

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第4044158号  
(P4044158)

(45) 発行日 平成20年2月6日(2008.2.6)

(24) 登録日 平成19年11月22日(2007.11.22)

(51) Int.Cl.

F 1

G06F 17/30 (2006.01)  
G06F 17/21 (2006.01)G06F 17/30 170A  
G06F 17/30 380E  
G06F 17/21 550J

請求項の数 1 (全 13 頁)

(21) 出願番号	特願平8-90030
(22) 出願日	平成8年3月19日(1996.3.19)
(65) 公開番号	特開平9-259148
(43) 公開日	平成9年10月3日(1997.10.3)
審査請求日	平成15年3月17日(2003.3.17)
審判番号	不服2006-16865(P2006-16865/J1)
審判請求日	平成18年8月3日(2006.8.3)

(73) 特許権者	596050355 加藤 雄二 神奈川県相模原市上鶴間本町 6-28-2 O
(72) 発明者	加藤 雄二 神奈川県相模原市上鶴間 1755-23

合議体  
審判長 田口 英雄  
審判官 長 由紀子  
審判官 多賀 実

最終頁に続く

(54) 【発明の名称】 符号照合装置及び符号照合方法

## (57) 【特許請求の範囲】

## 【請求項 1】

抽出する対象となる符号の種類を記憶して、符号を直接指定することなく範囲で指定する符号指定部と、

処理対象となる文書を探索して、符号に該当する文字が連続していれば連続する全ての文字をひとかたまりの符号と判断し、その文書中の符号を、符号として使用されていないものも含めて同時に抽出する符号抽出部と、

この符号に隣接して配置された文字列を、符号を付加した単語自体を指定することなく、自動的に一定長切り取って抽出する隣接文字列抽出部と、

前記符号と符号の抽出位置と前記抽出文字列とを一体に取り扱うレコードを、前記符号を主キーとし、前記符号の直前の1文字を従キーとしてソートするソート部と、

ソートされたレコードに含まれ前記符号に隣接して配置された一定長以下の文字列とその符号と符号の抽出位置とを、ソートされたレコードの順に表示出力する表示部とを備えたことを特徴とする符号照合装置。

## 【発明の詳細な説明】

## 【0001】

## 【発明の属する技術分野】

本発明は、特許出願明細書のような文書中から、符号付きの文字列を自動的に抽出して、符号付けミス等を点検するための出力を得る符号照合装置に関する。

## 【0002】

### 【従来の技術】

例えば特許出願明細書では、図面に記載された装置の各部品に連続番号のような符号をつけて、その符号を引用しながら、その装置の動作説明等を行う。

しかし、説明をすべき装置の部品点数や図面枚数が増えると、符号も多種類となって、重複して別々の部品に同一の符号をつけたり、同一の部品を参照するときは同一の符号をつけるべきところを符号を付け間違えたりする。

前者の例としては、「ボルト 15 とナット 15 とは、・・・」といった表現になる。また、後者の例としては、「この装置は車輪 23 を備えている。この車輪 22 は車軸 25 の両端に・・・」といった表現になる。

この他に、同一の部品を呼ぶ場合に、その呼び方が一致しないことがある。例えば、「この接続装置 21 は・・・、この連結装置 21 と・・・」といった表現も、しばしば見うけられる。こういった表現があると、読むものは意味の把握に迷ったり、誤解したりして、権利解釈上の争いを生じることもあり得る。

### 【0003】

そこで、明細書作成後は、明細書作成者が丹念に点検を行い、その点検により例えば、「連結装置 21」が「接続装置 21」に訂正される。

従って、この種のミスを防止するために、明細書作成担当者は、タイプアップされた原稿を何度も注意深く点検し、修正する作業を繰り返すようにしている。

### 【0004】

#### 【発明が解決しようとする課題】

このような符号付けミスは、明細書作成担当者が文章全体を丹念に点検すれば、必ず発見できるが、人間の注意力には限界がある。特に、長文の明細書であって、かなり離れた場所に符号の重複や付け間違いがあるときには、それを発見するのは容易でない。

また、当初作成した明細書にその後各種の追加修正を行ったような場合には、一挙に原稿を書き上げた場合に比べて、こういったミスが生じやすい。例えば、複数の者によって協力して明細書が作成されたような場合には、さらにミスが発生する率が高まる。

### 【0005】

しかし、特許出願後にこうしたミスを発見してもその修正の機会は制限されており、また、出願発明の要旨を左右するようなミスも皆無とはいえない。従って、符号付けミスといえどもあなどることはできない。

