

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7601063号
(P7601063)

(45)発行日 令和6年12月17日(2024.12.17)

(24)登録日 令和6年12月9日(2024.12.9)

(51)国際特許分類

F I

G 0 6 F 3/0481(2022.01)

G 0 6 F 3/0481

G 0 6 F 3/0482(2013.01)

G 0 6 F 3/0482

G 0 6 F 3/02 (2006.01)

G 0 6 F 3/02 5 3 0

G 0 6 F 15/02 (2006.01)

G 0 6 F 15/02 3 1 5 S

請求項の数 5 (全22頁)

(21)出願番号	特願2022-109378(P2022-109378)	(73)特許権者	000001443
(22)出願日	令和4年7月7日(2022.7.7)		カシオ計算機株式会社
(65)公開番号	特開2024-7937(P2024-7937A)		東京都渋谷区本町 1 丁目 6 番 2 号
(43)公開日	令和6年1月19日(2024.1.19)	(74)代理人	110001254
審査請求日	令和5年6月2日(2023.6.2)		弁理士法人光陽国際特許事務所
前置審査		(72)発明者	北村 健登
			東京都羽村市栄町 3 丁目 2 番 1 号 カシ
			オ計算機株式会社 羽村技術センター内
		審査官	宮本 昭彦

最終頁に続く

(54)【発明の名称】 入力支援装置、入力支援方法及びプログラム

(57)【特許請求の範囲】

【請求項 1】

入力された数式を対象にして数式のエラー箇所を検出する検出手段と、
前記検出手段によりエラー箇所が検出された場合に、入力位置を示すカーソルを前記エ
ラー箇所に対応する位置に表示させるとともに前記エラー箇所に対応する修正候補を前記
数式とは異なる行に表示させ、且つ、前記数式における前記エラー箇所の表示コントラ
ストを維持したまま前記数式における他の箇所の表示コントラストを低下させる表示制御
手段と、
を備え、
前記表示制御手段は、
前記数式の背景の表示濃度と前記エラー箇所での数式の表示濃度とを変化させずに前記他
の箇所での数式の表示濃度を前記数式の背景の表示濃度に近づけることにより、前記エラ
ー箇所の表示コントラストを維持したまま前記他の箇所の表示コントラストを低下させ、
且つ、前記カーソルを前記エラー箇所に対応する位置に表示させている間、前記修正候補
が非選択状態として識別可能なように、前記修正候補の表示コントラストを前記数式にお
ける前記他の箇所の表示コントラストよりも更に低下させた状態で表示させる、
ことを特徴とする入力支援装置。

【請求項 2】

前記表示制御手段は、前記カーソルを前記エラー箇所に対応する位置に表示させる場合
には、表示コントラストを低下させた前記他の箇所よりも表示コントラストが高くなるよ

うに前記カーソルを表示させる、

ことを特徴とする請求項 1 に記載の入力支援装置。

【請求項 3】

前記検出手段は、前記数式の入力中に、当該数式のエラー箇所を検出する、

ことを特徴とする請求項 1 又は 2 に記載の入力支援装置。

【請求項 4】

入力支援装置が実行する入力支援方法であって、

入力された数式を対象にして数式のエラー箇所を検出する検出工程と、

前記検出工程においてエラー箇所が検出された場合に、入力位置を示すカーソルを前記エラー箇所に対応する位置に表示させるとともに前記エラー箇所に対応する修正候補を前記数式とは異なる行に表示させ、且つ、前記数式における前記エラー箇所の表示コントラストを維持したままで前記数式における他の箇所の表示コントラストを低下させる表示制御工程と、

を含み、

前記表示制御工程は、

前記数式の背景の表示濃度と前記エラー箇所での数式の表示濃度とを変化させずに前記他の箇所での数式の表示濃度を前記数式の背景の表示濃度に近づけることにより、前記エラー箇所の表示コントラストを維持したままで前記他の箇所の表示コントラストを低下させ、且つ、前記カーソルを前記エラー箇所に対応する位置に表示させている間、前記修正候補が非選択状態として識別可能なように、前記修正候補の表示コントラストを前記数式における前記他の箇所の表示コントラストよりも更に低下させた状態で表示させる、

ことを特徴とする入力支援方法。

【請求項 5】

コンピュータを、

入力された数式を対象にして数式のエラー箇所を検出する検出手段、

前記検出手段によりエラー箇所が検出された場合に、入力位置を示すカーソルを前記エラー箇所に対応する位置に表示させるとともに前記エラー箇所に対応する修正候補を前記数式とは異なる行に表示させ、且つ、前記数式における前記エラー箇所の表示コントラストを維持したままで前記数式における他の箇所の表示コントラストを低下させる表示制御手段、

として機能させ、

前記表示制御手段は、

前記数式の背景の表示濃度と前記エラー箇所での数式の表示濃度とを変化させずに前記他の箇所での数式の表示濃度を前記数式の背景の表示濃度に近づけることにより、前記エラー箇所の表示コントラストを維持したままで前記他の箇所の表示コントラストを低下させ、且つ、前記カーソルを前記エラー箇所に対応する位置に表示させている間、前記修正候補が非選択状態として識別可能なように、前記修正候補の表示コントラストを前記数式における前記他の箇所の表示コントラストよりも更に低下させた状態で表示させる、

ことを特徴とするプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、入力支援装置、入力支援方法及びプログラムに関する。

【背景技術】

【0002】

従来、数式の入力を受け付けて演算する関数電卓が知られている。関数電卓は、入力した数式に構文エラーがある場合に、エラー箇所を表示するものが知られている。例えば、入力された数式のエラー箇所を通常表示からアンダーライン及び太字の強調表示に変更し、エラー箇所以外の箇所を通常表示する電子計算機が知られている（特許文献 1 参照）。

【先行技術文献】

【特許文献】

【0003】

【文献】特開2011-107772号公報

【発明の概要】

【発明が解決しようとする課題】

【0004】

しかし、特許文献1に記載の電子計算機では、強調表示する情報が表示画面内で混み合い、逆にエラー箇所が見つづらくなり、直観的にも分かりづらくなるおそれがあった。

【0005】

本発明の課題は、エラー箇所を見やすくしつつ、直観的に分かりやすく表示することである。

10

【課題を解決するための手段】

【0006】

上記課題を解決するために、本発明に係る入力支援装置は、入力された数式を対象にして数式のエラー箇所を検出する検出手段と、前記検出手段によりエラー箇所が検出された場合に、入力位置を示すカーソルを前記エラー箇所に対応する位置に表示させるとともに前記エラー箇所に対応する修正候補を前記数式とは異なる行に表示させ、且つ、前記数式における前記エラー箇所の表示コントラストを維持したままで前記数式における他の箇所の表示コントラストを低下させる表示制御手段と、を備え、前記表示制御手段は、前記数式の背景の表示濃度と前記エラー箇所での数式の表示濃度とを変化させずに前記他の箇所での数式の表示濃度を前記数式の背景の表示濃度に近づけることにより、前記エラー箇所の表示コントラストを維持したままで前記他の箇所の表示コントラストを低下させ、且つ、前記カーソルを前記エラー箇所に対応する位置に表示させている間、前記修正候補が非選択状態として識別可能なように、前記修正候補の表示コントラストを前記数式における前記他の箇所の表示コントラストよりも更に低下させた状態で表示させる、ことを特徴とする。

20

また、本発明に係る入力支援方法は、入力支援装置が実行する入力支援方法であって、入力された数式を対象にして数式のエラー箇所を検出する検出工程と、前記検出工程においてエラー箇所が検出された場合に、入力位置を示すカーソルを前記エラー箇所に対応する位置に表示させるとともに前記エラー箇所に対応する修正候補を前記数式とは異なる行に表示させ、且つ、前記数式における前記エラー箇所の表示コントラストを維持したままで前記数式における他の箇所の表示コントラストを低下させる表示制御工程と、を含み、前記表示制御工程は、前記数式の背景の表示濃度と前記エラー箇所での数式の表示濃度とを変化させずに前記他の箇所での数式の表示濃度を前記数式の背景の表示濃度に近づけることにより、前記エラー箇所の表示コントラストを維持したままで前記他の箇所の表示コントラストを低下させ、且つ、前記カーソルを前記エラー箇所に対応する位置に表示させている間、前記修正候補が非選択状態として識別可能なように、前記修正候補の表示コントラストを前記数式における前記他の箇所の表示コントラストよりも更に低下させた状態で表示させる、ことを特徴とする。

30

また、本発明に係るプログラムは、コンピュータを、入力された数式を対象にして数式のエラー箇所を検出する検出手段、前記検出手段によりエラー箇所が検出された場合に、入力位置を示すカーソルを前記エラー箇所に対応する位置に表示させるとともに前記エラー箇所に対応する修正候補を前記数式とは異なる行に表示させ、且つ、前記数式における前記エラー箇所の表示コントラストを維持したままで前記数式における他の箇所の表示コントラストを低下させる表示制御手段、として機能させ、前記表示制御手段は、前記数式の背景の表示濃度と前記エラー箇所での数式の表示濃度とを変化させずに前記他の箇所での数式の表示濃度を前記数式の背景の表示濃度に近づけることにより、前記エラー箇所の表示コントラストを維持したままで前記他の箇所の表示コントラストを低下させ、且つ、前記カーソルを前記エラー箇所に対応する位置に表示させている間、前記修正候補が非選択状態として識別可能なように、前記修正候補の表示コントラストを前記数式における前

