

# IT リテラシー I のエッセンス

～関数の引数に着目して～

横井 隆志

## はじめに

### 1. IT リテラシーの開講

2011 年度後期に開講して以来、2013 年度後期までに、入学前学修を含めて 7 回にわたって IT リテラシー (2013 年度より IT リテラシー I に改称) を実施してきた。開講の経緯は本会計大学院 紀要 今号に別に掲載した座談会で語っているので省略するが、シミュレーション・アプローチを具現化する基礎を造るために、主に表計算アプリケーション (Microsoft Office Excel) の基本的な使用法を習得させることを授業の到達目標に据えている。2011 年度後期の開講から現在までの間、当科目の対象が会計実務に携わる社会人であることを前提に、いかにして履修者に IT リテラシーのエッセンスを伝え、どうすれば修得させることができるかを念頭に置いて、実習課題や視覚資料、授業の実施方法は毎回アップデートを行ってきた。

開講以前は、表計算の必要性を強く認識しており、実務でも一定程度使用しているとされる履修者の習熟度やレベル感がどの程度で、基礎科目である IT リテラシーのスタート地点をどの程度の位置に設定すれば良いか迷う部分が少なからずあった。しかし、開講前のアンケートで使用状況を具体的に聞いていたり、実際に授業を実施したりしてみると、多

くの履修者が実務で Excel を“使っている”としている内容は、簡単な表の作成や、ほとんど完成されたシート上の限定された箇所への入力、あるいは、ごく基本的な特定の関数の使用に限定されることが明らかになった。それを受けて、Excel の基本操作と関数の使用法を修得することにより、ごくごく簡易的なシステムを構築できるだけのスキルの修得を IT リテラシーの到達目標に設定した。

### 2. IT リテラシー II の開講

IT リテラシー開講時より、履修者には一定程度の満足度を得ていただくことができた一方で、簡易的なシステム構築のための VBA によるプログラミングなど、より高度な Excel の利用を学べる IT リテラシーの上位科目を開設して欲しい、といった要望は常に寄せられていた。これを受けて、2013 年度より、LEC 会計大学院 修了生で IT リテラシー開講時よりゲスト講師として毎期末に登壇いただいた徐 伯昇 氏を兼任講師としてお招きし、筆者と共同で担当する科目として IT リテラシー II を開設するに至った。

### 3. 顕在化した課題

IT リテラシー II の開講を意識して、2013 年度前期の IT リテラシー I は徐先生にも同席いただくかたちをとった。その中で提案さ

れたのが「関数の確認テスト」を実施することであった。関数に含まれる引数がどのように機能し、結果として何をどのようなかたちで返すか、を記述してもらうかたちで実施した。そこで明らかになったのは、ほとんどの履修者が実習課題に向かう際にはある程度正確に関数を挿入して機能させることができる反面、関数の働きを言語化できる状態で正しく理解しているとは言えない現状であった。もっとも、関数にどのような引数が含まれ、それらをどのように指定するかは関数ウィザードやヘルプが補助してくれるし、ある程度の解説も相当数、Web 上で参照することができる状態にあるため、当初より、関数を記憶することは求めていない。しかし、複数の関数をネストする（ある関数の引数として関数を入れ子にする）場合などを想定すると、関数の引数をどのようなかたちで指定するか、何をどのようなかたちで返すかを正確に理解することが最低限、求められる。

また、IT リテラシーⅡが開講した後、データベースの基本的な考え方であるフィールドとレコードの概念など、IT リテラシーⅠで修得すべき基本的なエッセンスが過去に IT リテラシーⅠを受講した履修者の中で抜け落ちていることが顕在化した。

別の機会に筆者の授業を受講されたある受講生からは、資格試験を念頭に書かれているテキストを一通り学習し、試験にも合格したものの、その後、学習したことを複合的に組み合わせ、応用的に活用するところまでは至っていないとの声を聞かされた。さらに別の受講生からは、提示された課題に沿って実習を行う時にはスムーズに理解できるが、実際の要請に従って Excel の機能や関数を用いようとしたときに、なにをどのように使ったら良いかがわからない、との声が寄せられた。

## “Memory” から “Understand” へ

2013年4月に九州共立大の経済学部長に就任した、筆者が修士課程の時代にご指導いただいた恩師、船戸高樹先生は、学部教育で最も重視する方針のひとつに“Memory より Understand”を掲げている。即ち、単純に記憶する、暗記するのではなく、理解をすることを重視して教育を展開されている。船戸先生が示唆されるように、基本的な内容を正確に理解することこそ、学修した知識を短期的な知識に止まらず、自身のスキルとして創造的に活用できる力に育てる鍵であると考えられる。