特許出願明細書に限らず、図面を参照しながら符号を利用して説明を行う文書は少なくない。こうした文書が大量に印刷された後にミスが発見されても、その修正は容易でない。しかも、その正誤表を作成したり、その他煩雑な手数が増加するという問題もある。

### 【0006】

これを解決するために、例えば、特定の単語や特定の符号を指定して、全文を検索して点検することも考えられる。このような全文検索機能は、ワードプロセッサやパーソナルコンピュータ用のアプリケーションプログラムの機能として良く知られている。

しかし、明細書中で使用されている符号全部についてこのような検索処理を実行するのは、入力に手間がかかり、作業性が悪く、現実的でない。また、符号の付け間違いや単語自身を別の表現にしたような場合、正しい符号や単語を用いて検索しても抽出はされないから、ミスを発見できない。

従って、文書中で使用した単語の前後に付加する符号の付け間違いをできるだけ簡便に自動的に発見することができるようなツールの開発が要望される。

### 【0007】

#### 【課題を解決するための手段】

##### 構成 1

抽出する対象となる符号の種類を記憶して指定する符号指定部と、処理対象となる文書を探索して、その文書中の上記符号を抽出する符号抽出部と、この符号に隣接して配置された文字列を抽出する隣接文字列抽出部と、上記符号と符号の抽出位置と上記抽出文字列とを一体に取り扱うレコードを、上記符号をキーとしてソートするソート部と、ソートされ

10

20

30

40

50

たレコードに含まれ上記符号に隣接して配置された一定長以下の文字列とその符号と符号の抽出位置とを、ソートされたレコードの順に表示出力する表示部とを備えたことを特徴とする符号照合装置。

【0008】

説明

抽出する対象となる符号には、数字、アルファベット、記号等、任意のものが指定できる。指定の仕方は、特定の符号を直接指定してもよいし、0～9といった範囲を指定してもよい。即ち、処理対象となる文書中で、他の文字と区別できるような任意の文字が符号として指定できる。また、符号は1文字でも2文字以上の組み合わせで構成されていてもよい。

10

処理対象となる文書のデータ形式は任意である。この文書中には、符号を付加した多数の単語が使用されている。この装置は、その単語を含む文字列をこれに付加した符号とともに抽出する。ただし、単語自体を直接指定はしない。指定された符号を文中から全て抽出し、その符号の前又は後ろに位置する一定長以下の文字列を自動的に抽出することを特徴としている。

【0009】

これにより、符号とそれに付随している単語は漏れなく抽出できるし、符号に直接隣接していないなくても、その近くにある単語も無条件に抽出できる。なお、符号として使用されているわけではない「図1は、」等の数字も、符号と一緒に抽出される。これは、ノイズとして無視したり、一定の手法で排除する。

20

符号と符号の抽出位置と抽出文字列とを一体に取り扱うレコードを符号をキーとしてソートするのは、これらのレコードの全部または一部をソートされた順に表示するためである。符号をキーとしてソートした結果を表示出力すれば、同一符号を付加した文字列を近くに並べて、その比較を行って文字列の不一致を発見できる。また、符号を付加した単語以外の文字列を抽出して表示するのも、文書中で該当する部分を探索し易くするためである。

従って、これらのレコードはデータそのものでも、抽出位置等のデータをアクセスするポインタの集合であってもよい。また、レコードを構成するデータの種類は任意である。

【0010】

表示部は、ディスプレイやプリンタ等のようにレコードに含まれるデータの内容を表示出力することができる任意の装置でよい。

30

なお、表示出力する文字列を一定長以下としたのは、表示出力した場合に、出力用紙やディスプレイの画面上で各レコードを比較し易くするためである。即ち、同一の符号を付加した文字列を比較するには、各文字列を1行ずつ並べて表示することが好ましいからである。

また、一定長以下であるから、それ以下の文字数の文字列を表示することは差し支えない。文の頭の部分や短文ではごく短い文字列が抽出され得るからである。抽出する最大文字列はメモリの許す限り、自由に設定してよい。

符号の抽出位置を表示出力するデータに含めたのは、符号やその前後に付加する文字列に誤りを発見した場合に、文書中で該当する部分をすぐに探索できるようにするためである。従って、符号の抽出位置を、例えば、「第16段落第5文の第2文字目」というように直接表示するほか、隣接する文字列の位置等により間接的に表示してもよい。また、ディスプレイ上に表示した符号をクリックすると処理対象文書の該当位置が表示されるようなハイパーテキスト形式とするのもよい。

40

【0011】

構成2

構成1において、表示部が表示出力する文字列の長さは、照合対象となる符号を付加した単語のうちの語長が最長のものを表示できる長さに選定されることを特徴とする符号照合装置。