40

50

記他の箇所の表示コントラストよりも更に低下させた状態で表示させる、ことを特徴とする。

【発明の効果】

【0007】

本発明によれば、エラー箇所を見やすくできつつ、直観的に分かりやすく表示できる。

【図面の簡単な説明】

【0008】

【図1】本発明の実施の形態の電子卓上計算機の正面図である。

【図2】電子卓上計算機の機能構成を示すブロック図である。

【図3】第1の演算表示処理を示すフローチャートである。

10

【図4】(a)は、実施の形態の第1の具体例の表示画面を示す図である。(b)は、実施の形態の第1の具体例の表示画面を示す図である。(c)は、実施の形態の第1の具体例の表示画面を示す図である。(d)は、実施の形態の第1の具体例の表示画面を示す図である。(e)は、実施の形態の第1の具体例の表示画面を示す図である。

【図5】(a)は、実施の形態の第2の具体例の表示画面を示す図である。(b)は、実施の形態の第2の具体例の表示画面を示す図である。(c)は、実施の形態の第2の具体例の表示画面を示す図である。(d)は、実施の形態の第2の具体例の表示画面を示す図である。(e)は、実施の形態の第2の具体例の表示画面を示す図である。

【図6】第2の演算表示処理を示すフローチャートである。

【図7】(a)は、第1の変形例の表示画面を示す図である。(b)は、第1の変形例の表示画面を示す図である。(c)は、第1の変形例の表示画面を示す図である。(d)は、第1の変形例の表示画面を示す図である。(e)は、第1の変形例の表示画面を示す図である。

20

【図8】(a)は、第1の変形例の表示画面を示す図である。(b)は、第1の変形例の表示画面を示す図である。(c)は、第1の変形例の表示画面を示す図である。

【図9】第3の演算表示処理を示すフローチャートである。

【図10】(a)は、第2の変形例の第1の具体例の表示画面を示す図である。(b)は、第2の変形例の第1の具体例の表示画面を示す図である。(c)は、第2の変形例の第1の具体例の表示画面を示す図である。(d)は、第2の変形例の第1の具体例の表示画面を示す図である。(e)は、第2の変形例の第1の具体例の表示画面を示す図である。(f)は、第2の変形例の第1の具体例の表示画面を示す図である。

30

【図11】(a)は、第2の変形例の第2の具体例の表示画面を示す図である。(b)は、第2の変形例の第2の具体例の表示画面を示す図である。(c)は、第2の変形例の第2の具体例の表示画面を示す図である。(d)は、第2の変形例の第2の具体例の表示画面を示す図である。

【発明を実施するための形態】

【0009】

以下、添付図面を参照して本発明に係る実施の形態及び第1、第2の変形例を順に詳細に説明する。なお、以下に述べる実施の形態及び第1、第2の変形例には、本発明を実施するために技術的に好ましい種々の限定が付されているが、本発明の範囲を以下の実施の形態、第1、第2の変形例及び図示例に限定するものではない。

40

【0010】

(実施の形態)

図1～図5(e)を参照して、本発明に係る実施の形態を説明する。まず、図1及び図2を参照して、本実施の形態の装置構成を説明する。図1は、本実施の形態の電子卓上計算機1の正面図である。図2は、電子卓上計算機1の機能構成を示すブロック図である。

【0011】

本実施の形態の入力支援装置としての電子卓上計算機1は、いわゆる関数電卓であるものとする。図1に示すように、電子卓上計算機1は、各種キー群を有する操作部12と、表示部14と、を備える。

50

【 0 0 1 2 】

操作部 1 2 は、ユーザから数値や計算記号などの数式構成要素の入力操作を受けたり、各種処理の指示操作を受けたりするためのキー群であり、それぞれ固有の機能を割り当てられた複数のキーを備える。本実施の形態において、操作部 1 2 は、テンキー 2 0 a、演算キー 2 0 b、修正キー 2 0 c、関数キー 2 0 d、上矢印キー 2 1、下矢印キー 2 2、右矢印キー 2 3、左矢印キー 2 4、決定キー 2 5、「=」キー 2 6、各種の機能キー 2 0 e などを含む。

【 0 0 1 3 】

テンキー 2 0 a は、数値の入力を受け付けるキーであり、「0」キー～「9」キーを有する。演算キー 2 0 b は、演算子の入力を受け付けるキーであり、「+」キー、「-」キー、「×」キー、「÷」キーを有する。修正キー 2 0 c は、表示中の数値、数式の修正の入力を受け付けるキーであり、1文字の削除の入力を受け付ける D E L (Delete) キー、全削除の入力を受け付ける A C (All Clear) キーを有する。関数キー 2 0 d は、各種の関数の入力を受け付けるキーである。

【 0 0 1 4 】

また、上矢印キー 2 1、下矢印キー 2 2、右矢印キー 2 3 及び左矢印キー 2 4 は、表示部 1 4 内で編集対象位置や選択対象位置や、それらの位置を示すカーソルを所定の各方向に移動させる場合などに押下されるキーであり、これらのキーを総称してカーソルキーと呼ぶ。

【 0 0 1 5 】

決定キー 2 5 は、表示部 1 4 に表示された表示情報に関する選択、遷移などの各種処理の決定 (OK) の入力を受け付けるキーである。「=」キー 2 6 は、表示部 1 4 に表示された数式の等号の入力を受け付けるキーであり、数式の演算の実行指示 (演算指示) の入力を受け付ける実行キーとして機能する。

【 0 0 1 6 】

機能キー 2 0 e は、各種機能の入力を受け付けるキーであり、シフトキー、メニューキー、電源キーなどを含む。

【 0 0 1 7 】

表示部 1 4 は、ドットマトリクス方式の L C D (Liquid Crystal Display) などの表示パネルにより構成されており、操作部 1 2 などの操作に応じた文字や符号、記号、数式、演算結果などの各種データを背景の白地上に表示する表示部である。本実施の形態において、表示部 1 4 は、表示態様として、白地に、黒、ダークグレー、ライトグレー、白 (表示なし) の 4 つの階調 (表示濃度、グレースケールの色) での情報の表示が可能である。以下、「背景の白地に黒」を単に「黒」と表現し、「背景の白地にダークグレー」を単に「ダークグレー」と表現し、「背景の白地にライトグレー」を単に「ライトグレー」と表現するものとする。なお、表示部 1 4 の表示態様の階調数は、3、又は 5 以上としてもよい。

【 0 0 1 8 】

表示部 1 4 における表示態様 (4 つの階調) について、情報 (文字) の表示濃度が薄くなる (黒→白) ほど、より抑制的な表示態様であるものとして表現する。つまり、背景の白地に対する文字の表示色のコントラスト (表示濃度の差分) が低くなるほど、当該文字の表示態様 (コントラスト) が、より抑制的であるものとして表現する。なお、操作部 1 2 は、表示部 1 4 の表示パネル上に一体的に設けられて、タッチ入力を受け付けるタッチパネルを含む構成としてもよい。

【 0 0 1 9 】

ついで、図 2 を参照して、電子卓上計算機 1 の内部の機能構成を説明する。図 2 に示すように、電子卓上計算機 1 は、検出手段、表示制御手段としての C P U (Central Processing Unit) 1 1 と、操作部 1 2 と、R A M (Random Access Memory) 1 3 と、表示部 1 4 と、記憶部 1 5 と、を備える。電子卓上計算機 1 の各部は、バス 1 6 を介して接続されている。

10

20

30

40

50

【 0 0 2 0 】

C P U 1 1 は、電子卓上計算機 1 の各部を制御する。C P U 1 1 は、記憶部 1 5 に記憶された各種プログラムのうち指定されたプログラムを読み出して R A M 1 3 に展開し、展開されたプログラムとの協働で、各種処理を実行する。

【 0 0 2 1 】

操作部 1 2 は、テンキー 2 0 a、演算キー 2 0 b、修正キー 2 0 c、関数キー 2 0 d、カーソルキー（上矢印キー 2 1、下矢印キー 2 2、右矢印キー 2 3、左矢印キー 2 4）、決定キー 2 5、「 = 」キー 2 6、各種の機能キー 2 0 e などの複数の各種キーを有し、各種キーを介するユーザからの押下の操作入力を受け付け、その操作情報を C P U 1 1 に出力する。

10

【 0 0 2 2 】

R A M 1 3 は、情報を読み出し及び書き込み可能な揮発性の半導体メモリであり、C P U 1 1 に作業用のワークエリアを提供し、データ及びプログラムを一時的に記憶する。