LEC 会計大学院紀要 第 10 号に掲載された拙稿では、Excel の強みとして 1. 計算過程が残ること、2. 一覧性があること、3. 加工の容易さ、4. 関数の存在、5. VBA による拡張性を挙げ、Excel がエンドユーザーによる簡易システム開発環境となり得ることを指摘した。あわせて、関数をモジュールとしてとらえ、ネストすることによって処理の幅を広げることや、セルの入力規則の活用等による簡易的な GUI (Graphical User Interface) の実装により、Excel の操作に明るくないユーザーにも使いやすい汎用性をもった簡易システムを構築できる可能性を示唆した(横井, 2013)。

これらを実現するために、ふたつのポイントを挙げたい。ひとつには、個々の関数の働きを明確に言語化して伝え、受講者すなわち利用者自身もその働きについて言語化できる状態にすること、ふたつ目は、個々の関数の働きを言語化する中で、その関数が引数としてどのような要素を用いて (インプット) 何をどのようなかたちで返すのか (アウトプット) を明確に示せる状態にすること、である。

繰り返しになるが、実際に Excel と向き合

い、ワークシートの作成に取り組む際には、ヘルプや Web 上のコンテンツ、各種ウィザードが作業を助けてくれるため、個々の要素を記憶する必要は毛頭無い。その上で、関数であれば関数ウィザード等で必要な引数等が示されたときに、それらがいかんして働き、何をどう返すか、という具体的なイメージを描けるよう、学習した時点で理解しておくことが求められよう。また、実務などの様々な場面において何らかの必要性が生じた場合、この関数とこの関数をこのように組み合わせることができれば、このようなかたちで対応が可能である、ということをおある程度具体的にイメージできる程度に理解しておくことが求められる。

これらを念頭に、Excel による IT リテラシーの基礎を成す主要な関数について、いまいちど、初学者に語りかけるつもりで解説を試みたい。

## 関数の構造と機能

### IF 関数

IF 関数は論理式、真の場合、偽の場合の 3 つの引数から成り、論理式の成否によって二

者択一の条件分岐を行う関数である。論理式が成立するときには真の場合を、成立しないときには偽の場合を返す (図-1)。

=IF(論理式, 真の場合, 偽の場合)

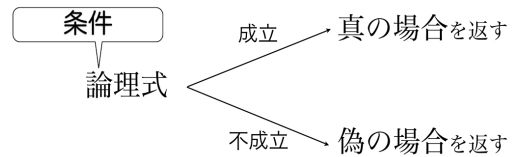


図-1 IF 関数実行のイメージ

その働きは明快であり、単独では 2 つの選択肢にしか対応しないものの、複数の IF 関数をネスト (入れ子) することにより、枝分かれを繰り返すかたちでより多くの選択肢に対応することができるため、実務その他で最も活用されている関数のひとつと言えよう。

より正確に IF 関数の働きを理解するために、実際に関数を入力する際に引数の入力を手助けする関数ウィザードを使って、ひとつひとつの引数について詳しく見てみよう。

関数の挿入ボタンをクリックし、[関数の分類]で論理のグループを選んで IF 関数を選択すると、ウィザード内に論理式、真の場合、偽の場合の 3 つの引数の入力欄が表れる。

図-2 関数ウィザードによる IF 関数の引数の入力画面

それぞれの引数入力欄の右側を見てみると、引数を直接指定、あるいは参照、計算した結果、個々の引数という単位で何を返しているかを確認することができる。例えば、論理式に〇〇=△△と指定し、それが成立している場合には引数入力欄の右側に“TRUE”と表示される。論理式に〇〇=××と指定し、それが成立しない場合には“FALSE”と表示される。言い換えれば、指定した論理式が成立する場合には” TRUE” を、成立しないときには”

FALSE “を返している、ということになる。すなわち、IF 関数は、論理式が TRUE を返すときには真の場合を、FALSE を返すときには偽の場合を返す、と言える。

同じように、TRUE や FALSE のかたちで結果を返す関数を論理関数という。関数ウィザードで「論理」のグループを選択すると IF 関数も含めて 7 つの関数が一覧に表示される。