説明

50

任意の語長の単語に符号が付加されているとき、全ての単語と符号との関係を表示出力させようすれば、最長の単語も表示できることが好ましい。そこで、表示出力する文字列の長さは、語長が最長のものを表示できる長さに選定した。なお、単語自体を意識して文字列を抽出するわけではないから、表示出力される文字列の長さは一定以下である。

#### 【0012】

##### 構成3

構成1において、表示出力する一定長以下の文字列が、抽出した符号の前方のものか後方のものかを選択するための、文字列位置選択部を備えたことを特徴とする符号照合装置。

##### 説明

「ねじ3」というように、文字列の後ろに符号を付けることに決めていれば、符号の後方に隣接する文字列の表示は不要である。しかも、符号の前方に隣接する文字列をできるだけ多く表示させたい。そんな場合には、必要な文字列のみを表示する。逆の場合も同様である。そこで、一方のみを選択して表示出力できるようにした。

#### 【0013】

##### 構成4

構成1において、符号抽出部は、符号として指定された文字が2以上連続して配置されているときはこれらを一組の符号として取り扱うことを特徴とする符号照合装置。

##### 説明

符号は「ねじ2」とか「ボルト4」というように一文字の場合もあるし、「ナット35」とか「ビス40A」というように2文字以上の場合もある。符号が連続して配置されている限り、それが何文字でも一組の符号として使用するのが一般的だから、このような処理をした。これにより、予め、全ての符号を、その組み合わせまで指定しなくても、文中から1文字または2文字以上を組み合わせた符号を自動的に抽出できる。

#### 【0014】

##### 構成5

構成1において、ソート部は、同一符号に隣接する各文字列を、それぞれその符号からみた距離が近いものを基準にソートすることを特徴とする符号照合装置。

##### 説明

符号として、例えば、数字の「2」が指定されているとき、「ボルト2」のみならず「図2」や「第2の」といったノイズも抽出され、これらが output に混在すると見にくくなる。そこで、符号に近い文字をキーとしてソートして、符号に隣接している文字が同じものを近くに並べるようにした。こうすれば、点検作業が容易になる。なお、ソート対象となる文字数は多いほど整理されて見やすくなるが、ソート時間が長くなる。実用上は、1~2文字で十分である。

#### 【0015】

##### 構成6

構成1において、ソート部は、2種以上の性格の異なる文字を符号として指定した場合、いずれか1種の文字を主キーに選定してソートし、表示部には、主キーに選定した符号とその他の符号とを区別して表示することを特徴とする符号照合装置。

##### 説明

1Aとか12Aといった数字と文字を組み合わせた符号は、数字のみを基準にソートして表示した方が便利な場合がある。そこで、いずれか一種の文字即ちこの例では数字を主キーとしてソートする。なお、その他の文字も従キーとしてソートするとよい。また、例えば1Aと12Aという符号を左詰めで並べて表示すると、数字部分と英字部分との境界が不明確になり見にくい。主キーに選定した符号を別の符号と区別するというのは、例えば、表示欄を別にしたり、位置合わせにより、表示位置を調整したりすることをいう。これで、表示されたレコードの比較が容易になる。

#### 【0016】

##### 構成7

処理対象となる文書と抽出する対象となる符号の種類を予め指定し、上記処理対象とな

10

20

30

40

50

る文書を探索して、その文書中の符号を抽出するとともに、抽出した各符号が符号をキーとしてソートした順番に並ぶように、各符号と、その符号に隣接して配置された所定長以下の文字列と、その符号の抽出位置とを対応させて表示することを特徴とする、コンピュータを利用した符号照合方法。

#### 説明

コンピュータを用いて文書中の符号を抽出し、その結果を見やすく表示するためには、このような方法が適する。各符号が符号をキーとしてソートした順番に並ぶようにしたのは、同一符号の付いた文字列を比較し易くするためである。対応させて表示するというのは、表示位置を対応させることをいい、これも、コンピュータの出力した文字列や符号を相互に比較し易くするためである。

10

#### 構成 8

処理対象文書から、表現を比較するための単語を含む文字列を抽出する手段と、  
抽出した単語とその前方の文字列およびまたは後方の文字列を含むそれぞれ一定長以下の後記の1行に含まれる文字列と、当該文字列の抽出位置を表示するデータとを含む、レコードを生成する手段と、  
前記各一行分の文字列を、前記表現を比較するための単語の表示位置が上下方向にみて一致するように、上下方向に1行ずつ並べて表示する表示手段を備えたことを特徴とする文字列照合装置。