【 0 0 2 3 】

表示部 1 4 は、C P U 1 1 などから入力される各種の表示情報を、4 つの階調の表示態様で表示パネルの表示画面に表示する。

【 0 0 2 4 】

記憶部 1 5 は、R O M (Read Only Memory) などにより構成され、情報の読み出しが可能な記憶部であり、各種データ及び各種プログラムを記憶している。特に、記憶部 1 5 は、後述する第 1 の演算表示処理を実行するための第 1 の演算表示プログラム P 1 を記憶する。

20

【 0 0 2 5 】

つぎに、図 3 ~ 図 5 (e) を参照して、電子卓上計算機 1 の動作を説明する。図 3 は、第 1 の演算表示処理を示すフローチャートである。図 4 (a) は、第 1 の具体例の表示画面 3 0 a を示す図である。図 4 (b) は、第 1 の具体例の表示画面 3 0 b を示す図である。図 4 (c) は、第 1 の具体例の表示画面 3 0 c を示す図である。図 4 (d) は、第 1 の具体例の表示画面 3 0 d を示す図である。図 4 (e) は、第 1 の具体例の表示画面 3 0 e を示す図である。図 5 (a) は、第 2 の具体例の表示画面 4 0 a を示す図である。図 5 (b) は、第 2 の具体例の表示画面 4 0 b を示す図である。図 5 (c) は、第 2 の具体例の表示画面 4 0 c を示す図である。図 5 (d) は、第 2 の具体例の表示画面 4 0 d を示す図である。図 5 (e) は、第 2 の具体例の表示画面 4 0 e を示す図である。

30

【 0 0 2 6 】

電子卓上計算機 1 において、例えば、操作部 1 2 の機能キー 2 0 e の電源キーが入力されて電源オンされたことをトリガとして、C P U 1 1 は、第 1 の演算表示プログラム P 1 を記憶部 1 5 から読み出し、読み出した第 1 の演算表示プログラム P 1 に従い、第 1 の演算表示処理を実行する。

【 0 0 2 7 】

まず、C P U 1 1 は、操作部 1 2 を介して、ユーザからの数式の入力を受け付け、入力された数式を表示部 1 4 に黒で表示する（ステップ S 1 1）。ステップ S 1 1 において、数式入力の表示画面では、入力箇所に黒のカーソルが点滅表示されているものとする。そして、C P U 1 1 は、数式の入力の終わりでかつ演算指示を意味する「 = 」キー 2 6 が押下入力されたか否かを判別する（ステップ S 1 2）。「 = 」キー 2 6 が入力されていない場合（ステップ S 1 2 ; N O）、ステップ S 1 1 に移行される。

40

【 0 0 2 8 】

「 = 」キー 2 6 が入力された場合（ステップ S 1 2 ; Y E S）、C P U 1 1 は、表示中の数式に構文エラーがあるか否かを判別する（ステップ S 1 3）。構文エラーは、数式の構文のエラーであり、演算子の連続入力（例えば、「 ~ ÷ × ~ 」）のエラーなどである。厳密には、数式のエラーの種類は、構文エラー以外にも、演算結果が演算範囲を超える又は数学的な誤り（0 で除算するなど）の計算エラーなどの他の種類のエラーもあるが、本実施の形態では、説明を簡単にするため、エラーとして構文エラーのみの判別を考えるも

50

のとする。

【 0 0 2 9 】

構文エラーがある場合（ステップ S 1 3 ; Y E S ）、C P U 1 1 は、数式に構文エラーがある旨の「 S y n t a x E R R O R 」と、数式表示に戻るための入力を受け付けることを示す「 B a c k 」(例えば、反転表示)とを、黒で表示部 1 4 に表示する(ステップ S 1 4)。B a c k の入力は、決定キー 2 5 又は「 = 」キー 2 6 の押下入力により受け付けるものとする。

【 0 0 3 0 】

そして、C P U 1 1 は、決定キー 2 5 又は「 = 」キー 2 6 を介して、ユーザから数式表示へ戻る旨の入力を受け付ける(ステップ S 1 5)。そして、C P U 1 1 は、入力された数式のうち、ステップ S 1 3 で構文エラーと判別されたエラー箇所を黒で表示部 1 4 に表示し、エラー箇所以外の箇所を黒よりも抑制的なダークグレーで表示部 1 4 に表示する(ステップ S 1 6)。数式内のエラー箇所は、1 つの場合だけでなく、複数の場合もあり得る。

10

【 0 0 3 1 】

そして、C P U 1 1 は、1 番目のエラー箇所(数式内のエラー箇所が 1 つの場合には、その 1 つのエラー箇所)にカーソル「 | 」を移動して黒で点滅表示する(ステップ S 1 7)。そして、C P U 1 1 は、操作部 1 2 を介して、表示中の数式の修正入力を受け付ける(ステップ S 1 8)。ステップ S 1 8 では、例えば、カーソルキー(上矢印キー 2 1 ~ 左矢印キー 2 4)の入力により、カーソルの移動入力を受け付け、修正キー 2 0 c の D E L キーの入力によりエラー箇所の不要箇所の削除入力を受け付け、演算キー 2 0 b などの入力により必要な演算子などの入力を受け付ける。

20

【 0 0 3 2 】

そして、C P U 1 1 は、「 = 」キー 2 6 が入力されたか否かを判別する(ステップ S 1 9)。「 = 」キー 2 6 が入力されていない場合(ステップ S 1 9 ; N O)、ステップ S 1 8 に移行される。「 = 」キー 2 6 が入力された場合(ステップ S 1 9 ; Y E S)、ステップ S 1 3 に移行される。

【 0 0 3 3 】

構文エラーがない場合(ステップ S 1 3 ; N O)、C P U 1 1 は、表示中の数式の演算を実行し、数式及び演算結果を黒で表示部 1 4 に表示し(ステップ S 2 0)、第 1 の演算表示処理を終了する。

30

【 0 0 3 4 】

ここで、図 4 (a) ~ 図 6 (e)を参照して、第 1 の演算表示処理の 2 つの具体例を説明する。第 1 の演算表示処理の第 1 の具体例として、第 1 の演算表示処理の開始後、図 4 (a)に示すように、ステップ S 1 1 で例えば、数式「 6 ÷ × 6 」が入力され、表示画面 3 0 a が表示部 1 4 に表示された場合を考える。表示画面 3 0 a は、黒の数式「 6 ÷ × 6 」を有する。この数式のうち、「 ÷ × 」は、演算子が連続しており、構文エラーのエラー箇所である。

【 0 0 3 5 】

そして、ステップ S 1 2 で、「 = 」キー 2 6 が押下入力されると、ステップ S 1 3 で、表示中の数式に構文エラーがあると判別され、図 4 (b)に示すように、ステップ S 1 4 で表示画面 3 0 b が表示される。表示画面 3 0 b は、黒の「 S y n t a x E R R O R 」と、黒の反転表示の「 B a c k 」とを有する。

40

【 0 0 3 6 】

そして、ステップ S 1 5 で、決定キー 2 5 又は「 = 」キー 2 6 が押下入力されると、図 4 (c)に示すように、ステップ S 1 6 , S 1 7 で表示画面 3 0 c が表示される。表示画面 3 0 c は、ダークグレーの「 6 」と、黒のエラー箇所の「 ÷ × 」と、ダークグレーの「 6 」とを、一連の数式として有する。また、表示画面 3 0 c において、黒のカーソル「 | 」が、エラー箇所「 ÷ × 」の中間位置に移動されて表示されている。このため、ステップ S 1 8 における修正するエラー箇所へのカーソルの大幅な移動が避けられ、エラー箇所の

50

修正が容易となる。

【0037】

そして、ステップS18で、ユーザからの右矢印キー23及び修正キー20cのDELキーの押下入力により、エラー箇所「÷×」のうちの「×」の削除の修正入力となされ、図4(d)に示すように、表示画面30dが表示される。表示画面30dは、ダークグレーの「6」と、黒の修正後のエラー箇所の「÷」と、ダークグレーの「6」とを、一連の数式として有する。そして、ステップS19で、「=」キー26が押下入力されると、ステップS13に移行され、表示中の数式に構文エラーがないと判別され、図4(e)に示すように、ステップS20で、演算結果の表示画面30eが表示部14に表示される。表示画面30eは、黒の数式「6÷6」と、黒の演算結果「1」と、を有する。

10

【0038】

ついで、第1の演算表示処理の第2の具体例として、図5(a)に示すように、ステップS11で例えば、数式「6÷×6+9×÷3」が入力され、表示画面40aが表示部14に表示された場合を考える。表示画面40aは、黒の数式「6÷×6+9×÷3」を有する。この数式のうち、「÷×」、「×÷」の2か所は、演算子が連続しており、構文エラーのエラー箇所である。