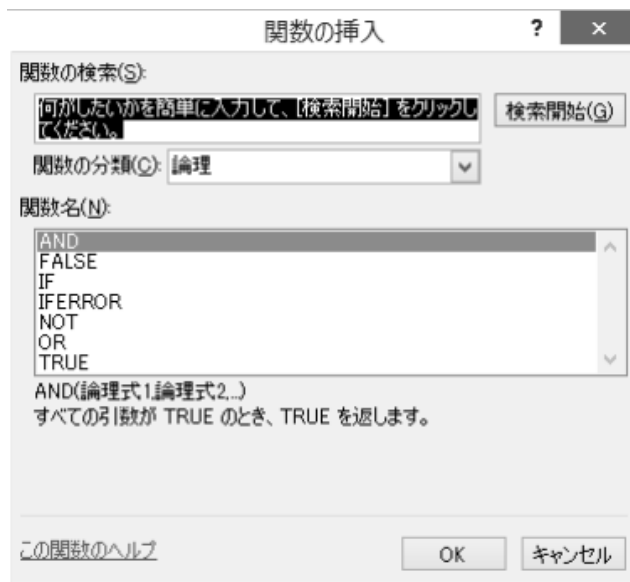


図-3 論理関数の選択画面

### 論理関数の性質

図-4 では、3 名の論文について 2 名の教員が評価を行ったと仮定する。このシートを使

って、AND 関数、OR 関数、NOT 関数の 3 つの働きをそれぞれ見ていこう。

	A	B	C	D	E	F
1	論文評価シート					
2	氏名	佐藤先生	鈴木先生	AND	OR	NOT
3	青木聡	A	A	TRUE	TRUE	FALSE
4	津川祥子	A	B	FALSE	TRUE	FALSE
5	松永弘	B	B	FALSE	FALSE	TRUE

図-4 論理関数実行のイメージ

## AND 関数

=AND(論理式 1, 論理式 2,...)

AND 関数の引数には 2 つ以上の論理式を指定し、指定した全ての論理式が成立する、すなわち TRUE を返す場合には TRUE を、そうでない場合は FALSE を返す。

セル D3 では、青木さんの評価について佐藤先生、鈴木先生のいずれもが A 評価をつけた場合には TRUE が、どちらか一人でも A 以外の評価をつけた場合には FALSE が返されるよう、AND 関数が入力されている。

=AND(B3="A", C3="A")

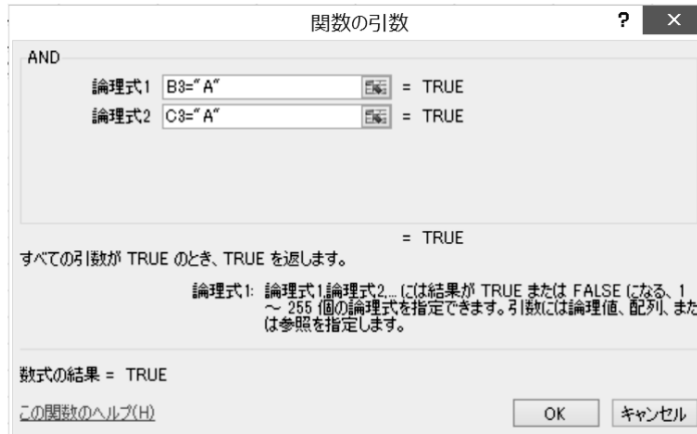


図-5 AND 関数の引数入力画面① TRUE が返される場合

佐藤先生の評価が A である、という論理式 B3="A", 鈴木先生の評価が A である、という論理式 C3="A"のいずれもが成立し、TRUE を返していることから、AND 関数の結果としても TRUE が返されていることがわかる。津川さんの評価について同様に AND 関数を入力すると、

=AND(B4="A", C4="A")

この場合、鈴木先生は B をつけているので 2 つ目の引数 C4="A"が FALSE を返し、AND 関数の結果としても FALSE が返される。



図-6 AND 関数の引数入力画面② FALSE が返される場合

セル D5 で松永さんの評価について AND 関数を入力すると

=AND(B5="A",C5="A")

B5="A",C5="A"のいずれの論理式も FALSE を返すため、AND 関数の結果としても FALSE が帰ってくる。これが AND 関数の働きである。

### OR 関数

=OR(論理式 1, 論理式 2,...)