#### 構成 9

文字列を抽出する手段が、処理対象文書から、表現を比較するための単語を含む文字列を抽出するステップと、  
文字列を生成する手段が、抽出した単語とその前方の文字列およびまたは後方の文字列を含むそれぞれ一定長以下の後記の1行に含まれる文字列と、当該文字列の抽出位置を表示するデータとを含む、レコードを生成するステップと、  
表示手段が、前記各一行分の文字列を、前記表現を比較するための単語の表示位置が上下方向にみて一致するように、上下方向に1行ずつ並べて表示するステップ、  
を含む、コンピュータを利用した文字列照合方法。

20

#### 【0017】

##### 【発明の実施の形態】

以下、本発明の実施の形態を、具体例を用いて詳細に説明する。

30

##### 符号照合装置の概略

図1は、本発明の符号照合装置の具体例を示すブロック図である。

本発明の装置は、例えば、テキストデータにより作成された特許出願明細書を読み込んで処理し、ディスプレイ等にこの図に示すような出力を得るコンピュータにより構成される。

図に示す処理対象となる文書1は、例えばフロッピーディスクやハードディスクに格納された文書データである。この例では、特許出願明細書の文書データの一部を図示している。この文書は例えばテキストデータ形式で格納されている。また、処理結果は表示部11に表示される。表示部11は、コンピュータのディスプレイやプリンタ等から構成される。

40

#### 【0018】

この文書1の処理を行うために、コンピュータの制御部は、符号抽出部3、隣接文字列抽出部4、バッファメモリ5、ソート部6、符号指定部7及び文字列位置選択部8という機能ブロックを備える。

この図に示したような各機能ブロックは、後で説明するような手順を実行する符号照合プログラムを一般的のコンピュータ上で実行させる際に、所定のタイミングで生成される。

#### 【0019】

符号抽出部3は、処理対象となる文書1を構成する文字を1文字ずつ検査して、符号として指定された文字を探索して抽出する機能をもつ。符号指定部7は、抽出すべき符号の種類を指定する機能を持ち、この例では、数字を符号と指定している。数字以外には、例え

50

ば、英字や「 - 」等を指定できる。

隣接文字列抽出部 4 は、こうして抽出された符号の前方あるいは後方に隣接した文字列を所定長抽出する処理を行う機能を持つ。なお前方の文字列を抽出するか後方の文字列を抽出するか、あるいは両方の文字列を抽出するかは、文字列位置選択部 8 の指定による。例えばこの例では、前方の文字列のみを抽出するように指定している。バッファメモリ 5 は、抽出された符号と隣接する文字列及びその符号の抽出位置を示すデータを組み合わせたレコードを、抽出順に一時格納しておくための記憶領域である。

#### 【 0 0 2 0 】

ソート部 6 は、こうして抽出されたレコードのソートを行い、符号順に配列して出力する部分である。なお、レコードというのは、表示部 11 に示した表の中の 1 行に含まれるデータのことである。このようなレコードは抽出された全ての符号について 1 個ずつ生成される。

これらのレコードが符号順にソートされると、図 1 の表示部 11 に示したとおりの出力結果が得られる。

#### 【 0 0 2 1 】

なお、上記各機能を実現するプログラム言語は、例えばベーシックにより記述することができる。例えば、オープンしたファイルを 1 文ずつ読み込むのは、「LINE INPUT」命令、その中から指定された符号の位置を検出するのは「INSTR」命令、指定した位置から指定した数だけ文字列を抽出するのは「MID」命令により記述することができる。バッファメモリは、抽出された符号と隣接する文字列及びその符号の抽出位置を示すデータを要素とするレコードを DIM 命令を用いて定義して保存する。このときソートキーとなるデータをレコード中に含めておけば、あとからそのキー順にソートして出力することができる。

#### 【 0 0 2 2 】

##### 符号照合処理の概要

図 2 には、処理対象文書の構造説明図を示す。

この図 2 及び図 3 を用いて、処理対象文書から符号を抽出する方法の概要を説明する。

この図は図 1 に示した処理対象の文書 1 の構造を詳細に示しており、特許出願明細書の一部を表す。一般に特許出願明細書は、段落番号により段落ごとに区切られている。この例では、段落番号の次の文を第 1 文、その次の文を第 2 文というように数えて、符号の抽出位置の表示方法を定義している。各文は改行により区切られている。ここで、例えば第 2 文中の「ボルト 12」という部分の「12」を符号として抽出したとする。「12」は「ボルト」という単語に付加された符号である。このとき、その抽出位置及びその符号の直前の例えば 15 文字の文字列が合わせて抽出される。

#### 【 0 0 2 3 】

文字列は「2枚の板 10 と 11 とは、ボルト」となり、符号は「12」、符号抽出位置は「18（段落）」「2（文）」「23（文字目）」となる。これらのデータによって、図に示すようなレコード 15 が生成される。処理対象の文書は必ずしもこういった構造でなくてよく、もし段落番号に相当するものがない場合には、第何ページの第何行目といった表現により抽出位置を表せばよい。

