チェックディジットを計算しよう

情報処理学会・学会誌「情報処理」
2024年4月1日 09:11

久野　靖（電気通信大学）

まだまだ続く「教科『情報』の入学試験問題って？」，2025年の試験開始がいよいよ近づいて，大学入試センター以外に，「情報」の独自入試を実施する大学からの模擬試験問題なども入手できるようになってきました[1) 2)]．

その中から今回は，京都産業大学が協力高校において実施した模擬試験の問題のうちの1問，チェックディジットの問題を取り上げます．独自入試の問題なんて難しそう，と思いましたか？

確かに，大学入試センターの試験にないようなタイプの問題もありますが，この問題は基本的な知識を問う第1問に続く第2問（京都産業大学のフォーマットでは「[II]」と記しています）として，基本的な考える力を見るものとなっています．

この問題を含む模擬試験の全問題や解答例・解説は京都産業大学の入試情報サイト [3] から「2025年度入学者選抜『情報』模擬問題（サンプル問題）」として入手できますので，必要な方は参照してください．では，始めましょう．

目次

チェックディジットって？

皆様は「チェックディジット」という言葉を知っていますか？　知っている人も知らない人もいると思います．「情報I」の範囲でこれに近いのは「パリティビット」だと思いますが，そちらは知っていますね．

パリティビットとは，ビット列に対して1ビット余分のビット（パリティビット）を付け加え，合わせたビットのうち「1」の数がいつも偶数（またはいつも奇数）になるようにするものです．そうしておくと，どこか1ビットが伝送エラーなどで変わってしまったとき，「1」の数が偶数（または奇数）でなくなっているので，エラーを発見できる，という仕掛けです．

そして，チェックディジットとは，普通の（というか十進表記の）数字列に対して1桁以上余分の数字を付け加え，合わせた数字列の各数字をもちいて「ある計算をすると，ある条件が成り立つ」ようにします（そうなるように付け加える数字を選ぶ）．すると，合わせた数字列のどこか1桁以上が伝送エラーなどで変わってしまったとき，「ある条件が成り立たなく」なっていることで，エラーを発見できることがあります（そうなるように「ある計算」を決めておく）．

たとえば，本稿で取り上げる問題では，3桁の数字XYZに，次の規則で計算した1桁のチェックディジットCを付け加えたXYZCが「合わせた数字列」です．

```
C = (X×5 + Y×2 + Z) mod 7    ---- (K)
```

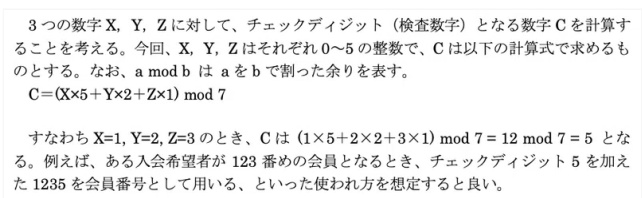
この場合，(K)が「ある計算」で，4桁目の数字がCと等しいことが「ある条件」です．そして，XYZCのどれかが変わってしまうと，この条件が成り立たなくなって，エラーが発見できます（本稿の問題の場合は，最後から2番目の節で説明しますが，どれか1つが変わった場合，必ず発見できます）．

上で2カ所，「1桁以上」と書いたのにお気づきですか．パリティチェックでは2つのビットが反転してしまうとエラーが発見され損なうのですが，十進だと付け加える数字に10通りの選択肢があるので，2つ以上の桁でもエラーが発見できる場合が出てきます．加える桁数を増やせば，その能力も増やせます．このエラー検出力と，（手計算する可能性も考える場合）計算のしやすさなどを考えて，チェックディジットの計算方式は決められます．

チェックディジットは私たちの身の回りでもつかわれています．書籍のISBN [4] や加工食品などのバーコード [5] が代表的です．これらでは，手で書き写すときの転記間違いや，バーコードリーダの読み間違いに対処する上で，チェックディジットによるエラー検出が有用なのです．