OR 関数も AND 関数と同様、引数には 2 つ以上の論理式を指定する。OR 関数は、指定した複数の論理式のうちいずれかが成立する、すなわち、ひとつでも TRUE を返す論理式が含まれる場合には TRUE を、全ての論理式が成立しない、すなわち全ての論理式が FALSE を返す場合には FALSE を返す。

セル E3 に

=OR(B3="A",C3="A")

と入力する。

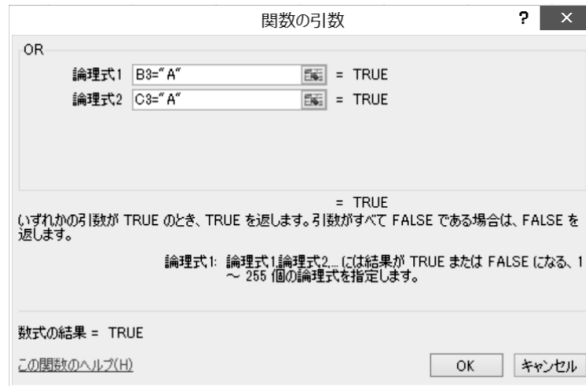


図-7 OR 関数の引数入力画面① TRUE が返される場合

この場合、指定した B3="A", C3="A"のいずれの論理式も成立し、TRUE を返してきており、OR 関数の実行結果としても TRUE を返している。

次に、セル E4 に以下の様に入力する。

=OR(B4="A",C4="A")

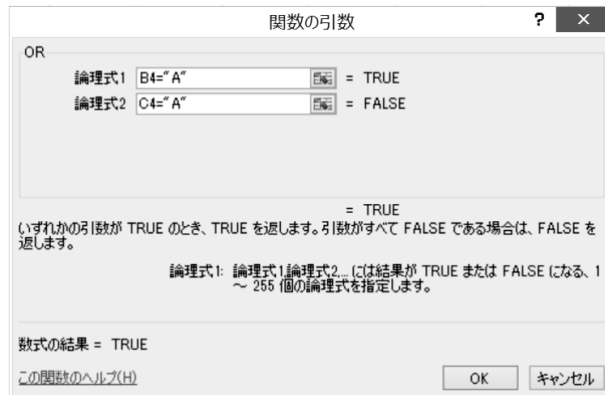


図-8 OR 関数の引数入力画面② TRUE が返される場合 その 2

この場合、ひとつ目の論理式 B4="A"は成立して TRUE を、ふたつ目の論理式は C4="A"は成立せず、FALSE を返しているが、OR 関数はひとつでも TRUE を返す論理式を含んでいれば TRUE を返すため、結果として TRUE が返されている。

今度はセル E5 に

=OR (B5="A", C5="A")

を入力する。

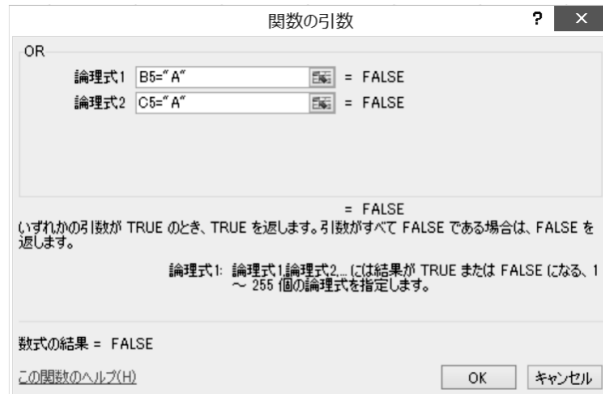


図-9 OR 関数の引数入力画面③ FALSE が返される場合

この場合は、B5="A", C5="A"のいずれの論理式も成立しないため、OR 関数の結果としても FALSE を返している。

AND 関数、OR 関数ともに、全ての論理式が TRUE を返す場合、は TRUE を、全ての論理式が FALSE を返す場合に FALSE を返す点は共通しているが、TRUE を返す論理式と FALSE を返す論理式が混在する場合に AND 関数は FALSE を返すのに対し、OR 関数は TRUE を返す、という点が異なる。

## NOT 関数

=NOT (論理式)

NOT 関数の引数には単一の論理式を指定し、論理式が成立する、すなわち TRUE を返す場合には FALSE を、成立しない場合、すなわち FALSE を返す場合には TRUE を返す。

セル F3 に以下の式を入力する。

=NOT (B3="A")

