期限以後に提出されたレポートはいかなる理由があっても受理しない。

電子メールで提出するレポートに関しての締切の意味は、2004年8月19日(木)23時59分59秒(日本標準時)以前の時刻でメールが発信されていればよいこととする¹。電子メール以外で提出するレポートに関しては、多元数理科学研究科事務室前の提出箱に提出するか、以下の「その他の提出に関する注

¹電子メールの発信日時は、その気になれば偽造することが可能である。この辺りは「紳士協定」としたい。要するに提出期限の翌朝に届いていればよい。

意事項」で示す方法で提出すること。この場合は、2004年8月19日(木)17時(必着)を締め切りとする。

3.3 レポートに関する連絡

レポートの採点結果の連絡および、口頭試問を行う場合の連絡については、2004年8月26日(木)までに各個人宛に以下の「その他の提出に関する注意事項」で示す連絡方法で連絡する。

3.4 レポートの提出方法と提出先

レポートの提出方法は以下の規定を厳守すること。この規定に反するレポートは採点対象とはしない。

3.4.1 カテゴリAのレポートの提出方法

【大学院生】 \LaTeX またはワードプロセッサなどの出力をPDF形式に変換したものを電子メールで提出すること。

【学部生】 \LaTeX またはワードプロセッサなどの出力をPDF形式に変換したものを電子メールで提出するか、または、通常のレポートのようにレポート用紙に書いて提出すること。

いずれの場合にも、1つのファイルまたは1枚の用紙に複数の問題の答案を書いてはならない。また、同一の問題の答案を異なる方法で提出した場合には、「レポート用紙」に書かれた答案を採点対象とする。

3.4.2 カテゴリBのレポートの提出方法

電子メールを用いて、プログラムソースコードを提出すること。

3.4.3 電子メールでの提出方法

電子メールでレポートを提出する場合の送付先は

computer-lecture-report@math.nagoya-u.ac.jp

とする。²電子メールの発信者アドレスは、情報メディア教育センターのアドレス(学部生の場合)または、多元数理科学研究科のアドレス(大学院生の場合)であること。ただし、以下の「その他の提出に関する注意事項」を参照のこと。

電子メールの内容は以下の通りとする。

- レポートは「添付ファイル」として電子メールに添付すること。複数の問題の答案を単一のファイルとしてはならない。一つの問題の答案が複数のファイルからなる場合には内藤または久保に相談すること。
- 一通の電子メール中に複数の添付ファイルをいれてもよい。
- 「添付ファイル」のファイル名は以下の規約に従うこと。
 - カテゴリBの答案で、単一のC言語のプログラムソースコードである場合には、`Prob_B_XX_YYYYYYYY.c`というファイル名にすること。

²これはレポート提出専用のアドレスであるので、質問等はこれまで通り computer-lecture@math.nagoya-u.ac.jp に送付すること。また、このアドレスにレポートを送信すると、送信者アドレスが「正当」なものであれば、自動的に受理通知のメールが返信される。

- カテゴリAの答案で、単一のPDFファイルの場合には、Prob_A_XX_YYYYYYYY.pdf というファイル名にすること。
- 大学院生が提出すべき「講義内容要約」については、PDFファイルでSummary_YYYYYYYY.pdf というファイル名にすること。
- カテゴリBの答案で、一つのレポート問題の答案が複数のファイルから構成される場合には、それら全てのファイルを Prob_B_XX_YYYYYYYY というサブディレクトリに格納し、そのサブディレクトリを tar でまとめ、gzip または bzip2 で圧縮し、Prob_B_XX_YYYYYYYY.tar.gz または Prob_B_XX_YYYYYYYY.tar.bz2 というファイル名にすること。

ただし、XXの部分には問題番号が入り、YYYYYYYYの部分には学籍番号が入る。たとえば、学籍番号069800001の学生の問題番号 Prob_A_01 のPDFファイルは Prob_A_01_069800001.pdf となる。