【0039】

そして、ステップS12で、「=」キー26が押下入力されると、ステップS13で、表示中の数式に構文エラーがあると判別され、図5(b)に示すように、ステップS14で表示画面40bが表示される。表示画面40bは、黒の「Syntax ERROR」と、黒の反転表示の「Back」とを有する。そして、ステップS15で、決定キー25又は「=」キー26が押下入力されると、図5(c)に示すように、ステップS16、S17で表示画面40cが表示される。表示画面40cは、ダークグレーの「6」と、黒のエラー箇所「÷×」と、ダークグレーの「6+9」と、黒のエラー箇所「×÷」と、ダークグレーの「3」とを、一連の数式として有する。また、表示画面40cにおいて、黒のカーソル「|」が、1番目のエラー箇所「÷×」の中間位置に移動されて表示されている。

20

【0040】

そして、ステップS18で、ユーザからの右矢印キー23及び修正キー20cのDELキーの押下入力により、1番目のエラー箇所「÷×」のうちの「×」の削除の修正入力となされる。さらに、図5(d)に示すように、ステップS18で、ユーザからの右矢印キー23及び修正キー20cのDELキーの押下入力により、2番目のエラー箇所の「×÷」のうちの「÷」の削除の修正入力となされ、表示画面40dが表示される。表示画面40dは、ダークグレーの「6」と、黒の修正後のエラー箇所「÷」と、ダークグレーの「6+9」と、黒の修正後のエラー箇所「×」と、ダークグレーの「3」とを、一連の数式として有する。

30

【0041】

そして、ステップS19で、「=」キー26が押下入力されると、ステップS13に移行され、表示中の数式に構文エラーがないと判別され、図5(e)に示すように、ステップS20で、演算結果の表示画面40eが、表示部14に表示される。表示画面40eは、黒の数式「6÷6+9×3」と、黒の演算結果「28」と、を有する。

40

【0042】

以上、本実施の形態によれば、電子卓上計算機1は、入力された数式(例えば、「6÷×6」)を対象にして数式のエラー箇所を検出し、エラー箇所(例えば、「÷×」)が検出された場合に、例えば表示部14に表示された表示画面30bに示すように、エラー箇所の表示態様を維持(黒に維持)したままで数式における他の箇所(例えば、「6」、「6」)の表示態様を抑制的な表示態様(ダークグレー)に切り替えるCPU11を備える。

【0043】

このため、エラー箇所の表示態様が維持されるので、エラー箇所を太字及び下線に変更して表示情報が混み合う構成よりも、エラー箇所を見やすくできつつ、さらに他の箇所の表示態様をエラー箇所よりも抑制的な表示態様に切り替えるので、エラー箇所と他の箇所

50

とを確実に識別して瞬時にエラー箇所を把握でき、直観的に分かりやすく表示できる。また、数式の演算全体（数式入力～演算結果表示）の時間を短縮できる。また、数式が長い場合にも、1つ以上（特に複数）のエラー箇所を瞬時に把握できる。

【0044】

また、CPU11は、入力された数式の演算指示（「＝」キー26）が入力された場合（時）に、当該数式のエラー箇所を検出する。このため、数式の演算指示を入力した際に、エラー箇所をまとめて修正でき、ユーザの負担を軽減できる。

【0045】

また、CPU11は、エラー箇所の修正の入力に基づいて修正された数式のエラー箇所を検出し、修正された数式のエラー箇所がない場合に、修正された数式の演算を行い、その演算結果を表示する。このため、エラー箇所のない数式の正確な演算結果を表示できるとともに、演算結果の視認により数式にエラー箇所がないことをユーザが確実に認識できる。

10

【0046】

また、数式の表示態様は、コントラスト（背景の白地に対する文字の表示濃度の差分）である。エラー箇所以外の他の箇所（ダークグレー）のコントラストは、エラー箇所（黒）のコントラストよりも低い。このため、白地に対する文字のコントラストを変更可能な表示部14を用いることができ、カラーの表示部を用いるよりも、電子卓上計算機1のコストを低減できる。特に、表現色が白地に対する黒（表示濃度が異なるグレーを含む）の一色に限定されている表示を行う場合であっても、エラー箇所を見やすくできつつ、直観的に分かりやすく表示できる。

20

【0047】

（第1の変形例）

図6～図8（c）を参照して、上記実施の形態の第1の変形例を説明する。図6は、第2の演算表示処理を示すフローチャートである。図7（a）は、表示画面50aを示す図である。図7（b）は、表示画面50bを示す図である。図7（c）は、表示画面50cを示す図である。図7（d）は、表示画面50dを示す図である。図7（e）は、表示画面50eを示す図である。図8（a）は、表示画面50fを示す図である。図8（b）は、表示画面50gを示す図である。図8（c）は、表示画面50hを示す図である。

【0048】

上記実施の形態では、数式のうちのエラー箇所をユーザが全て手動で修正する構成であったが、本変形例では、数式のうちのエラー箇所について、複数の修正候補が表示され、当該修正候補をユーザが択一的に選択入力して修正する構成とする。

30

【0049】

本変形例の装置構成として、上記実施の形態と同様に、電子卓上計算機1を用いるものとする。ただし、記憶部15には、第1の演算表示プログラムP1に代えて、後述する第2の演算表示処理を実行するための第2の演算表示プログラムが記載されているものとする。

【0050】

つぎに、図6～図8（c）を参照して、本変形例の電子卓上計算機1の動作を説明する。電子卓上計算機1において、例えば、操作部12の機能キー20eの電源キーが入力されて電源オンされたことをトリガとして、CPU11は、第2の演算表示プログラムを記憶部15から読み出し、読み出した第2の演算表示プログラムに従い、第2の演算表示処理を実行する。

40

【0051】

図6に示すように、まず、ステップS31～S35のそれぞれは、図3の第1の演算表示処理のステップS11～S15と同様である。そして、CPU11は、ステップS31で入力された数式のうち、ステップS33で構文エラーであると判別されたエラー箇所のうちの1番目のエラー箇所を、対象エラー箇所として選択する（ステップS36）。数式内のエラー箇所は、少なくとも1つである。

50

【 0 0 5 2 】

そして、CPU 11は、入力された数式のうち、ステップS 13で構文エラーと判別されたエラー箇所を黒で表示部14に表示し、エラー箇所以外の箇所を黒よりも抑制的なライトグレーで表示部14に表示する（ステップS 37）。そして、CPU 11は、ステップS 36で選択中の対象エラー箇所（数式内のエラー箇所が1つの場合には、その1つのエラー箇所）にカーソル「|」を移動して黒で点滅表示し、対象エラー箇所に対応する複数の修正候補をライトグレーで表示する（ステップS 38）。

【 0 0 5 3 】

そして、CPU 11は、操作部12を介して、表示中の数式の複数の修正候補の択一的な選択入力を受け付ける（ステップS 39）。そして、CPU 11は、表示中の数式のうち、対象エラー箇所を黒で表示部14に表示したまま、対象エラー箇所以外のエラー箇所（エラー箇所が複数ある場合）を黒よりも抑制的なダークグレーで表示し、ステップS 39で選択入力された修正候補を黒で表示部14に表示する（ステップS 40）。

【 0 0 5 4 】

そして、CPU 11は、ステップS 39で選択入力された修正候補で表示中の対象エラー箇所を修正してライトグレーで表示部14に表示する（ステップS 41）。そして、CPU 11は、表示中の数式に構文エラーがあるか否かを判別する（ステップS 42）。

【 0 0 5 5 】

構文エラーがある場合（ステップS 42；YES）、ステップS 36に移行される。構文エラーがない場合（ステップS 33又はS 42；NO）、ステップS 43に移行される。ステップS 43は、図3のステップS 20と同様である。

【 0 0 5 6 】

ここで、図7（a）～図8（c）を参照して、第2の演算表示処理の具体例を説明する。第2の演算表示処理の開始後、図7（a）に示すように、ステップS 31で例えば、数式「 $6 \div \times 6 + 9 \times \div 3$ 」が入力され、表示画面50aが表示部14に表示された場合を考える。表示画面50aは、黒の数式「 $6 \div \times 6 + 9 \times \div 3$ 」を有する。この数式のうち、「 $\div \times$ 」、「 $\times \div$ 」は、演算子が連続しており、構文エラーのエラー箇所である。

【 0 0 5 7 】

そして、ステップS 32で、「=」キー26が押下入力されると、ステップS 33で、表示中の数式に構文エラーがあると判別され、図7（b）に示すように、ステップS 34で表示画面50bが表示される。表示画面50bは、黒の「Syntax ERROR」と、黒の反転表示の「Back」とを有する。そして、ステップS 35で、決定キー25又は「=」キー26が押下入力されると、ステップS 36で、1番目のエラー箇所「 $\div \times$ 」が対象エラー箇所として選択される。

【 0 0 5 8 】

そして、図7（c）に示すように、ステップS 37、S 38で表示画面50cが表示される。表示画面50cは、ライトグレーの「6」と、黒の1番目のエラー箇所（対象エラー箇所）「 $\div \times$ 」と、ライトグレーの「 $6 + 9$ 」と、黒の2番目のエラー箇所「 $\times \div$ 」と、ライトグレーの「3」とを、一連の数式として有する。また、表示画面50cにおいて、黒のカーソル「|」が、1番目のエラー箇所「 $\div \times$ 」の中間位置に移動されて表示されている。数式の全てのエラー箇所のうち、前側（左側）のエラー箇所から後側（右側）のエラー箇所へ順に移動する修正の対象エラー箇所にカーソル「|」が移動される。