本設問におけるチェックディジット

設問の方では，問題の冒頭7行で，この設問で扱うチェックディジットについて説明されています（**図1**）．



3つの数字X，Y，Zに対して、チェックディジット（検査数字）となる数字Cを計算することを考える。今回、X，Y，Zはそれぞれ0〜5の整数で、Cは以下の計算式で求めるものとする。なお、a mod b は aをbで割った余りを表す。

C＝(X×5＋Y×2＋Z×1) mod 7

すなわちX=1, Y=2, Z=3のとき、Cは (1×5＋2×2＋3×1) mod 7 = 12 mod 7 = 5 となる。例えば、ある入会希望者が123番めの会員となるとき、チェックディジット5を加えた1235を会員番号として用いる、といった使われ方を想定すると良い。

図1　チェックディジットの説明

チェックディジットとは何かという，上で説明した知識はなくても読めますが，あった方が分かりやすいですね．まあ，試験問題ですから，そういうものでしょう．計算規則が一番大切なので，しつこく再掲しておきます．

```
C = (X×5 + Y×2 + Z) mod 7    ---- (K)
```

もとの数字列がXYZの3桁，チェックディジットをつけたものはXYZCの4桁ということで，簡単ですね．

設問（A）：チェックディジットの役割・性質

設問（A）は，チェックディジットの役割・性質について書かれた文章の正誤問題です（**図2**）．

設問（A）

C のチェックディジットの役割や性質について以下に列挙した。これらのうち、正しいものには ○ を、正しくないものには × を、解答用紙の 1 ～ 5 の枠内にそれぞれ記載せよ。

(1) 上の会員番号のようなものを設計する場合、C は最後の数字として置く必要がある。
(2) C があることにより、X, Y, Z の数字の入力間違いを検出できる場合がある。
(3) C があることにより、X, Y, Z の数字の入力間違いを誤り訂正できる。
(4) C の値が同じになる X, Y, Z の値の組合せは複数存在する。
(5) X, Y, Z の値を単純に加算したチェックサムを使う方式と違い、桁の入れ替わりを検出できる場合がある。

図2　設問（A）

では1つずつ検討してみましょう．

（1）はどうでしょう．Cは最後に置くことが多くはありますが，X，Y，Z，Cが正しく取れさえすれば計算に不都合はないので，順番を違えても（その順番にすべて統一されているかぎり）問題ありません．というわけで×です．

もっと具体的に挙げた方がいいでしょうか．たとえば，計算式を同じにしたまま，「CXYZ」のようにチェックディジットを先頭に置くようにします．X = 1，Y = 2，Z = 3のとき，C = 5ですから変えた方式だと会員番号は「5123」ですが，先頭がCだと分かっているので，これでチェックに不都合はありませんね．

（2）はどうでしょう．X，Y，Zのどれかが違う数字になってしまったとします．その結果，式(K)で計算される値が元と変化する「場合がある」というのはいいでしょうか（この点について，あとでま

た触れます）．もし変化していれば，現在書かれているCの値とは違うので，間違いがあったと分かります．ということで，入力間違いがあれば検出できる「場合があり」ます．というわけで○です．

これも具体例を挙げましょう．例に挙がっている「1235」で，最初の数字「1」が「2」に間違ってしまい，「2235」になったとします．すると，チェックディジットの計算は「(2×5+2×2+3) mod 7 = 3」ですから，ついているチェックディジットの「5」と異なり，入力間違いがあったと分かります．

（3）はどうでしょう．誤り訂正できる，ということは，さまざまな間違いがあったとき，元の正しいX，Y，Zに必ず戻せるという意味になります．1桁の数字Cは，10通りの場合しか表わせませんから，「X，Y，Zのどれとどれが間違っているか」ということと「それらが正しくはいくつか」ということを表わすには場合の数が足りません．場合の数が足りれば必ずできるとはまったく言えないですが，今の場合はできない理由があることを確認したかったので．というわけで×です．