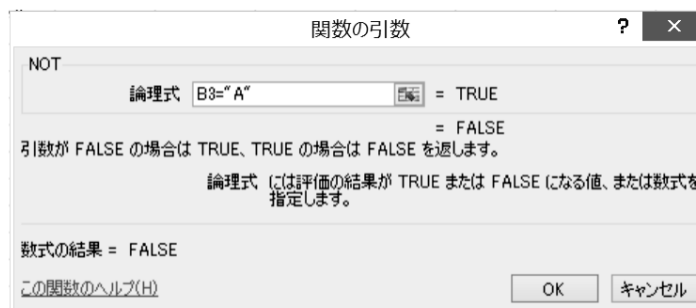


図-10 NOT 関数の引数入力画面① FALSE が返される場合

すると、B3="A"という論理式が成立し、TRUE を返すため、NOT 関数の結果としては FALSE が帰ってくる。セル F4 に入力した式

=NOT (B4="A")

でも同様である。逆に、セル F5 に

=NOT (B5="A")

と入力した場合には、論理式 B5="A"が成立せず、FALSE を返すため、NOT 関数の結果としては TRUE が返される。

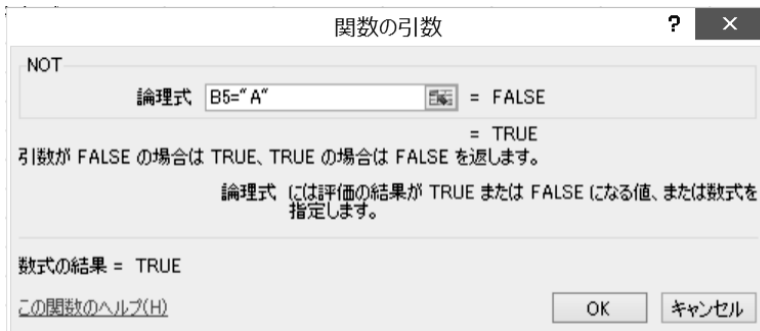
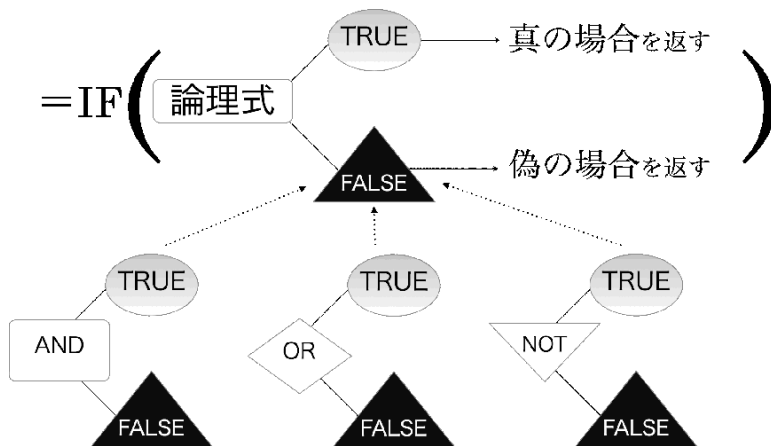


図-11 NOT 関数の引数入力画面② TRUE が返される場合

IF 関数の論理式が TRUE と FALSE の何れかを返していることを認識しておけば、これら

の論理関数を IF 関数の論理式にネストすることは容易にイメージできるであろう。



AND 関数、OR 関数、NOT 関数のいずれも TRUE または FALSE で結果を返すため、同様に TRUE か FALSE の何れかを返す IF 関数の論理式にネストし、条件分岐させることができる。

図-12 IF 関数と AND, OR, NOT の 3 つの論理関数の実行イメージ

これらの論理関数に代表されるように、Excel には、単独では極めてシンプルな機能を提供し、単独での使用よりもむしろ他の関数とネストすることにより大きな相乗効果を発揮する関数が多数、用意されている。IT リ

テラシー等の授業でこれらの関数を扱う際、当初は、簡単にその関数の構成を確認した後、はじめから関数にネストする方法を扱っていたが、現在は必ず、ひとつひとつの関数が単独でどのように機能し、何をどのように返す



かを例題により確認するようにしている。それは、実際に扱ったネストの例に用途を限定されないようにすること、そして、その関数の働きを正確に理解することが様々な要請に応じて創造的に関数を組み合わせるための基礎となるからである。

AND 関数や OR 関数を IF 関数の論理式にネストすることにより、より柔軟に条件設定を行うことが可能となる。

なお、NOT 関数は単独で IF 関数の論理式に用いる必要があるケースは考えにくいですが、例えば、AND 関数で論理式 a、論理式 b が成立し、かつ、論理式 c が成立しない場合に TRUE を返したい、といった場合に有用であろう。