こうして、図 1 に示した表示部 11 には、符号とその符号を付加した単語だけでなく、その単語を含む一定文字数以下の文字列や抽出位置が表示される。

もし、この出力により、符号やその前方の文字列の表現に誤りが発見された場合には、符号抽出位置を参照して、処理対象の文書の該当箇所を調べ、正しい表現に修正する。符号抽出位置と符号の前方の文字列とは、共に、該当箇所を探す作業に効果的に利用される。前後の関係がわかるから、文字列が長いほど該当箇所が見つけやすい。

#### 【 0 0 2 4 】

図 3 には、抽出直後のデータの内容説明図を示す。

図 2 に示したような要領で、処理対象となった文書の符号を探索し、各符号について上記のようなレコードを順に生成していくと、この図 3 に示すような順番に 1 行ずつレコード

10

20

30

40

50

が並ぶ。この順は、符号の抽出順であるから、同一の符号を付加した単語であっても列方向に見たとき全くばらばらに並ぶ。これでは、同一の単語に同一の符号を付加しているかどうかを見比べるのに不便である。そこで、本発明では、符号順にこのレコードをソートして、図1に示すように整理して抽出結果を表示するようにしている。

#### 【0025】

##### 符号付けミスの発見方法

図4には、符号付けミスの例説明図を示す。

上記のような出力を得ると、この例に示すような各種の符号付けミスを容易に発見できる。

この出力は、図1や図2等で示したものとは別の文書について処理した例を示したものである。例えばここには図に示すように、符号「12」を付加した文字列が上から4レコード分表示されている。これを見ると、「ボルト12」とすべきところを「ボルト12」としたり、「ボルヨ12」としたりして、いずれもタイプミスをしていることがよく分かる。特許出願明細書を通読しながらこういったミスを探しても、注意深く字を確認していくかないとうっかり見落とす場合も多い。しかしながら、本発明のように符号とその前方の文字列とを上下に対応させて並べて表示すれば、ささいなミスも発見し易い。

#### 【0026】

また、次の4行分の出力で符号「20」の前方の文字列を比較すると、「制御部」「主制御部」「制御装置」というように、同一部分を様々な言い方で表現していることがわかる。長文の明細書になると、こうした表現の不一致を犯すことは少なくない。本人は同一物を指しているつもりでも、第3者が読むと別の物を指しているように読み取れるため、表現は統一したい。しかしながら、これも、長文明細書の場合には、点検の際に気が付きにくい。本発明によれば、これが容易に発見できる。

#### 【0027】

また、その次の4行分の出力から、符号「25」を付加した単語は「接合部」ということが分かる。しかし、「接合部24」というように、符号付けミスを犯している。これは、この出力によって一目瞭然である。また、「支柱25」という表現が並んでいることから、「支柱」という単語に符号を付け間違えたということも容易に発見できる。

ところで、本発明では符号を付加した単語それ自体を直接指定することはしない。符号の付いている単語の自体を予め全て指定する作業は大変に手数がかかり、実用的でない。

#### 【0028】

そこで、本発明では、符号の直前または直後にある文字を自動的に一定数だけ切り取って表示する。こうすれば、この文字数以下の単語は、必ず抽出した文字列中に含まれる。抽出した文字列を比較すれば、不要な文字との区別は容易である。また、同一符号を付したとしても表現の異なる単語が一緒に抽出できる。

#### 【0029】

しかも、実際に使用してみると、単語を抽出する場合には、その前後の文字も表示されていたほうが、他の部分との比較や本文中での探索等に便利であるという、予期しない効果が生じた。

即ち、符号の付け間違いや単語の表現の間違いは、同一の符号を付けた他の部分との比較による。このとき、どの表現が正しいか前後の文を参照して決めることがある。従って、前後の文が抽出されていると、本文を参照しなくても、修正対象を決定できる。また、符号と単語との間に誤って他の文字が挿入されてしまったような場合にも分かりやすい。

また、修正対象が決まった後は、本文を取り出して修正作業をするが、そのとき、修正対象を見つけ出す作業では、その単語の前後の文字が重要な手がかりになる。これにより、長い文章中から速やかに修正箇所を見つけることができる。