- ファイルの添付方法は、Netscape メール（学部生の場合）または mew（大学院生の場合）のファイル添付方法に従えばよい。
- C言語プログラムソースコードの場合には、ソースコードの先頭にコメントとして、学籍番号、氏名、問題番号を明記すること。
- プログラムの実行形式を電子メールに添付しないこと。
- レポートの答案を Adobe Acrobat PDF 形式で行う場合には、A4用紙で正常な出力が得られること。先頭ページには、学籍番号、氏名、問題番号が出力されること。
- 電子メールでレポートを提出する場合には、同じ問題に対して複数回レポートを出してもよい。その場合、最後に送られたもののみが採点対象となる。

3.4.4 提出箱への提出時の注意事項

電子メールを用いず紙に書いた答案を提出する場合には、多元数理科学研究科事務室前にある、レポート提出箱に提出するか、次の「その他の提出に関する注意事項」にある提出方法を従うこと。1枚の用紙に複数の問題を解答してはならない。この場合、用紙はA4サイズ片面を利用し、一つの問題の答案が複数枚に渡るときには、それらを左上をクリップでとめること。（ステップラでとめてはいけない）また、問題答案の全てのページに学籍番号・氏名・問題番号を明記すること。

3.4.5 その他の提出に関する注意事項

提出締め切りおよび授業担当者からの連絡期日が夏休み中になるため、以下の対応を行う。

【カテゴリAのレポートの提出について】 学部生に関しては、カテゴリAのレポートを郵送で提出してもよい。郵送またはそれに類似する方法で提出する場合には、以下の条件に従うこと。

1. 送付先は
〒464-8602
名古屋市千種区不老町
名古屋大学大学院多元数理科学研究科
内藤 久資 宛

すること。
2. 送付方法は「ゆうパック」または「宅配便」等の受領証明を行えるもの、または「配達証明」を用いた郵便物とすること。
3. 提出する封筒等の表面に「レポート提出」と明記すること。

【電子メールでの提出について】 電子メールでのレポートの提出については、情報メディア教育センターのアドレス（学部生の場合）または、多元数理科学研究科のアドレス（大学院生の場合）であることが望ましいが、次に述べる「連絡先の登録」に電子メールアドレス（携帯電話等のアドレスを除く）を登録した場合にのみ、登録したアドレスからの電子メールも受理する。

【連絡先の登録】 授業担当者からの連絡を行う際の連絡先は、情報メディア教育センターのアドレス（学部生の場合）または、多元数理科学研究科のアドレス（大学院生の場合）とする。

3.4.6 その他の注意事項

- 問題に関する質問がある場合には、「講義の掲示板」を利用するか、授業の電子メールアドレス宛に質問を行うこと。
- 夏休み中に直接質問に来る場合には、電子メールなどでアポイントメントをとること。（夏休み中に質問に来たからと言って、大学にいるかどうかわからないので）
- 提出状況および成績に関する問い合わせを行うことはかまわないが、返答は電子メールによってのみ行い、その返答先は、情報メディア教育センターのアドレス（学部生の場合）、多元数理科学研究科のアドレス（大学院生の場合）、「連絡先」に登録したアドレス以外には行わない。
- 成績に関わる担当者からの連絡は、電子メールによってのみ行い、情報メディア教育センターのアドレス（学部生の場合）、多元数理科学研究科のアドレス（大学院生の場合）、「連絡先」に登録したアドレスにのみ行う。その際には「内藤の電子署名」がついた署名付きのメールで連絡する。

4 レポートの採点方法

採点方法は「中間レポート」の採点方法に準ずる。

すなわち、おおそ A, B, C, D の4ランクで採点し、その得点は A = 10, B = 6, C = 3, D = 0 とする。³ 中間レポートと同様に、上記の中間の点も存在する。また、特に優れた答案に対しては、10点を越える点を与える場合がある。具体的に各カテゴリの答案の評価基準は以下のとおりである。