例)=AND(論理式 a, 論理式 b, NOT(論理式 c))

## VLOOKUP 関数

=VLOOKUP(検索値, 範囲, 列番号, 検索方法)

VLOOKUP 関数は検索を行う関数であり、検索値、範囲、列番号、検索方法の 4 つの引数により構成される。まとまったデータの中から任意の値を抽出することができるため、その活用の幅は広く、IF 関数と並んで様々な場面で最も多く活用されている関数であるといえよう。一方で、IT リテラシー教育の現場では多くの受講生がつまづくポイントのひとつであり、Excel を扱う市販の様々な解説書を見ても、正確な理解を得るために必ずしも十分な説明がなされているとは言えない状況である。その問題意識から横井(2011)でも解説を試み、何名かの読者から VLOOKUP 関数というものをようやく理解できたとの反響を賜った。そこでは「VLOOKUP 関数は、検索値を範囲(対照表)の 1 列目と照らし合わせ、該当す

る行の指定した列番号に位置する値を返す」とその働きを解説した。より平易な言葉にすれば、

**検索値を範囲の 1 列目の上から下へ向かって指定した検索方法に従って順に検索していき、該当する行で、列番号で指定した位置にあるセルの中身を返す**

という言い方ができるであろう。本稿では、より効率的な VLOOKUP 関数の入力やより実践的な活用を念頭に、VLOOKUP 関数を構成する引数の性質に着目したい。

それでは、VLOOKUP 関数の 4 つの引数について見ていこう。

検索値は、指定した範囲からデータを抽出するためのキーとなる要素であり、これを範囲の 1 列目で検索する、すなわち、検索値と範囲の 1 列目とを照らし合わせることになる。従って、範囲(一般に、対照表とも呼ばれる)を作成する際は、その 1 列目には検索値に対応した形式のデータが配置されている必要がある。

検索方法は、TRUE か FALSE か、という論理値によって指定する。TRUE を指定した場合には近似値も含めて、FALSE を選択した場合には完全に一致するもののみが検索される。なお、TRUE を指定して近似検索を行うことで、指定した範囲に該当するかどうかを検索することが可能になるが、その場合は、範囲の 1 列目はそれぞれの範囲の下限の数値を昇順に配置する必要がある。

列番号は検索によって行が確定された後、該当する行でどの列を返すかを示す。範囲で指定した領域の中で左から相対的に何列目を返すかを、2 列目なら“2”, 3 列目なら“3”というかたちで直接指定する。

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	製品コード	製品名	カテゴリ	タイプ	CPU	ストレージ	メモリ	ディスプレイ	Opt. Drive	USB	映像出力	価格
3	0101	StraPro	Pro	Desktop	3.0 GHz 2TB HDD		8 GB 別売	BD-R		8 Port 2×Thunderbolt		¥248,000
4	0102	StraPro	Pro	Desktop	2.6 GHz 1TB HDD		4 GB 別売	DVD-Spr		8 Port 2×Thunderbolt		¥189,800
5	1101	iStra	Consumer	Desktop	2.8 GHz 1TB HDD		4 GB 27inch HD	BD-R		4 Port 1×Thunderbolt		¥158,800
6	1102	iStra	Consumer	Desktop	2.4 GHz 1TB HDD		4 GB 21inch HD	DVD-Spr		4 Port 1×Thunderbolt		¥164,800
7	0201	StraBook Pro	Pro	Note	2.4 GHz 320GB SSD		4 GB 15inch	DVD-Spr		3 Port 1×Thunderbolt		¥139,800
8	0202	StraBook Pro	Pro	Note	2.26 GHz 500GB HDD		2 GB 15inch	DVD-Spr		3 Port 1×Thunderbolt		¥119,800
9	1201	iStraBook	Consumer	Note	1.8 GHz 128GB SSD		4 GB 12inch	none		2 Port 1×Thunderbolt		¥119,800
10	1202	iStraBook	Consumer	Note	1.8 GHz 160GB HDD		2 GB 12inch	none		2 Port 1×Thunderbolt		¥99,800
11												
12	製品コード	製品名	カテゴリ	タイプ	CPU	ストレージ	メモリ	ディスプレイ	Opt. Drive	USB	映像出力	価格
13		0201	StraBook Pro									
14												
15												

図-13 VLOOKUP 関数の使用例

図-13 では、A13:A15 にそれぞれ製品コードを入力することで、B 列以降に製品一覧から 3 製品分のスペック等を抽出して比較する。まず、セル B13 に以下の式を入力し、ひとつ