具体例として，「1235」で，今度はZの値「3」が「1」に，つまり「1215」になったとします．チェックディジットの計算は「(1×5+2×2+1) mod 7 = 3」です．これとさっきの「2235」を例に使います．つまり，「1215」でも「2235」でも，「3」と計算されるチェックディジットが「5」になっている，という状況はまったく同じで，この状況からどの桁を直すかといったことは分かりようがありません．ですから，少なくともこの場合，誤り訂正できません．また，「1235」が「1245」に間違ったような例だと，(1×5＋2×2＋4) mod 7 = 13 mod 7 = 6 とチェックディジットの計算は5から6へ1だけ増えているので，「XやYが増えるとチェックディジットは少なくとも5あるいは2増えるはずなので，これはZが1増えたのだと特定できる」と思う人もいるかもしれません．しかしこれも，元は「1545」でYが3減った（チェックディジットは6減った，つまりmod 7で考えれば1増えた），元は「1246」で間違ったのはチェックディジットC，などの可能性があります．

（4）はどうでしょう．これは，（2）のところで言った「場合がある」と関係しています．つまり，X，Y，Zが変化したのに，式(K)で計算される値が元と変化しない場合があるか，というのがこの問いで問われていることです．これも場合の数を考えれば分かります．Cの値は10通り，それに対して，X，Y，Zの組合せは，それぞれ0〜5のどれかですから，6×6×6 = 126通りです．したがって，同じCになるX，Y，Zの組合せは複数あると言えます．というわけで○です．

具体例は……上の2つの具体例がそのまま利用でき，X，Y，Zが「223」でも「121」でもチェックディジットは「3」で同じです．それだけでは簡単なので，どうやって「223」とチェックディジットが同じになる数を見つけたか，種明かしをしましょう．まず，Xを2から1に減らしたとします．Xは5倍しますから，(K)の式のmod 7を取る前の値でいうと5減りますね．この値が変化しないためには，「Yの項で4増やし、Zの項で1増やす」「Zの項で5増やす」などをすればいいわけです．前者なら（Yは2倍するので2増やして）「144」，後者なら「128」……と言いたいところですが，X，Y，Zは0〜5ですから，8にmod 7を適用して1で，「121」となります．mod 7は足し算している途中では何回適用してもしなくても最終結果は同じになりますから，Zの数値が大きくなってしまったときは使ってよいのです．

（5）はどうでしょう．単純に加算したチェックサムというのはどういう意味でしょう．単純な加算つまり「C = X+Y+Z」で計算するわけですから，その順番が入れ替わっても同じ値になり，誤りを検

出できません（実際はCが2桁にならないように，mod 7などの演算を加えるでしょうけれど，議論の要点は同じです）．一方，(K)の式では，Xは5倍，Yは2倍，Zはそのままで加えますから，X，Y，Zの入れ替わりがあるとCが違う値となり，検出できる場合がある，ということです．つまり〇です．

具体例ですが，「123」に対するチェックディジットは「5」でしたが，XとZを入れ換えた「321」ではどうでしょう．「(3×5+2×2+1) mod 7 = 6」ですからチェックディジットが異なり，確かに検出できる場合があります．

設問（B）：普通に計算する

設問（B）は簡単で，まあ [X, Y, Z, C] という記法が分かって，チェックディジットの計算方法が分かっていることを見ましょう，という基本問題ですね（**図3**）．

以下、この X, Y, Z, C で構成される数字列［X，Y，Z，C］を考える。

設問（B）
数字列［2，5，3，C］のとき，C に当てはまる数は何か。

図3　設問（B）

答えはもちろん，「253」に対するチェックディジットですから，「(2×5+5×2+3) mod 7 = 2」となり，「2」となります．

設問（C）：指定されたチェックディジットになるZを求める

今度はもう少し難しくて，チェックディジットが与えられていて，その値になるZを求めましょう，という問題です（**図4**）．

設問（C）
数字列［3，4，Z，5］がこの条件を満たすとき，Z に当てはまる数は何か。

図4　設問（C）

まず，Zが0の場合を計算します．「(3×5+4×2+0) mod 7 = 2」ですね．今はチェックディジットが「5」となるようにしたいので，あと3増えればよく，Zもあと3増やして「3」が答えです．