#### 【0030】

##### 文書の修正処理全体の流れ

この図により本発明の装置による符号抽出処理全体の流れを説明する。

図5は、本発明による文書修正処理のフローチャートである。

10

20

30

40

50

図において、先ずステップ S 1 で対象文書を指定する。具体的には、特許出願明細書を電子化して格納した文書フロッピーディスクをコンピュータにセットしてから、符号照合処理プログラムを起動して、対象文書名を入力する。次のステップ S 2 で、そのプログラムの動作を開始させる。次のステップ S 3 では、先に説明した要領で符号抽出処理を実行する。このプログラムの符号抽出処理の具体的な各ステップは、後で図 6 を用いて説明する。

#### 【 0 0 3 1 】

ステップ S 4 では、出力ファイルをオープンする。この出力ファイルは、最終的に表示部 11 に表示出力するデータを格納するファイルである。ステップ S 5 では、この出力ファイルに、符号抽出処理の結果を書き出す。さらに、ステップ S 6 では、出力ファイルをディスプレイに表示するか、プリンタで印刷するかを判断している。どちらにするかは、利用者が指定する。その選択に応じて、ステップ S 7 ではディスプレイ表示をし、ステップ S 8 ではプリント出力をする。ここで、符号照合プログラムは終了する。

10

#### 【 0 0 3 2 】

次のステップ S 9 では、出力表示された内容を利用者が見て、先に説明した要領で実際に符号付けミスを探索する。長文の明細書の場合には、かなり大量の出力が表示されるが、前後の符号や文字列中に含まれる単語が一致しているかどうかを見比べていく作業は、比較的スピーディに進む。

こうして発見したミスはメモしたり、またプリント出力された用紙中にマークをする。

そして、処理対象となった文書をワードプロセッサ等により呼び出して、該当箇所の修正をおこなう。

20

#### 【 0 0 3 3 】

##### 符号抽出処理

図 6 には、本発明の装置の符号抽出処理動作フローチャートを示す。

これは、図 5 のステップ S 3 の処理を具体化したものである。これが、本発明の符号照合プログラムの最も特徴的な処理を行う部分である。

先ずステップ S 1 において、抽出対象符号の指定を行う。抽出対象符号は例えば数字、アルファベット、ハイフンといったものである。これは、利用者がプログラム実行のつど指定してもよいが、予めプログラム内で指定しておき、必要に応じて利用者が選択するようにもよい。通常は同一利用者の使用する符号の種類はほぼ一定だからである。

30

#### 【 0 0 3 4 】

次のステップ S 2 において、図 5 のステップ S 1 で指定された処理対象文書がオープンされる。さらにステップ S 3 で出力バッファ領域が指定される。先に説明したように、処理対象文書の符号を探索して生成した多数のレコードは、後からまとめてソートされる。そこで、ソート処理まで、生成したレコードを一時保存するために、この出力バッファ領域が確保される。

#### 【 0 0 3 5 】

ステップ S 4 では、処理対象文書から 1 文を読み、ステップ S 5 で符号検出が行われる。テキストデータでは、文書を構成する各文は改行により区分されている。この文を 1 文ずつ読み取り、その先頭の文字から順に 1 文字ずつ、指定された符号に該当するかどうかを判断する。符号が検出されると、その符号をメモリに保持して、次の文字が符号かどうかを調べる。符号に該当する文字が連続していれば連続する全ての文字をひとかたまりの符号と判断する。従って、符号でない文字が検出されるまで、この処理を繰り返し、符号でない文字が検出されるとこれまでメモリに保持していたいくつかの文字を符号として抽出する（ステップ S 6）。ステップ S 7 では、符号の位置データを抽出する。これは、例えば符号を構成する幾つかの文字のうちの先頭の文字の位置とする。

40

#### 【 0 0 3 6 】

次に、ステップ S 8 で、符号の前方あるいは後方の文字列をバッファに格納できる分だけ格納する。どの文字列を何文字抽出するかは、別途指定しておく。

ステップ S 9 は、1 文について全ての文字に対して符号探索処理を実行したかどうかを判

50

断して、ステップ S 5 からステップ S 8 のループ処理を繰り返す制御を行う。

ステップ S 10 では、処理対象の文書の全文について、この符号探索処理を終了したかどうかを判断してステップ S 4 からステップ S 9 のループ処理を繰り返す制御を行う。

#### 【0037】

全文について、符号探索処理が終了すると、ステップ S 11 でソートキーが指定される。このとき、どのキーを主キーとし、どのキーを従キー即ち主キーに準じてソートするキーとするか等が指定される。この指定はそのつど行ってもよいが、プログラムで自動設定しておることが好ましい。同一の利用者が希望する出力形式は通常一定だからである。

その後、ステップ S 12 で、出力バッファの内容が、指定されたキーについてソートされる。こうして、符号抽出処理の結果が図 5 のステップ S 5 でオープンしたファイルに書き込まれる。