目の製品名を抽出する。

=VLOOKUP(A13, A2:L10, 2, FALSE)

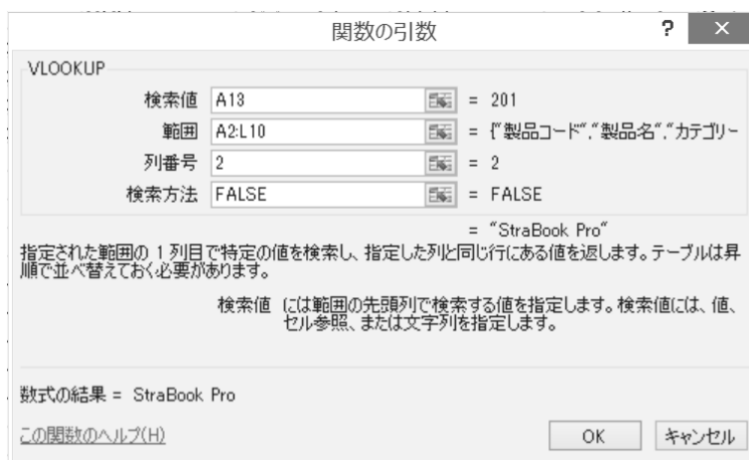


図-14 VLOOKUP 関数の引数入力画面

検索値には 1 行目の製品コードであるセル A13 を参照する。

範囲は 1 列目が検索値すなわち製品名に対応するように、さらには列番号の指定によって価格までの一通りの要素を抽出できるように、A2:L10 を指定する。このとき、各項目の項目名（フィールド名）が入力されている行を一緒に選択してしまっても範囲の先頭行はフィールド名と自動認識される。

列番号は、範囲に指定した A2:L10、つまり A 列〜L 列の中で相対的に何列目を返すか、ということになる。製品コードが 1 列目、製品名が 2 列目、カテゴリが 3 列目・・・とな

り、セル B13 では製品名を返すので 2 列目の“2”を指定する。

検索方法は、範囲の 1 列目、すなわち A3:A10 の配置をみると昇順に配置されているわけではなく、また、製品コードはひとつの製品に 1 つずつ与えられた一意の数値であるため、完全一致検索を行うよう、FALSE を指定する。

式の入力が完了したらオートフィルで水平方向、垂直方向へ複写することを念頭に、参照しているセル番地を適宜固定する。

=VLOOKUP(\$A13, \$A\$2:\$L\$10, 2, FALSE)

これをオートフィルによって水平方向へ L13 まで複写し、さらにそのまま垂直方向へ 15 行目まで複写すると、複写先では各行の製品コードに対応した製品名が返される。これは、指定した範囲の中での列の位置を表している列番号は、相対参照しているセル番地とは性質が異なり、オートフィルをしても自動的に変化することがないためである<sup>注1)</sup>。これに対応する方法として、横井(2011)では、列番号だけが入力された行を用意しておいて、VLOOKUP の列番号の引数としてその行の適切なセルを相対参照する方法、MATCH 関数を用いる方法を紹介した。そこで、次の様な質問を受けた。それは、与えられた実習課題で実際に VLOOKUP 関数に MATCH 関数をネストしてみると、MATCH 関数が VLOOKUP 関数の列番号を補い得ることが理解できるが、様々な関数が存在する中でなぜ MATCH 関数を VLOOKUP 関数の列番号にネストするという判断に至ることができるのか、という問いである。この問いへの答えも、ひとつひとつの引数の性質を明確に理解することによる関数の正確な理解こそが異なる関数を結びつける鍵となる、と言える。この点を意識しながら、最後に MATCH

関数をあらためて見てみよう。

## MATCH 関数

MATCH 関数は検査値、検査範囲、照合の型の 3 つの引数から成る関数である。

=MATCH(検査値, 検査範囲, 照合の型)

MATCH 関数は、検査範囲の中で検査値を照合の型に従って検査し、上から、あるいは左から相対的に何番目に位置するかを返す関数である。言うまでもなく、検査値と検査範囲が対応関係にある必要がある。また、検査値が検査範囲の中で上から、あるいは左から相対的に何番目に位置するかを返すという性質上、検査範囲を複数の行・列にまたがって指定することはできない。

照合の型は、検査範囲が昇順で並んでいる場合に近似するものを検査する場合は 1 を、検査値に完全一致するものを検査する場合は 0 を、検査範囲が降順で並んでいる場合に近似するものを検査する場合は -1 をそれぞれ指定する。