一応、検算もしておきます．「(3×5+4×2+3) mod 7 = 5」ですから，ちゃんと5になっています．

もう1つ，別の解法をお見せしましょう．こっちはやや数学ぽいです．以下，いちいち「mod 7を取る」とか書かないで済ますために，「=」の代わりに，「mod 7を取ると右辺と左辺が等しい」という意味の「[mod 7 =]」という記号を使います．たとえば，7+1 [mod 7 =] 1 と書けます．これを使うと，この問題に対応する，Zを含む方程式は

3×5+4×2+Z [mod 7 =] 5

となります．これを変形していきます．

15+8+Z [mod 7 =] 5　（掛け算を計算した）
23+Z [mod 7 =] 5　（足し算を計算した）
28+Z [mod 7 =] 10　（初項を7の倍数にするため両辺に5を加えた）
Z [mod 7 =] 3　（両辺をmod 7した）

というわけで，Zは3です．この解法が好みという方もいるでしょうか．

設問（D）：指定されたチェックディジットになるYとZの組を求める

最後の問題は，チェックディジットが与えられていて，Zだけでなく Y も任意に決めるとしたら，何通りの組合せがあるか，という問題です（**図5**）．

設問（D）
数字列［1，Y，Z，5］がこの条件を満たすとき，Y，Z の組合せは何通りあるか。

図5　設問（D）

設問は何通りかですが，答えに確信を持つため，組合せを全部列挙することにします．設問（C）のやり方を援用して，Yが0のところから順にチェックディジットが5になる場合を調べます．

「100」ではCが5 → そのまま「100」
「110」ではCが0 → Zが5で「115」
「120」ではCが2 → Zが3で「123」
「130」ではCが4 → Zが1で「131」
「140」ではCが6 → Zは6にはできない!!
「150」ではCが1 → Zが4で「154」

なるほど．途中まで全部のYに対してZが決められるのではと思っていたけれど，各数字が0〜5という制約のため，できない場合があるのですね．というわけで，答えは「5」通りとなります．

こちらも[mod 7 =]を使う別解をお見せします．今度はYとZが変数になります．

5+2Y+Z [mod 7 =] 5
7+2Y+Z [mod 7 =] 7　（初項を7の倍数にするため両辺に2を加えた）
2Y+Z [mod 7 =] 0　　（両辺をmod 7した）

ここからは(Y,Z)の組を列挙してくと，(0,0) (1,5) (2,3) (3,1) (5,4)の5つが答えとなります．これもY = 4のときはZ = 6となり，0〜5という条件を満たせません（同じ問題なので当り前）．

少し数学ぽいけど補足です

これで一応設問が解けるだけの解説はしましたが，なんでこうなっているの，と疑問に思う人のためにもう少し補足させてください．

まず，なぜ「5X+2Y+Z」で，なぜ「mod 7」なのか，ということです．この問題ではチェックディジットを1桁にしたので，足し算で2桁以上になったときに1桁に納める計算が必要です．それなら「mod 10」でいいのでは，と思うかもしれません．そこで試しに，YとZが0で，Xのみ0〜9（実際に使う範囲では0〜5ですが）としたときの5Xの「mod 7」と「mod 10」の計算をしてみます．

X	0	1	2	3	4	5	6	7	8	9
5X	0	5	10	15	20	25	30	35	40	45
mod 7	0	5	3	1	6	4	2	0	5	3
mod 10	0	5	0	5	0	5	0	5	0	5

つまり，YとZが0の場合，mod 7ではXの何が何に変わったとしても（0〜5の範囲なら……実際は0〜6の範囲なら）発見できるけれど，mod 10ではそういかない，と分かります．これは，7と5が互いに素（2数の最大公約数が1）であることによります．そして，2Y+Zがどの値でも同じことが言えます．

Yについても，またZについても同じことが言えます（2と7，1と7の最大公約数は1）．ということは，このチェックディジット方式ではX，Y，Zどれか1つが間違ったときは必ず発見できるのです．