4.1 カテゴリAに関する採点方法

通常の数学のレポートの採点と同様である。その採点の基準は次の通り。

- A: 答案として全く問題のないもの。
- B: 答案としてやや問題があるが、出題の主旨をよく理解して回答しているもの。
- C: 答案として問題があるもの。単に計算のみが書かれていて、答案として読むことができないものもは最高でもCとしてしか評価しない。
- D: 全くでたためな答案であったり、出題の主旨を全く理解していないもの。また、明らかな間違いを含むものもDとする。

³ 「中間レポート」では A = 10, B = 6, C = 3, D = 0 であった。

4.2 カテゴリ B に関する採点方法

4.2.1 採点の基準

採点の基準は次の通り。

- A: プログラムとして全く問題の無い場合。
- B: プログラムの動作としては正常であるが、プログラムのアルゴリズムや、記述方法に少々の問題のある場合。
- C: プログラムの動作としては正常であるが、プログラムのアルゴリズムや、記述方法に重大な問題のある場合。
- D: プログラムの動作に問題がある場合。

具体的な採点方法として以下のような基準を用いる。

D とする場合

- プログラムがコンパイルできないとき。
- 問題に示された仕様になっていない場合。
- 出題意図を理解していないと思われる場合。
- バグがある場合。

最高でも C とする場合

- プログラムに対するコメントが全くない場合。
- 明らかに C 言語のプログラミングがわかっていないと思われる場合。
- 適切な関数の利用法が行われていない場合。
- 制御方法に大きな問題があると判断される場合。
- アルゴリズムに大きな問題があると判断される場合。
- プロトタイプ宣言がない場合。
- 文法上問題のある場合。

最高でも B とする場合

- コメントが適切でない場合。
- 制御方法が適当でないと判断される場合。
- アルゴリズムが適当でないと判断される場合。
- プログラムの記述方法が適当でないと判断される場合。
- コンパイル時に「警告」が出る場合。(ただし、利用しない変数が存在することに関する警告はあってもかまわない。)

4.3 単位認定の基準

単位認定は、大学院生と学部生は異なった基準で行う。以下の条件は、単位を認定するための必要条件であり、最終的に提出された結果により、基準が下がる可能性もある。

4.3.1 単位認定の基準

以下の必要条件を満たすこと。

1. 問題文中に「必修問題」と明記したのすべての問題に解答とすること。
2. 問題文中に「選択問題」と明記した問題の中から1問を選択し解答すること⁴。
3. カテゴリ A の平均的な評価が B 以上であること。すなわち、合計点で 30 点以上であること。
4. カテゴリ B の平均的な評価が B 以上であること。すなわち、合計点で 54 点以上であること。
5. 大学院生の場合には、以上に加えて、教務委員会からの指示にある、「講義の要約」を \LaTeX で書いてその PDF ファイルを電子メールで送付すること。その締め切りも、これらのレポートの締め切りと同一とする。

4.3.2 単位認定と成績の基準について

1 回目の講義で提示した単位認定の基準は以下の通りであった。

合格最低基準(可) 講義で解説したアルゴリズムを用いて、C 言語による正しいプログラムを書けること。

特に、標準的なプログラムを正しく書けること。(要するに、C 言語のプログラミングと初歩的なアルゴリズムが最低限のレベルで理解できていること。)

良 講義で解説したアルゴリズムを正しく理解し、C 言語による正しいプログラムを書けること。(要するに、初歩的なアルゴリズムを正しく理解し、それを C 言語で正しくプログラミングできること。)

優 講義で解説したアルゴリズムを応用して、アルゴリズムを自分で考え、それを C 言語で正確にプログラムできること。特に、C 言語の仕様を正しく理解できていること。

この基準に沿って、プログラミングとアルゴリズムの理解が十分でないと判断される場合には単位を認定しない。したがって、カテゴリ A 及びカテゴリ B の両方の平均的な評価が B 以上であることが合格最低基準となる。また、成績(優・良・可)の基本的な基準は