	A	B	C	D	E	F	G
1	課題提出状況						
2							
3	日付		提出者	評点		名前入力欄	提出順
4	1月11日	022	小泉	10		柴崎	
5		11.02	小林	8			
6		15.14	山田	9			
7		21.07	柴崎	10			
8		22.09	海原	6			
9		23.01	岡田	10			
10	1月12日	2.02	池本	8			
11		1.12	島田	7			
12		9.08	橋口	8			
13		11.12	兼子	10			

図-15 MATCH 関数の使用例

図-15 は、課題の提出状況が提出日時・評点とあわせて入力されたシートである。ここ

で、セル F4 に提出者の名前を入力すると、その学生が何番目に提出したかをセル G4 で返

したい。そこで、セル G4 に MATCH 関数を入力する。

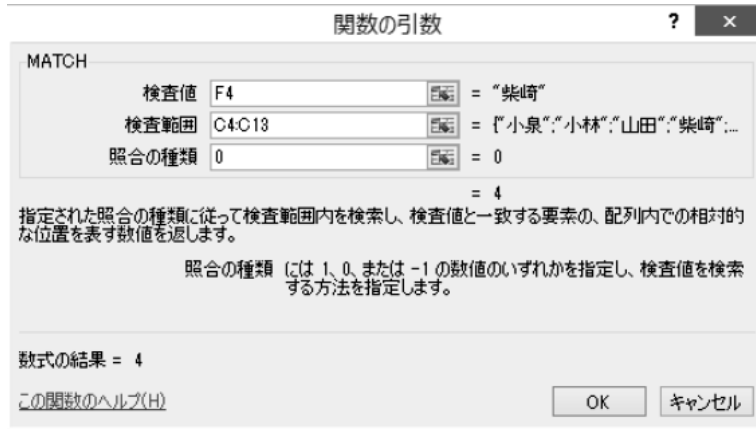


図-16 MATCH 関数の引数入力画面

検査値には、提出順を調べたい学生の名前を入力するセル F4 を参照する。

検査範囲は、セル F4 に入力する学生の名前に対応する、提出順を調べる元になる範囲、すなわち、C4:C13 を指定する。

照合の種類には、今回は検査値に指定したセル F4 に入力した文字列と完全に一致するものが C4:C13 の中で何番目に位置するかを返したいので、完全に一致するものを検査するため“0”を指定する。すると、以下の式が完成する。

=MATCH(F4, C4:C13, 0)

すると、セル F4 に入力された “柴崎” という文字列は「小泉、小林・・・」と並んでいる中で 4 番目に位置するものと完全に一致することがわかり、結果として 4 が返される。

このように、MATCH 関数が指定した検査範囲の中で検査値が相対的に何番目に位置するかを返す関数であることを認識すると、その返され方、すなわちジグソーパズルのピース

が、VLOOKUP 関数の列番号に共通し、あてはまるとイメージすることができるであろう。

## おわりに

本稿で扱った関数は以前にも一通り解説を行ったものであるが、今回、学習者の知識をより確かなものにするために、7 回の IT リテラシー I の授業の経験を踏まえて再度、解説を試みた。前回はひとつひとつの関数を平易な言葉で表現することに主眼をおいたものであったが、今回は更に、関数をネストさせてより実践的に活用するための土台を確かなものにすることを目指し、関数を構成するひとつひとつの引数に着目し、その働きや結果の返し方に焦点を当てた。ユーザーが構築する論理をスプレッドシートに具体的に表現する際の助けになれば幸いである。

受講者、あるいは諸先生方のフィードバックを受けてここに至ったプロセスは、ある目的に向かって関数などによりワークシートを構築するという行為そのものが論理を整理し、

筋道立てて具体化する行為であること、そして、IT リテラシーとは論理的思考とそれを具現化する方法論を併せ持つことである、ということを書者の中であらためて明確にした。

最後に、徐先生をはじめ諸先生方や積極的な受講者の有益なフィードバックに対し、心からの感謝を述べ、本稿を締めくくりたい。

---

**(注記)**

注 1) 横井 (2010) の図 7 でも示したとおりである。

**(参考文献)**

- 横井隆志. 2010. 「Excel による IT リテラシー教育(1)」(研究ノート)LEC 会計大学院紀要 第 8 号
- 横井隆志. 2013 「創造的エンドユーザーのための IT リテラシー」LEC 会計大学院紀要 第 10 号