あとそういうわけで，X，Y，Zの範囲が0〜5（実際は6でもよい）にしてあったのも，1つの間違いは必ず発見できるようにしたかったからだと思われます．問題ではそのことを使っていないのですが．

チェックディジットはどうでしたか？

さて，問題の解説は以上でしたが，いかがでしたか？　最後にまとめとして，この問題全体を振り返ってみます．

まず，チェックディジットという題材についてですが，「情報I」で必ず学ぶパリティチェックの自然な拡張となっています．そして十進ベースなので計算も楽で，身近な実用例も複数あり，結果として「情報I」の学習を深めてくれるものになっています．

設問（A）の正誤問題については，1つの設問の中の5つの問題ということで，このチェックディジットの方式をさまざまな視点から見ることになっています．試験の場ではそんな余裕はないと言われるかもしれませんが，いくつかの数にチェックディジットをつけてみて，この方式ではどのような数ができて，どのような特徴があるのか，すこし観察してから取り掛かれると良い結果になるように思います．

設問（A）の解き方は，本稿にあるように，一般論をもとに考えることも，具体的な反例（？）をもとに考えることもできます．1つの問題に複数のやりかたでアプローチすることは「考える力」を養う上でとても効果的ですから，皆様も1つの方法でできたと言って次に進んでしまうのでなく，複数のやりかたにチャレンジすることを勧めます．具体例の方はそれこそたくさん作れますし．

また，そういうチャレンジをすることで，本稿の「種明かし」のような法則やメソッドがいろいろ見つけられます．これも頭のトレーニングにお勧めです．

後半の設問（B）〜（D）ですが，（B）が基本となるチェックディジットの計算，（C）がチェックディジットを与えて数の1つを決めさせる問題，（D）が数の2つを決めさせる問題というふうに，難易度の幅を大きく取ってさまざまなレベルの受験者に対応しています．

設問（B）〜（D）の解き方ですが，与えられている式どおりに計算する（B）は別として，（C）と（D）は本稿でお見せしたように，アルゴリズムぽく（？）解く方法と，数学ふうに式を簡単化して解く方法が可能です．この問題ではどちらでも似たような手間ですが，これも両方のやり方になじんでおくことが，「考える力」の練習にもなりますし，どちらか片方が圧倒的に有利な問題に備える上でも有用です．こちらも問題がもっとほしければ，違うチェックディジットでやる，X，Y，Zの違うものでやる，などで無尽蔵にできます．

というわけで，京都産業大学は将来の受験者に向けて「このような試験をおこないますよ」という広報のために問題を作ったのでしょうけれど，私たちはそのレベルを超えて「練習になる，ためになる」問題を手にいれたような気がします．ぜひ活用してください．

参考文献

1）小宮常康：国公立大学における情報入試，情報処理，Vol.65，No.2，pp.e6-e9 (2024).
http://doi.org/10.20729/00231768
2）谷　聖一：私立大学における情報入試，情報処理，Vol.65，No.2，pp.e10-e13 (2024).
http://doi.org/10.20729/00231769
3）京都産業大学：入試情報サイト 2025年度入学者選抜「情報」模擬問題（サンプル問題），
https://www.kyoto-su.ac.jp/admissions/exam/entrance/general_infoplus/sample.html
4）日本図書コード管理センター：ISBNの検算，　https://isbn.jpo.or.jp/index.php/fix__calc_isbn/
5）一般財団法人流通システム開発センター：GS1事業者コード・GTIN（JANコード）とは，
https://www.gs1jp.org/code/jan/about_jan.html

（2024年2月21日受付）
（2024年4月1日note公開）

■**久野　靖**（正会員）
電気通信大学 特命教授，筑波大学 名誉教授．理学博士．プログラミング言語，ユーザインターフェース，情報教育に興味を持つ．

情報処理学会ジュニア会員へのお誘い

小中高校生，高専生本科～専攻科1年，大学学部１～3年生の皆さんは，情報処理学会に**無料で入会**できます．**会員になると有料記事の閲覧，情報処理を学べるさまざまなイベントにお得に参加できる等のメリットがあります**．ぜひ，入会をご検討ください．入会は**こちら**から！