【 0 0 5 9 】

さらに、表示画面50cは、1番目のエラー箇所「 $\div \times$ 」に対応するライトグレーの「1. $\div$ 2. $\times$ 」を修正候補として有する。より具体的には、1番目の修正候補「 $\div$ 」及び2番目の修正候補「 $\times$ 」から1つが選択入力可能である。例えば、下矢印キー22の押下入力により、修正候補の選択入力モードに移行され、表示中のカーソル「|」が消去され、右矢印キー23、左矢印キー24の押下入力により、1つの修正候補がアクティブにされ、決定キー25の押下入力により、アクティブな修正候補の選択が決定される。

【 0 0 6 0 】

10

20

30

40

50

そして、ステップS 3 9で、上述したように、ユーザからの下矢印キー2 2、右矢印キー2 3、左矢印キー2 4の押下入力により、1番目のエラー箇所（対象エラー箇所）に対応する1番目の修正候補「÷」が選択入力されたものとする。すると、図7（d）に示すように、ステップS 4 0で、表示画面5 0 dが表示される。表示画面5 0 dは、ライトグレーの「6」と、黒の1番目のエラー箇所「÷×」と、ライトグレーの「6 + 9」と、ダークグレーの2番目のエラー箇所「×÷」と、ライトグレーの「3」とを、一連の数式として有し、1番目のエラー箇所「÷×」に対応する選択された黒の1番目の修正候補「1 . ÷」と、未選択のライトグレーの2番目の修正候補「2 . ×」とを、選択済の修正候補として有する。このように、対象エラー箇所としての1番目のエラー箇所「÷×」に着目すると、2番目のエラー箇所「×÷」は、黒の対象エラー箇所「÷×」の修正と連動して、当該黒の対象エラー箇所「÷×」と異なるエラー箇所であることを明確にするため、ダークグレーに表示される。

10

【0 0 6 1】

なお、ステップS 3 9で、仮に、ユーザからの下矢印キー2 2、右矢印キー2 3、左矢印キー2 4の押下入力により、1番目のエラー箇所（対象エラー箇所）に対応する2番目の修正候補「×」が選択入力された場合には、図7（e）に示すように、ステップS 4 0で、表示画面5 0 eが表示される。表示画面5 0 eは、ライトグレーの「6」と、黒の1番目のエラー箇所「÷×」と、ライトグレーの「6 + 9」と、ダークグレーの2番目のエラー箇所「×÷」と、ライトグレーの「3」とを、一連の数式として有し、1番目のエラー箇所「÷×」に対応する未選択のライトグレーの2番目の修正候補「1 . ÷」と、選択された黒の2番目の修正候補「2 . ×」とを、選択済の修正候補として有する。

20

【0 0 6 2】

そして、ステップS 4 1で、選択された修正候補「1 . ÷」により、黒の1番目のエラー箇所「÷×」が、ライトグレーの「÷」に修正されて表示される。そして、ステップS 4 2で、エラー箇所「×÷」が残っているため、表示中の数式に構文エラーがあると判別され、ステップS 3 6に移行される。

【0 0 6 3】

同様にして、ステップS 3 6で、未修正の新たな1番目（旧2番目）のエラー箇所「×÷」が対象エラー箇所として選択される。そして、図8（a）に示すように、ステップS 3 7、S 3 8で表示画面5 0 fが表示される。表示画面5 0 fは、ライトグレーの「6 ÷ 6 + 9」と、黒の1番目のエラー箇所「×÷」と、ライトグレーの「3」とを、一連の数式として有する。表示画面5 0 fにおいて、対象エラー箇所「×÷」は、表示画面5 0 d又は5 0 eでのダークグレーから黒の表示に戻る。また、表示画面5 0 fにおいて、黒のカーソル「|」が、1番目のエラー箇所「×÷」の中間位置に移動されて表示されている。

30

【0 0 6 4】

さらに、表示画面5 0 fは、1番目のエラー箇所「×÷」に対応するライトグレーの「1 . × 2 . ÷」を修正候補として有する。そして、ステップS 3 9で、ユーザからの下矢印キー2 2、右矢印キー2 3、左矢印キー2 4の押下入力により、1番目の修正候補「×」が選択入力されたものとする。すると、図8（b）に示すように、ステップS 4 0で、表示画面5 0 gが表示される。表示画面5 0 gは、ライトグレーの「6 ÷ 6 + 9」と、黒の1番目のエラー箇所「×÷」と、ライトグレーの「3」とを、一連の数式として有し、1番目のエラー箇所「×÷」に対応する選択された黒の修正候補「1 . ×」と、未選択のライトグレーの修正候補「2 . ÷」とを選択済の修正候補として有する。

40

【0 0 6 5】

そして、ステップS 4 1で、選択された修正候補「1 . ×」により、黒の1番目のエラー箇所「×÷」が、「×」に修正されてライトグレーで表示される。そして、ステップS 4 2で、エラー箇所が残っていないため、表示中の数式に構文エラーがないと判別され、ステップS 4 3に移行される。そして、図8（c）に示すように、ステップS 4 3で、演算結果の表示画面5 0 hが表示される。表示画面5 0 hは、黒の数式「6 ÷ 6 + 9 × 3」と、黒の演算結果「2 8」と、を有する。

50

【 0 0 6 6 】

以上、本変形例によれば、CPU 11は、数式にエラー箇所が検出された場合に、修正候補を表示部 14に表示させる。このため、修正候補を選択入力することにより、数式のエラー箇所を、修正の入力ミスを低減して容易かつ確実に修正でき、ユーザの負担を軽減できる。

【 0 0 6 7 】

(第2の変形例)

図9～図11(d)を参照して、上記実施の形態の第2の変形例を説明する。図9は、第3の演算表示処理を示すフローチャートである。図10(a)は、第1の具体例の表示画面60aを示す図である。図10(b)は、第1の具体例の表示画面60bを示す図である。図10(c)は、第1の具体例の表示画面60cを示す図である。図10(d)は、第1の具体例の表示画面60dを示す図である。図10(e)は、第1の具体例の表示画面60eを示す図である。図10(f)は、第1の具体例の表示画面60fを示す図である。図11(a)は、表示画面70aを示す図である。図11(b)は、表示画面70bを示す図である。図11(c)は、表示画面70cを示す図である。図11(d)は、表示画面70dを示す図である。

10

【 0 0 6 8 】

上記実施の形態では、入力した数式の演算指示が入力されたときに、エラー箇所を表示して修正する構成であったが、本変形例では、数式の入力中(演算指示前)に、エラー箇所を表示して修正する構成とする。

20

【 0 0 6 9 】

本変形例の装置構成として、上記実施の形態と同様に、電子卓上計算機1を用いるものとする。ただし、記憶部15には、第1の演算表示プログラムP1に代えて、後述する第3の演算表示処理を実行するための第3の演算表示プログラムが記載されているものとする。

【 0 0 7 0 】

つぎに、図9～図11(d)を参照して、本変形例の電子卓上計算機1の動作を説明する。電子卓上計算機1において、例えば、操作部12の機能キー20eの電源キーが入力されて電源オンされたことをトリガとして、CPU 11は、第3の演算表示プログラムを記憶部15から読み出し、読み出した第3の演算表示プログラムに従い、第3の演算表示処理を実行する。

30

【 0 0 7 1 】

図9に示すように、まず、CPU 11は、操作部12を介して、ユーザからの数式の入力を受け付け、入力された数式を表示部14に設定色で表示する(ステップS51)。ステップS51の設定色とは、初期設定が黒であるものとし、その後に、後述するステップS60、S61で設定された色(ダークグレー又は黒(黒の場合は全ての数式))とする。

【 0 0 7 2 】

そして、CPU 11は、表示中の数式に構文エラーがあるか否かを判別する(ステップS52)。構文エラーがない場合(ステップS52; NO)、CPU 11は、「=」キー26が押下入力されたか否かを判別する(ステップS53)。「=」キー26が入力されていない場合(ステップS53; NO)、ステップS51に移行される。

40

【 0 0 7 3 】

構文エラーがある場合(ステップS52; YES)、CPU 11は、ステップS51で入力された数式のうち、ステップS52で構文エラーと判別されたエラー箇所を黒で表示部14に表示し、エラー箇所以外の箇所を黒よりも抑制的なダークグレーで表示部14に表示する(ステップS54)。数式内のエラー箇所は、1つだけでなく、複数の場合もあり得る。