可 カテゴリ A・カテゴリ B の平均的な評価が共に B である場合。

良 カテゴリ A・カテゴリ B の平均的な評価がいずれか一方が A- で他方が B である場合。

優 カテゴリ A・カテゴリ B の平均的な評価が共に A- である場合。

となる。⁵

5 カテゴリ B のプログラムに関する要求事項および注意事項

カテゴリ B 問題に対する答えは以下の仕様の下でプログラムを作成すること。

- 「Section 5.1: プログラムに関する仕様」に指定された仕様を厳守すること。
- 「Section 5.2: プログラムに対する要求事項」の内容に沿ったプログラムであること。

これらの仕様を満たさない答えは採点対象としない可能性がある。

5.1 プログラムに関する仕様

- すべての問題は ANSI の規格内の C 言語で記述すること⁶。文献 [1, 付録 B] に記載されている標準関数は、数学関数 (math.h に定義されている関数) を除いて自由に利用してよい。

⁴ 選択問題の複数問に解答してはならない。選択問題に関して、複数問の答えが提出されている場合には、そのうち最も低い評価の答えを評価対象とする。

⁵ 平均的な評価が B とは、平均点が 6 点、平均的な評価が A- とは、平均点が 8 点程度であることを指す。

⁶ すなわち、文献 [1] に記述されている規格の範囲内であること。

- 特に指定のない限り, 入力 は標準入力から行う。
- 特に指定のない限り, 出力 は標準出力に出力する。
- 問題文中および入出力に用いられる数値の表現は, 特に指定のない限り 10 進数とする。
- 特に指定のない限り, 問題文中で指定された数 (型) の意味は以下の通りとする。

「整数」 入力においては, `int` で表現できる数であって, 符号 (+ または -) がつくことがある。出力においては, 負の数の時にのみ符号をつけ, 非負の時には符号をつけない。

「正の整数」または「非負整数」 入力においては, `unsigned int` で表現できる数であって, `unsigned int` 型の値として正 (または非負) の値をとるものである。正の数の場合には, 符号 (+) がつくことがある。出力においては, 符号をつけない。

「平衡 3 進数」 $-1, 0, 1$ に対応する数字として, それぞれ `n, z, p` を用いる。それぞれの文字の間に空白をいれてはいけない。

「一変数多項式」 上に示した「整数」のならば一変数多項式をあらわす。この場合, 「整数」のならばを前から順に降べきの順の係数とみなす。係数は指定された係数環の表示方法に従うこと。

例えば, 係数環が \mathbb{F}_3 の場合に “`0_2_0_0_1`” は $2x + x^4$ を表す。

出力は降べきの順に係数を上に示した「整数」のならばとして出力すること。

その他, 特に指定のない限り次の仕様に沿うこと。

文字コード体系

- コード体系は ASCII コード体系と仮定する。
- プログラムコード内の「コメント」中にあらわれる日本語文字のコード体系は「EUC-JP 日本語文字コード」とする。

オブジェクトのバイト幅など

- `char` 型変数が占めるメモリ内のビット数は 8 ビットであると仮定する。
すなわち `limits.h` 内で定義されるマクロ “`CHAR_BIT`” の値は 8 であり, 「1 バイト」とは「8 ビット」であると仮定してよい。
- 各整数型変数が占めるメモリ内のビット数は $8 \times \text{sizeof}(\text{型})$ であると仮定してよい。
各整数型の `sizeof` 演算子の結果の値は以下の通りであるとする。

【大学院生】

```
sizeof(short) = sizeof(unsigned short) = 2,
sizeof(int)   = sizeof(unsigned int)   = 4,
sizeof(long)  = sizeof(unsigned long)  = 8,
```

【学部生】

```
sizeof(short) = sizeof(unsigned short) = 2,
sizeof(int)   = sizeof(unsigned int)   = 4,
sizeof(long)  = sizeof(unsigned long)  = 4,
```