#### 【0038】

##### ソートキー

図 7 は、文字列をソートキーに含めた効果の説明図を示す。

これまで説明したように本発明を利用するには、主たるソートキーは符号となるが、この他に、例えば符号の前方の文字列のうち符号に最も近い文字 1 文字を従キーに指定すると、この図に示すような効果が得られる。

この図の (a) と (b) には、対象文書中の符号「2」を付加した単語を含むいくつかの文字列が表示されている。(b) の各レコードは、符号を主キーとし、その直前の 1 文字を従キーとしソートされている。従って、符号のみに着目してソートした (a) の出力と比較すると、同一の語尾の単語が互いに集中して並ぶように表示され、相互に文字列を比較し易い配置となっている。

なお、2 文字以上を従キーとすればさらに見やすくなるが、ソート時間が長くなる恐れがあり、また、実用的にも、数文字で十分という結果が得られている。

#### 【0039】

##### 異種の文字を含む符号

図 8 には、異種の文字から成る符号のソート方法説明図を示す。

例えば図に示すように、「1 A」とか「1 2 A」というように、数字と文字とを組み合わせた符号を使用することも多い。このような符号を抽出してそのままソートすると、例えば図の (a) に示すようになる。

#### 【0040】

即ち、符号は数値データではなく文字データであるから、通常この図に示すようにバッファに左詰めで格納される。従ってそのままソートすると、この (a) の右側に示すように、「1 2 A」が「1 A」の前になるように並べられて出力される。しかし、一般には、符号を数字順に使用し、文字はその数字の添え字のように補助的に使用されるから、ソート結果も数字順に並ぶことが好ましい。そこで、本発明の装置は、図の (b) に示すように、いったん抽出した符号をバッファ内でその文字種の境界が揃うように位置合わせする。こうして位置合わせをした後ソートすると、図の (b) の右側に示したように、数字順でかつ英字順に符号が並ぶ。また、文字種の境界が揃うように位置合わせしたので、符号同士の比較も容易である。

#### 【0041】

(c) の例は、「1 1」「3」といった符号の他に、「1 - 1」といった符号を混ぜて使用した例を示す。この場合にも「-」の右側の数字は添え字的な使い方であり、「-」の左側の数字順にソートすることが好ましい。そこで、図の (d) に示すように位置合わせをしてからソートし、左側の数字を主キーとし、その他の文字を従キーとなるようにした。

これにより、文字列と符号とを比較し易い出力が得られる。

#### 【0042】

##### 符号抽出条件

図 9 には、抽出条件の説明図を示す。

10

20

30

40

50

本発明によれば、文字列や符号の抽出条件を選択することにより様々な利用形態が考えられる。符号を「数字」とか「英字」というように指定すると、極めて簡潔に一括指定ができる。一方、例えば、全ての符号でなく、そのうちの特定の符号のみを指定して処理をおこなうと、特定の符号の前や後の文字列だけを文中から抽出して比較し、文章の点検等を行うことができる。

#### 【0043】

また、符号の前方の文字列を抽出して表示するか後方の文字列を抽出して表示するか、あるいは両方を抽出して表示するか、さらには、何文字を表示するか等を指定できるようにしておけば、必要以上の文字列表示を抑制したりして、より見やすい出力を得ることができる。

10

また逆に、符号の前後の文字列をできるだけ沢山表示したり、少し離れた部分の文字列を表示させて、「1は制御装置である。」とか「3は車軸を支える軸受けである。」といった表現の点検も可能にすることができる。

#### 【0044】

このほか、処理対象の文書点検後も、その出力をを利用して、文中で使用されている符号の種類や単語を確認すれば、その文書に新たな文を追加するような場合に、符号付けミスを防止できる。即ち、新しい単語に対しては重複しない単語を選択し、すでに使用されている単語に対しては同一の表現で同一の単語を間違いなく使用できる。