【 0 0 7 4 】

そして、CPU 11は、ステップS54で黒で表示されたエラー箇所のうち、1番目のエラー箇所に対応する複数の修正候補をライトグレーで表示部14に表示する(ステップ

50

S 5 5)。そして、C P U 1 1 は、操作部 1 2 を介して、ステップ S 5 5 で表示中の数式の複数の修正候補の択一的な選択入力を受け付け、修正候補の入力があるか否かを判別する (ステップ S 5 6)。

【 0 0 7 5 】

修正候補の入力がある場合 (ステップ S 5 6 ; Y E S)、C P U 1 1 は、ステップ S 5 5 で表示された複数の修正候補のうち、ステップ S 5 6 で選択入力された修正候補を黒で表示部 1 4 に表示する (ステップ S 5 7)。そして、C P U 1 1 は、ステップ S 5 7 で選択入力された修正候補に対応するエラー箇所を修正してダークグレーで表示部 1 4 に表示する (ステップ S 5 8)。

【 0 0 7 6 】

そして、C P U 1 1 は、ステップ S 5 8 で修正された数式に構文エラーがあるか否かを判別する (ステップ S 5 9)。修正候補の入力が無い場合 (ステップ S 5 6 ; N O)、又は構文エラーがある場合 (ステップ S 5 9 ; Y E S)、C P U 1 1 は、ステップ S 5 1 で入力される新規数式の設定色としてダークグレーを設定し (ステップ S 6 0)、ステップ S 5 1 に移行する。つまり、ステップ S 6 0 により、数式にエラー箇所 (黒で表示) が残っている場合に、ステップ S 5 1 で新たに入力される数式の文字がダークグレーで表示されるものとする。

【 0 0 7 7 】

構文エラーがない場合 (ステップ S 5 9 ; N O)、C P U 1 1 は、ステップ S 5 1 で入力時に表示される全ての数式の設定色として黒を設定し (ステップ S 6 1)、ステップ S 5 1 に移行する。つまり、ステップ S 6 1 により、数式にエラー箇所がない場合には、ステップ S 5 1 で入力済及び新たに入力される数式の全ての文字が黒で表示されるものとする。

【 0 0 7 8 】

「 = 」キー 2 6 が入力された場合 (ステップ S 5 3 ; Y E S)、ステップ S 6 2 に移行される。ステップ S 6 2 は、図 3 の第 1 の演算表示処理のステップ S 2 0 と同様である。

【 0 0 7 9 】

ついで、図 1 0 (a) ~ 図 1 1 (d) を参照して、第 3 の演算表示処理の 2 つの具体例を説明する。第 3 の演算表示処理の第 1 の具体例として、第 3 の演算表示処理の開始後、図 1 0 (a) に示すように、ステップ S 5 1 で例えば、「 $6 \div \times$ 」の数式が入力され、表示画面 6 0 a が表示部 1 4 に表示された場合を考える。表示画面 6 0 a は、黒の数式「 $6 \div \times$ 」を有し、その数式の後に黒のカーソル「 | 」を有する。この数式のうち、「 $\div \times$ 」は、演算子が連続しており、構文エラーのエラー箇所である。

【 0 0 8 0 】

そして、ステップ S 5 2 で、表示中の数式に構文エラーがあると判別され、図 1 0 (b) に示すように、ステップ S 5 4 , S 5 5 で表示画面 6 0 b が表示される。表示画面 6 0 b は、数式として、黒よりも抑制的なダークグレーの「 6 」と、黒のエラー箇所「 $\div \times$ 」と、黒のカーソル「 | 」と、を有する。また、表示画面 6 0 b において、黒のカーソル「 | 」は、数式の入力中であるため、エラー箇所への移動はなされず、数式の後に表示されている。

【 0 0 8 1 】

さらに、表示画面 6 0 b は、1 番目のエラー箇所「 $\div \times$ 」に対応するライトグレーの「 E r r o r : 1 . \div 2 . \times 」を修正候補として有する。より具体的には、1 番目の修正候補「 $\div$ 」及び 2 番目の修正候補「 $\times$ 」から 1 つが選択入力可能である。

【 0 0 8 2 】

そして、ステップ S 5 6 で、ユーザからの下矢印キー 2 2、右矢印キー 2 3 及び左矢印キー 2 4 の押下入力により、1 番目の修正候補「 $\div$ 」が選択入力されたものとする。すると、図 1 0 (c) に示すように、ステップ S 5 7 で、表示画面 6 0 c が表示される。表示画面 6 0 c は、ライトグレーの「 6 」と、黒のエラー箇所「 $\div \times$ 」とを、一連の数式として有し、黒の「 E r r o r : 」と、エラー箇所「 $\div \times$ 」に対応する選択された黒の 1 番目

10

20

30

40

50

の修正候補「 $1 \div$ 」と、未選択のライトグレーの2番目の修正候補「 $2 \times$ 」とを、選択済の修正候補として有する。

【0083】

なお、ステップS56で、仮に、ユーザからの下矢印キー22、右矢印キー23、左矢印キー24の押下入力により、エラー箇所に対応する2番目の修正候補「 $\times$ 」が選択入力された場合には、図10(d)に示すように、ステップS57で、表示画面60dが表示される。表示画面60dは、ライトグレーの「6」と、黒のエラー箇所「 $\div \times$ 」とを、一連の数式として有し、黒の「Error:」と、エラー箇所「 $\div \times$ 」に対応する未選択のライトグレーの1番目の修正候補「 $1 \div$ 」と、選択された黒の2番目の修正候補「 $2 \times$ 」とを、選択済の修正候補として有する。

10

【0084】

そして、図10(d)に示すように、ステップS58で、表示画面60eが表示される。表示画面60eは、ダークグレーの「6」と、選択された修正候補「 $1 \div$ 」によりエラー箇所が修正された後のダークグレーの「 $\div$ 」とを一連の数式として有し、数式の後に黒のカーソル「|」を有する。そして、ステップS59で、エラー箇所が残っていないため、表示中の数式に構文エラーがないと判別され、ステップS61を介してステップS51に移行される。ステップS51では、エラー箇所がないため、ステップS61で設定色が黒に変更された数式「 $6 \div$ 」の続きの数式が入力されて黒で表示される。例えば、続きの数式として数値「6」が入力されると、黒の数式「 $6 \div 6$ 」が表示される。

【0085】

20

そして、ステップS52で、表示中の数式に構文エラーがないと判別され、ステップS53で「=」キー26が入力されると、図10(f)に示すように、ステップS62で、数式「 $6 \div 6$ 」が演算され、演算結果の表示画面60fが表示部14に表示される。表示画面60fは、黒の数式「 $6 \div 6$ 」と、黒の演算結果「1」とを有する。

【0086】

ついで、第3の演算表示処理の第2の具体例として、例えば、「 $6 \div \times 6 + 9 \times \div 3$ 」の数式(全数式とする)が入力される場合を考える。全数式のうち、「 $\div \times$ 」、「 $\times \div$ 」の2か所は、演算子が連続しており、構文エラーのエラー箇所である。

【0087】

上記全数式のうち、「 $6 \div$ 」までの入力は、ステップS51～S53のループでの実行により、数式の入力及び黒の数式の表示が実行される。そして、ステップS51で、「 $\times$ 」が入力されると、黒の数式「 $6 \div \times$ 」が表示され、ステップS52で、表示中の数式に構文エラーがあると判別される。そして、ステップS54で、ダークグレーの「6」と黒の(1番目の)エラー箇所「 $\div \times$ 」とが一連の数式として表示され、ステップS55で1番目のエラー箇所「 $\div \times$ 」に対応するライトグレーの修正候補「Error: $1 \div 2 \times$ 」が表示される。

30

【0088】

しかし、ステップS56で、修正候補の入力がなされないものとする。すると、ステップS60を介して、再度ステップS51に移行され、数式として、ダークグレーの「6」と、黒の1番目のエラー箇所「 $\div \times$ 」とが一連の数式として表示され、1番目のエラー箇所「 $\div \times$ 」に対応するライトグレーの修正候補「Error: $1 \div 2 \times$ 」も表示されたままとなる。

40

【0089】

同様に、ステップS51～S56, S60のループの実行が繰り返され、図11(a)に示すように、ステップS51で、全数式の最後の「3」が入力されると、表示画面70aが表示される。表示画面70aは、ダークグレーの「6」と、黒の1番目のエラー箇所「 $\div \times$ 」と、ダークグレーの「 $6 + 9$ 」と、黒の2番目のエラー箇所「 $\times \div$ 」と、ダークグレーの「3」とを、一連の数式として有し、1番目のエラー箇所「 $\div \times$ 」に対応するライトグレーの修正候補「Error: $1 \div 2 \times$ 」を有する。