- `char, short, int, long` の各型において, 負の数は「2 の補数」によって表現されていると仮定してかまわない。

文字の意味

- 「文字」とは, 非印字文字を含む 1 バイトのデータのことである。
- 「英字」とは, 標準関数 `isalpha` が非 0 の値を返す文字のことである。
- 「大文字」とは, 標準関数 `isupper` が非 0 の値を返す文字のことである。
- 「小文字」とは, 標準関数 `islower` が非 0 の値を返す文字のことである。
- 「数字」とは, 標準関数 `isdigit` が非 0 の値を返す文字のことである。

- 「英数字」とは, 標準関数 `isalnum` が非 0 の値を返す文字のことである。
- 「印字可能文字」とは, 標準関数 `isprint` が非 0 の値を返す文字のことである。
- 「空白文字集合」とは, 標準関数 `isspace` が非 0 の値を返す文字のことであり, 以下の「タブ文字」, 「改行文字」, 「垂直タブ文字」, 「改ページ文字」, 「復帰文字」, および「空白文字」のいずれかのことである。
- 「タブ文字」(ASCII コード `0x09`) とは, C 言語においては「`\t`」で表現される文字である。
- 「改行文字」(ASCII コード `0x0a`) とは, C 言語においては「`\n`」で表現される文字である。
- 「垂直タブ文字」(ASCII コード `0x0b`) とは, C 言語においては「`\v`」で表現される文字である。
- 「改ページ文字」(ASCII コード `0x0c`) とは, C 言語においては「`\f`」で表現される文字である。
- 「復帰文字」(ASCII コード `0x0d`) とは, C 言語においては「`\r`」で表現される文字である。
- 「空白文字」とは, ASCII コード `0x20` で表される文字である。

行の意味

- 「行」とは, 改行文字, 復帰文字, ファイル終端を区切りとした文字の並びのことである。
- 「テキスト」とは, 印字可能文字, 空白からなるデータのことである。
- テキストファイルの改行コードは ASCII コード `0x0a` であると仮定する。
- 「空白文字集合」の中には「改行文字」が含まれているが, 一方で, 「`\r`行」とは, 改行文字, 復帰文字とファイル終端を区切りとした文字の並びのことである」としてあるので, 「行」の中に改行文字, 復帰文字は, 行の終端文字としてしかあらわれない。

データの入力

- 特に指定しない限り, 標準入力からデータの入力を行う。
- 入力行の 1 行の最大文字数は行末の改行コードを含め 1023 文字以内と仮定してかまわない。
- 「 N 個の整数 (正の整数, 非負整数) を入力する」という表現は, 入力行の 1 行に「空白文字集合」を区切り文字集合とするトークンとして N 個の整数 (正の整数, 非負整数) が入力されることを意味する。したがって, 入力行の先頭に空白文字集合に含まれる文字があらわれる場合がある。
- 1 行に含まれる「整数をあらわす文字列」などの最大トークン数は 10 であると仮定してかまわない。したがって, 一変数多項式の最大次数は 9 となる。
- データ入力において 1 組のデータは全て 1 行に記述されているものとする。
- データ入力はただ一度だけ行われるものとする。すなわち, 複数回のデータ入力は行わないと仮定してよい。
- データの読み取りのために `scanf` 関数または, それに類似の関数を用いるとエラー処理が困難となるため, これらの関数を利用してデータを読み取ることは禁止する。
- プログラムの検証時には, 問題の仕様に沿わないデータの入力は行わない。

データの出力

- 特に指定しない限り, 標準出力へのデータの出力を行う。
- 出力行の先頭に空白文字集合に含まれる文字を出力してはならない。
- 出力行の最後には一つだけの改行文字を入れること。
- 出力行の最後に不要な空白文字が入ることはかまわないとする。
- 出力する数値等は, その問題に適した形の数のフォーマット (上記を参照) で出力すること。

- 複数の数値を出力する時には、それぞれの数値は一つの空白文字で区切られ、すべてを一行に出力すること。
- 問題の仕様に沿わない、不必要な出力を行ってはならない。