自分の書いた文書はもとより、他人の書いた文書についてこのような加筆を行う作業ではその効果が顕著である。

20

また、本発明は、特許出願明細書のみならず、符号を単語に付加して記述を行う各種の文書の点検作業等に広く利用することができる。

#### 【図面の簡単な説明】

【図1】本発明の符号照合装置の具体例を示すブロック図である。

【図2】処理対象文書の構造説明図である。

【図3】抽出直後のデータの内容説明図である。

【図4】符号付けミスの例説明図である。

【図5】本発明による文書修正処理のフローチャートである。

【図6】本発明の装置の符号抽出処理動作フローチャートである。

【図7】文字列をソートキーに含めた効果の説明図である。

30

【図8】異種の文字から成る符号のソート方法説明図である。

【図9】抽出条件の説明図である。

#### 【符号の説明】

1 処理対象となる文書

3 符号抽出部

4 隣接文字列抽出部

5 バッファメモリ

6 ソート部

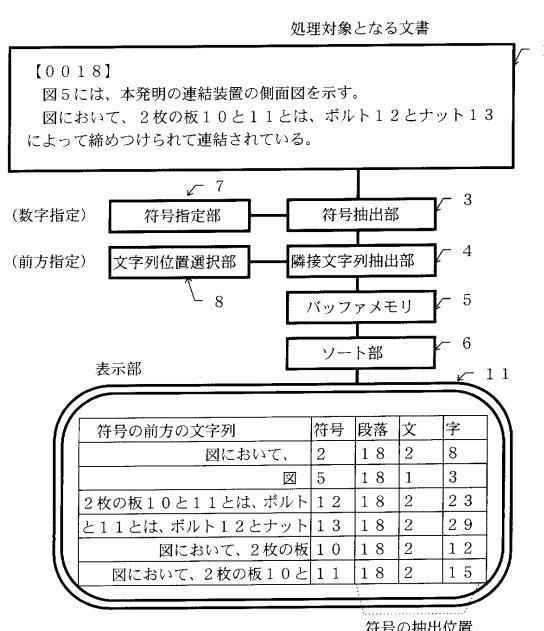
7 符号指定部

8 文字列位置選択部

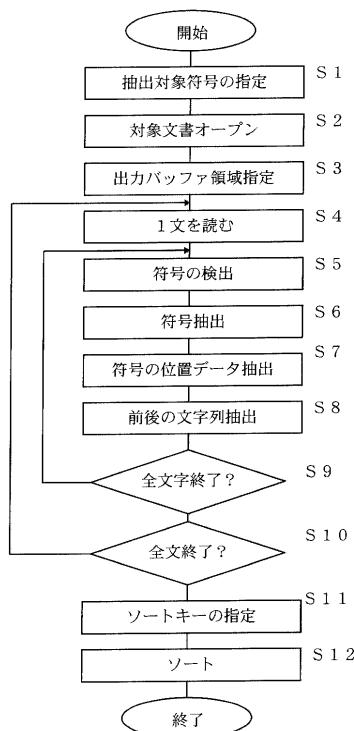
1 1 表示部

40

【図1】



【図6】



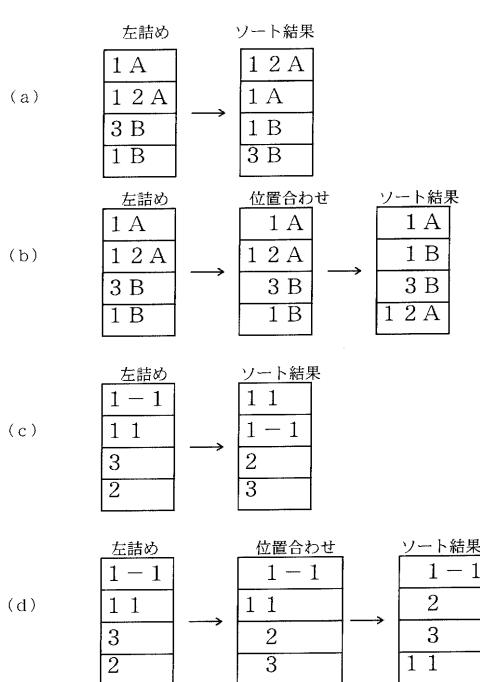
本発明の装置の符号抽出処理

【図7】



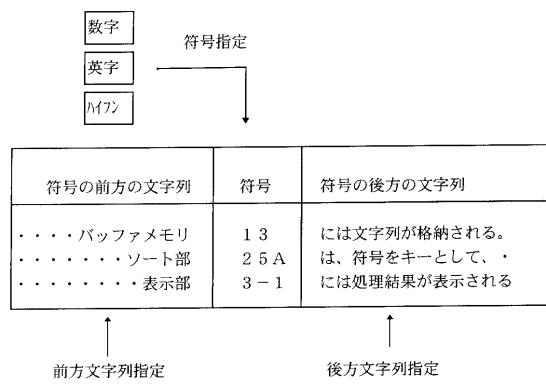
文字列をソートキーに含めた効果の説明図

【図8】



異種の文字から成る符号のソート方法説明図

【図9】



抽出条件の説明図

---

フロントページの続き

(56)参考文献 特開昭63-163956(JP,A)

特開平1-205263(JP,A)

特開平4-296970(JP,A)

倉田昌典、菅沼明、牛島和夫, 日本語文章推敲支援ツール『推敲』のパーソナルコンピュータでの実用化, 日本ソフトウェア科学会第4回大会論文集, 1987年, 51-54頁, B-1-2