【0090】

50

そして、ステップS56で、1番目のエラー箇所「 $\div \times$ 」に対応する修正候補「 $1 \div$ 」が入力されると、図11(b)に示すように、ステップS57で、表示画面70bが表示される。表示画面70bは、ダークグレーの「6」と、黒の1番目のエラー箇所「 $\div \times$ 」と、ダークグレーの「 $6 + 9$ 」と、黒の2番目のエラー箇所「 $\times \div$ 」と、ダークグレーの「3」とを、一連の数式として有し、黒の「Error:」と、選択済の黒の修正候補「 $1 \div$ 」と、未選択のライトグレーの修正候補「 $2 \times$ 」とを有する。そして、ステップS58で、選択済の修正候補「 $1 \div$ 」により、1番目のエラー箇所「 $\div \times$ 」が修正され、ダークグレーの「6」と、ダークグレーの修正後の1番目のエラー箇所「 $\div$ 」と、ダークグレーの「 $6 + 9$ 」と、黒の2番目のエラー箇所「 $\times \div$ 」と、ダークグレーの「3」とが、一連の数式として表示される。

10

【0091】

しかし、ステップS59で、2番目のエラー箇所「 $\times \div$ 」により、表示中の数式に構文エラーがあると判別され、ステップS60を介して、ステップS51に移行される。ステップS51では、表示画面における表示中の数式は変わらず、また修正後のエラー箇所に対応する修正候補は表示されない。

【0092】

そして、ステップS51で、新規入力がない場合に、ステップS52で、表示中の数式に構文エラーがあると判別される。そして、ステップS54で、表示画面における表示中の数式は変わらず、ステップS55で、当該表示画面中で、2番目のエラー箇所「 $\times \div$ 」に対応するライトグレーの修正候補「Error: $1 \times 2 \div$ 」が表示される。

20

【0093】

そして、ステップS56で、2番目のエラー箇所「 $\times \div$ 」に対応する1番目の修正候補「 $1 \times$ 」が入力されると、図11(c)に示すように、ステップS57で、表示画面70cが表示される。表示画面70cは、ダークグレーの「6」と、ダークグレーの修正後の1番目のエラー箇所「 $\div$ 」と、ダークグレーの「 $6 + 9$ 」と、黒の2番目のエラー箇所「 $\times \div$ 」と、ダークグレーの「3」とを、一連の数式として有し、黒の「Error:」と、選択済の黒の修正候補「 $1 \times$ 」と、未選択のライトグレーの修正候補「 $2 \div$ 」とを有する。そして、ステップS58で、選択済の修正候補「 $1 \times$ 」により、2番目のエラー箇所「 $\times \div$ 」が修正され、ダークグレーの「6」と、ダークグレーの修正後の1番目のエラー箇所「 $\div$ 」と、ダークグレーの「 $6 + 9$ 」と、ダークグレーの修正後の2番目のエラー箇所「 $\times$ 」と、ダークグレーの「3」とが、一連の数式として表示される。

30

【0094】

そして、ステップS59で、表示中の数式に構文エラーがないと判別され、ステップS61を介して、ステップS51に移行される。ステップS51では、設定色が黒の「6」と、黒の修正後の1番目のエラー箇所「 $\div$ 」と、黒の「 $6 + 9$ 」と、黒の修正後の2番目のエラー箇所「 $\times$ 」と、黒の「3」とが、一連の数式として表示される。

【0095】

そして、ステップS51で、新規入力されず、ステップS52で、表示中の数式に構文エラーがないと判別され、ステップS53で、「=」キー26が入力されると、図11(d)に示すように、ステップS62で、演算結果の表示画面70dが表示される。表示画面70dは、黒の数式「 $6 \div 6 + 9 \times 3$ 」と、黒の演算結果「28」と、を有する。

40

【0096】

以上、本変形例によれば、CPU11は、数式の入力中に、当該数式のエラー箇所を検出する。このため、数式の入力中に常時エラー箇所を確認でき、数式の演算全体（数式入力～演算結果表示）の時間の早い段階でエラー箇所を把握及び修正でき、数式の演算全体の時間を短縮できる。また、ユーザがエラー箇所の発生の都度又はまとめて当該エラー箇所を修正でき、修正の利便性を向上できる。

【0097】

また、CPU11は、数式の入力中に、エラー箇所の修正の入力に基づいて修正された数式のエラー箇所を検出し、修正された数式のエラー箇所がなく、かつ当該修正された数

50

式の演算指示が入力された場合に、修正された数式の演算を行い、演算の結果を表示部 14 に表示する。このため、エラー箇所のない数式の正確な演算結果を表示できるとともに、演算結果の視認により数式にエラー箇所がないことをユーザが確実に認識できる。

【0098】

なお、本変形例において、上記の例では、数式の構文エラーの判別時に、数式のうちのエラー箇所の表示態様を黒に維持し、エラー箇所以外の箇所の表示態様を黒よりも抑制的なダークグレーにする構成としたが、これに限定されるものではない。例えば、数式の構文エラーの判別後に、数式のうちのエラー箇所の表示態様を所定色（例えばダークグレー）にし、エラー箇所以外の箇所の表示態様を例えば黒にし、当該エラー箇所に対応する複数の修正候補を表示し、操作部 12 を介する修正候補の選択入力後に、数式のうちのエラー箇所の表示態様を所定色（ダークグレー）に維持し、エラー箇所以外の箇所の表示態様を所定色（ダークグレー）よりも抑制的な表示態様（例えば、ライトグレー）にする構成としてもよい。

10

【0099】

以上の説明では、本発明に係るプログラムのコンピュータ読み取り可能な媒体として記憶部 15（ROM）を使用した例を開示したが、この例に限定されない。その他のコンピュータ読み取り可能な媒体として、フラッシュメモリなどの不揮発性メモリ、CD-ROMなどの可搬型記録媒体を適用することが可能である。また、本発明に係るプログラムのデータを通信回線を介して提供する媒体として、キャリアウエーブ（搬送波）も本発明に適用される。

20

【0100】

なお、上記実施の形態における記述は、本発明に係る入力支援装置、入力支援方法及びプログラムの一例であり、これに限定されるものではない。例えば、上記実施の形態、第 1 及び第 2 の変形例の少なくとも 2 つを適宜組み合わせる構成としてもよい。

【0101】

また、上記実施の形態及び変形例において、数式のエラー箇所が検出された場合に、エラー箇所の維持する表示態様が、背景の白地に対する文字のコントラストであり、エラー箇所以外の他の箇所のコントラストが、エラー箇所のコントラストよりも低い構成としたが、これに限定されるものではない。例えば、表示部 14 が背景の黒地に情報（文字）を表示するものである場合に、エラー箇所の維持する表示態様が、背景の黒地に対する文字のコントラスト（黒地に対する文字の表示濃度の差分）であり、エラー箇所以外の他の箇所のコントラストが、エラー箇所のコントラストよりも低い（例えば、エラー箇所の文字が黒地に白で、他の箇所の文字が黒地にグレー）構成としてもよい。

30

【0102】

また、数式のエラー箇所の維持する表示態様が、表示する文字の太さ（表示太さとする）であり、エラー箇所以外の他の箇所の表示太さが、エラー箇所の表示太さよりも細い構成としてもよい。また、数式のエラー箇所の維持する表示態様が、表示する文字のサイズ（表示サイズとする）であり、エラー箇所以外の他の箇所の文字の表示サイズが、エラー箇所の文字の表示サイズよりも小さい構成としてもよい。また、数式のエラー箇所の維持する表示態様が、表示する文字のフォント（表示フォントとする）であり、エラー箇所以外の他の箇所の文字の表示フォントが、エラー箇所の文字の表示フォントよりも抑制的（例えば、同じ文字でも黒のドット数が少ない種類のフォント）である構成としてもよい。また、電子卓上計算機 1 の表示部 14 が、カラーの表示情報を表示可能である場合に、数式のエラー箇所の維持する表示態様が、表示する文字のカラーの表示色（カラー表示色とする）であり、エラー箇所以外の他の箇所の文字のカラー表示色が、エラー箇所の文字のカラー表示色よりも抑制的（例えば、暗いカラーの色）である構成としてもよい。

40

【0103】

また、上記実施の形態及び変形例において、入力支援装置として、関数電卓である電子卓上計算機 1 を用いる構成としたが、これに限定されるものではない。入力支援装置としては、数式を入力及び表示可能で、その表示態様を制御可能な装置であればよく、通常の

50

電子卓上計算機や、電子卓上計算機以外の携帯機器、据え置き型の情報処理装置など、他の入力支援装置を用いるとしてもよい。

【 0 1 0 4 】

本発明の実施の形態及び変形例を説明したが、本発明の範囲は、上述の実施の形態及び変形例に限定するものではなく、特許請求の範囲に記載された発明の範囲とその均等の範囲を含む。

以下に、この出願の願書に最初に添付した特許請求の範囲に記載した発明を付記する。付記に記載した請求項の項番は、この出願の願書に最初に添付した特許請求の範囲の通りである。

〔 付 記 〕

10

< 請求項 1 >

入力された数式を対象にして数式のエラー箇所を検出する検出手段と、

前記検出手段によりエラー箇所が検出された場合に、当該エラー箇所の表示態様を維持したままで前記数式における他の箇所の表示態様を抑制的な表示態様に切り替える表示制御手段と、