その他

利用可能メモリ量に関しては、情報メディア教育センター及び多元数科学研究科のワークステーションの実装に準ずる。もし、リソースの不足が生じた場合には内藤または久保に相談すること。

5.2 プログラムに対する要求事項

カテゴリBのプログラムに関しては、以下の内容を要求する。

プログラムの記述に関して

- プログラムを記述する場合には適切に関数に分割すること。関数に分割した場合は、関数の仕様をコメントとして書き、プロトタイプ宣言をすること。関数の仕様とは、それぞれの仮引数の意味と戻り値の意味を示し、関数の振る舞いの意味を明確にすることを言う。
- アルゴリズムについては、そのアルゴリズムに対する説明をコメントとして書くこと。ただし、講義で解説したアルゴリズムを利用する場合には、「どのようなアルゴリズムを利用したか」のみを簡単に書けばよい。
- 講義で解説していないアルゴリズムを利用する場合には、アルゴリズムについて、詳細な説明をコメントとして付け加えるか、該当する「カテゴリA」の問題番号を明示すること。特に、正しい入力データが与えられる限り、プログラムが必ず終了する根拠を明確に述べる必要がある。

プログラムの実行とその結果に関して

- 特に指定していない限り、プログラムは「ほぼ一瞬」で終了すること。（プログラムの実行時間の上限は指定しないが、多元数科学研究科のシステムにおいて、標準的な負荷状態で1秒を越えるような問題は出題していない。）
- 特に指定していない限り、結果が指定の型によって表現可能である場合には、正しい結果を出力できること。
- プログラム内部で計算対象のオブジェクトを「より広い型」に変換して計算してはならない。例えば、「整数」(int 型)を計算するために、プログラム内部で unsigned int 型、long 型など、より大きな数値を表現可能な型のオブジェクトに変換して計算してはならない。

プログラムのコンパイル条件

- 提出されたカテゴリBの答案（プログラム）は、以下の環境において処理を行い、その結果を検証する。
ハードウェア Sun Microsystems 社, Sun Fire 280R (CPU: Ultra SPARC III, 1.0GHz),
または Sun Fire V210 (CPU: Ultra SPARC IIe, 1.0GHz).
オペレーティングシステム Solaris 9.
C コンパイラ gcc version 3.2.1.
コンパイラオプション -Wall -ansi -pendantic
この環境は、大学院生については、利用するハードウェアは異なるが、実習時のソフトウェア環境と同一のものである。
- 学部生に関しては、実習環境とは異なるため、もし、処理結果に疑問が生じた場合には、実習環境と同一の環境で結果を検証する場合がある。

6 レポート問題

6.1 カテゴリA

● Prob-A-01 【必修問題】

正の整数 x に対して、 x の「1の補数」及び「2の補数」とは何かを述べ、 x の1の補数を \bar{x} 、2の補数を \hat{x} とするとき、 $\hat{x} = \bar{x} + 1$ が成り立つことを示しなさい。この事実を用いて、多くの計算機で負の整数の表現として補数が使われている理由を論じなさい。

● Prob-A-02 【必修問題】

自然数 b に対して、 $B = 2b + 1$ とします。整数 n の平衡 B 進表示を求めるためのアルゴリズムを示しなさい。

● Prob-A-03 【必修問題】

n までの素数を求めるためのエラトステネスのふるいのアルゴリズムを示しなさい。（この問題では計算量を示す必要はない）

● Prob-A-04 【必修問題】

フィボナッチ数列

$$F_{n+2} = F_{n+1} + F_n, \quad n \geq 1, \quad F_1 = 1, \quad F_0 = 0$$

の第 n 項 F_n を次の再帰的関数呼出しを用いて求めようとするときの関数呼出しの回数を求めなさい。

```
unsigned int fibonacchi(unsigned int n)
{
    if (n == 0U) return 0U;
    if (n == 1U) return 1U;
    return fibonacchi(n-1) + fibonacchi(n-2);
}
```

また、一般に再帰的関数呼び出しを用いることの利点・欠点について論じなさい。

● Prob-A-05 【必修問題】

$0 < x < 1$ をみたす有理数 x を以下の形の連分数に展開する。

$$x = \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}}$$

これを x の正則連分数展開と呼びます。 $0 < x < 1$ をみたす有理数の正則連分数展開を求めるアルゴリズムを示しなさい。

6.2 カテゴリ B

● Prob-B-01 【必修問題】

2つの正の整数を入力する。それらの最小公倍数を出力するプログラムを書きなさい。

● Prob-B-02 【必修問題】

100000 までのすべての素数を小さいものから順に出力するプログラムを書きなさい。

● Prob-B-03 【必修問題】

1つの整数を入力する。その数の平衡3進表示を出力するプログラムを書きなさい。

● Prob-B-04 【必修問題】

2つの正の整数を入力する。それらを順に n, m としたとき、二項係数 $\binom{n}{m}$ (これは ${}_n C_m$ と書くことがある) を出力するプログラムを書きなさい。ただし、 $1 \leq n \leq 34, 0 \leq m \leq n$ と仮定する。

● Prob-B-05 【必修問題】

2つの正の整数を入力する。それらの全ての公約数を出力するプログラムを書きなさい。結果は値の重複なく小さい順に出力すること。

● Prob-B-06 【必修問題】

2つの正の整数を入力する。それらを順に n, m (ただし、 $m \leq n$ を仮定する) としたとき、 m/n の正則連分数展開の係数 $\{a_n\}$ を空白で区切って出力するプログラムを書きなさい。ただし、この問題は n/m を既約分数で表したときの分母は 1024 以下であると仮定します。

● Prob-B-07 【必修問題】

2つの正の整数を入力する。それらを順に n, m (ただし、 $m \leq n$ を仮定する) としたとき、 m/n の10進小数表示を出力するプログラムを書きなさい。答が有限小数で表示可能な時には有限小数として表示すること。循環小数として表現される時には、循環節を $\{ \}$ で囲みます。ただし、この問題は n/m を既約分数で表したときの分母は 1024 以下であると仮定します。

● Prob-B-08 【必修問題】

2行にわたって、1行ごとに5つの非負整数を入力する。その1行目の数値を順に w_0, d_0, h_0, m_0, s_0 、2行目の数値を順に w_1, d_1, h_1, m_1, s_1 としたとき、 w_0 週 d_0 日 h_0 時間 m_0 分 s_0 秒から w_1 週 d_1 日 h_1 時間 m_1 分 s_1 秒までが w_2 週 d_2 日 h_2 時間 m_2 分 s_2 秒であるかを計算するプログラムを書きなさい。出力は w_2, d_2, h_2, m_2, s_2 を順に一つの空白で区切って標準出力に出力しなさい。ただし、 w_2, d_2, h_2, m_2, s_2 は、それらが0であってもそれを出力することとします。

入力および出力は $0 \leq w_i < 100, 0 \leq d_i < 7, 0 \leq h_i < 24, 0 \leq m_i < 60, 0 \leq s_i < 60$ をみたとします。また、 w_0 週 d_0 日 h_0 時間 m_0 分 s_0 秒は w_1 週 d_1 日 h_1 時間 m_1 分 s_1 秒よりも「前の時間」であることは仮定してかまいません。

● Prob-B-09 【選択問題】

コマンドライン引数の第1引数に素数を指定し、標準入力から2行にわたる整数の列を入力する。コマンドライン引数に指定した素数を p としたとき、標準入力から入力された整数の列のそれぞれを以下のように \mathbb{F}_p を係数体とする一変数多項式とみなす。この時、その2つの多項式の最大公約多項式の係数を降べきの順に空白で区切って出力するプログラムを書きなさい。ただし、コマンドライン引数には素数以外の数は与えないこと、入力される係数は0以上 p 未満であることは仮定してよい。

● Prob-B-10 【選択問題】

コマンドライン引数に1つの正の整数を与える。それを k としたとき、4本の塔と k 枚の円盤を持つハノイの塔の円盤の移動の様子を記述するプログラムを書きなさい。

一般に「 k 枚の円盤と p 本の塔を持つハノイの塔」とは以下のような問題である。

- p 本の柱には $0, \dots, p-1$ の番号が与えられている。
- 大きさの異なる k 枚の円盤 (小さい順に $0, \dots, k-1$ の番号が与えられている) が、番号0を持つ柱に下から大きな順に並んでいる。
- これを1回に1枚づつ移動し、番号 $p-1$ を持つ柱に下から大きな順に並ぶように移動する。
- 小さな円盤の上には大きな円盤を置くことはできない。

ここで、移動の様子は、1行に1回の移動を書き、先頭には移動する円盤の番号、元の柱番号、移動先の柱番号を、1つの空白を区切りとして出力する。なお、この問題は $k \geq 3$ に対して、同じ円盤の枚数を持つ「3本のハノイの塔」の手順よりも真に少ない手順の解を出力すること。

● Prob-B-11 【選択問題】

以下の構文規則に従う「算術式」(Expression) を、「逆ポーランド記法」に変換するプログラムを書きなさい。

入力は1行ごとに1つの算術式が記述されたテキストファイルを標準入力から読み込む。出力は1つの算術式に対する変換結果を1行に記述したテキストファイルを標準出力に出力する。入力される算術式は、各構成要素が1つ以上の空白で区切られている。出力する逆ポーランド記法の式は、各構成要素が1つ以上の空白で区切られているものとする。

なお、入力された算術式が構文規則に一致しないときには“Error”と出力すること。

【算術式の構文定義】 算術式の構文定義は以下の規則に従う。

```

<Expression> ::= <Term> | <Expression> <Additive Operator> <Term>
<Term> ::= <Factor> | <Term> <Multiplicative Operator> <Factor>
<Factor> ::= <Primary Expression>
                | <Primary Expression> <Power Operator> <Factor>
<Primary Expression> ::= <Identifier> | ( <Expression> )
<Identifier> ::= A | ... | Z | a | ... | z
<Additive Operator> ::= + | -
<Multiplicative Operator> ::= * | / | %
<Power Operator> ::= ^

```

【注意】

1. 出力の末尾に「空白文字」がつくことはかまいません。
2. この問題は複数行に渡る入力を与えられることがあります。

6.2.1 各問題の入出力例

ここでは、カテゴリBの各問題について、入出力の例または実行例を挙げておきます。

Prob-B-01

入力例
出力例

Prob-B-02 20までの素数を出力した例.

Prob-B-03

入力例
出力例

Prob-B-04

入力例
出力例

Prob-B-05

入力例
出力例

Prob-B-06

入力例
出力例

Prob-B-07

入力例
出力例

Prob-B-08

入力例

出力例

Prob-B-09 ./a.out 3として $p = 3$ を与えたとき,

入力例
出力例

Prob-B-10 ./a.out 3として $k = 3$ を与えたときの解は以下のようになる.

1. 柱0にある円盤0を柱1に移動する.
2. 柱0にある円盤1を柱2に移動する.
3. 柱0にある円盤2を柱3に移動する.
4. 柱2にある円盤1を柱3に移動する.
5. 柱1にある円盤0を柱3に移動する.

それに対しては、以下のように出力する.

```
0 0 1
1 0 2
2 0 3
1 2 3
0 1 3
```

Prob-B-11 以下のように複数行にわたる入力を与えられることがある.

```
( a + b ) * c ^ d / ( e - f ) ^ g
a + b * c ^ d / e - f ^ g
( a + b - c ^ d / e ^ f ^ g
a + b - c ^ d / e ^ f ^ g
```

それに対しては、以下のように出力する.

```
a b + c d ^ * e f - g ^ /
a b c d ^ * e / + f g ^ -
Error
a b + c d ^ e f g ^ ^ / -
```