を備えることを特徴とする入力支援装置。

< 請求項 2 >

前記表示制御手段は、前記検出手段によりエラー箇所が検出された場合に、修正候補を表示させる、

ことを特徴とする請求項 1 に記載の入力支援装置。

20

< 請求項 3 >

前記検出手段は、前記入力された数式の演算指示が入力された場合に、当該数式のエラー箇所を検出する、

ことを特徴とする請求項 1 又は 2 に記載の入力支援装置。

< 請求項 4 >

前記検出手段は、前記エラー箇所の修正の入力に基づいて修正された数式のエラー箇所を検出し、

前記表示制御手段は、前記修正された数式のエラー箇所がない場合に、当該修正された数式の演算を行い、当該演算の結果を表示する、

ことを特徴とする請求項 3 に記載の入力支援装置。

30

< 請求項 5 >

前記検出手段は、前記数式の入力中に、当該数式のエラー箇所を検出する、

ことを特徴とする請求項 1 又は 2 に記載の入力支援装置。

< 請求項 6 >

前記検出手段は、前記数式の入力中に、前記エラー箇所の修正の入力に基づいて修正された数式のエラー箇所を検出し、

前記表示制御手段は、前記修正された数式のエラー箇所がなく、かつ当該修正された数式の演算指示が入力された場合に、当該修正された数式の演算を行い、当該演算の結果を表示する、

ことを特徴とする請求項 5 に記載の入力支援装置。

40

< 請求項 7 >

前記表示態様は、コントラストであり、

前記他の箇所のコントラストは、前記エラー箇所のコントラストよりも低い、

ことを特徴とする請求項 1 又は 2 に記載の入力支援装置。

< 請求項 8 >

入力された数式を対象にして数式のエラー箇所を検出する検出工程と、

前記検出工程においてエラー箇所が検出された場合に、当該エラー箇所の表示態様を維持したままで前記数式における他の箇所の表示態様を抑制的な表示態様に切り替える表示制御工程と、

を含むことを特徴とする入力支援方法。

50

< 請求項 9 >

コンピュータを、
入力された数式を対象にして数式のエラー箇所を検出する検出手段、
前記検出手段によりエラー箇所が検出された場合に、当該エラー箇所の表示態様を維持したままで前記数式における他の箇所の表示態様を抑制的な表示態様に切り替える表示制御手段、
として機能させるためのプログラム。

【符号の説明】

【 0 1 0 5 】

1	電子卓上計算機	10
1 1	C P U	
1 2	操作部	
2 0 a	テンキー	
2 0 b	演算キー	
2 0 c	修正キー	
2 0 d	関数キー	
2 1	上矢印キー	
2 2	下矢印キー	
2 3	右矢印キー	
2 4	左矢印キー	20
2 5	決定キー	
2 6	「 = 」キー	
2 0 e	機能キー	
1 3	R A M	
1 4	表示部	
1 5	記憶部	
1 6	バス	

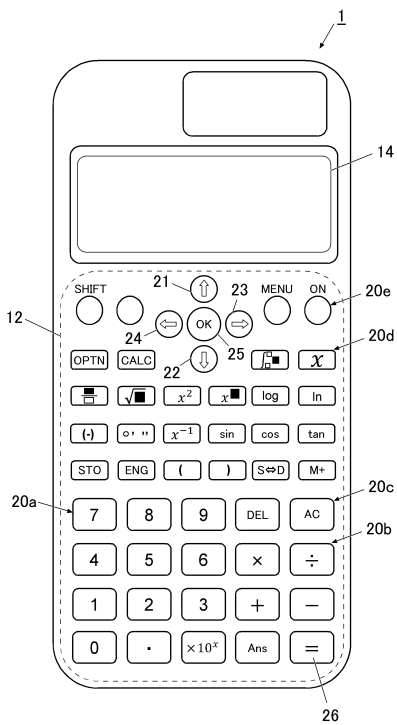
30

40

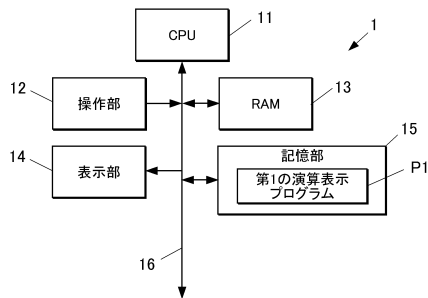
50

【図面】

【図 1】



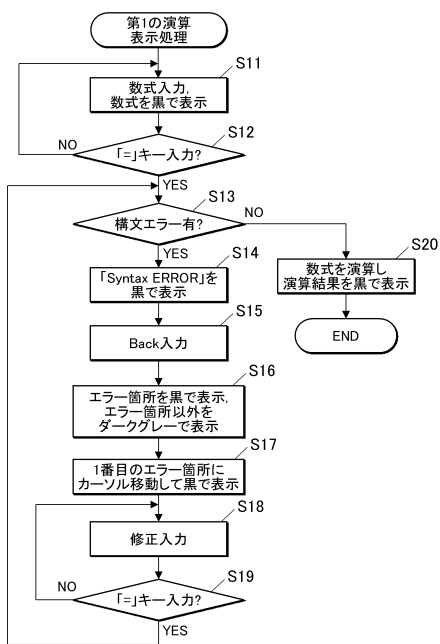
【図 2】



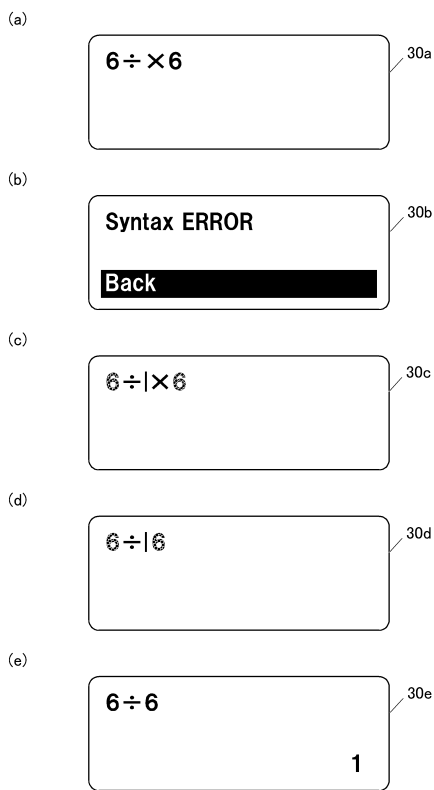
10

20

【図 3】



【図 4】

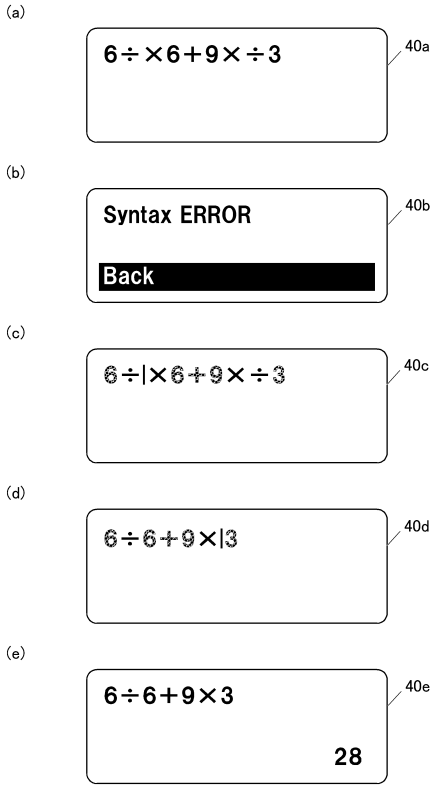


30

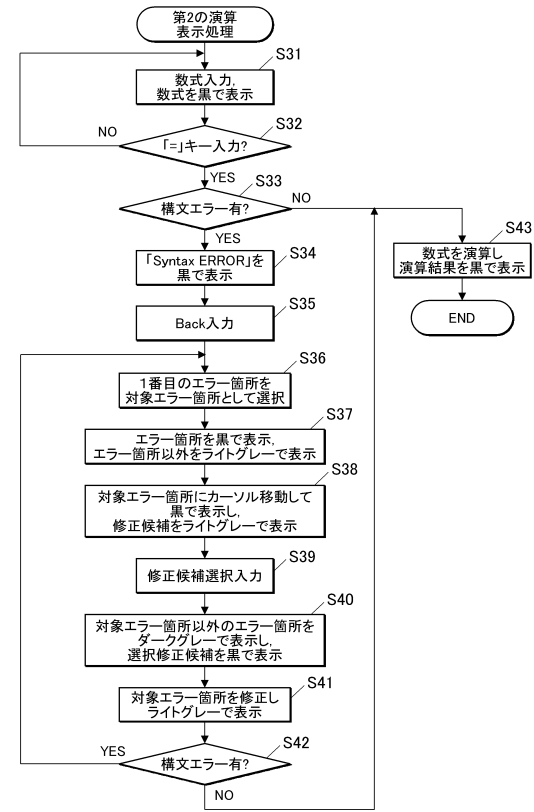
40

50

【図 5】



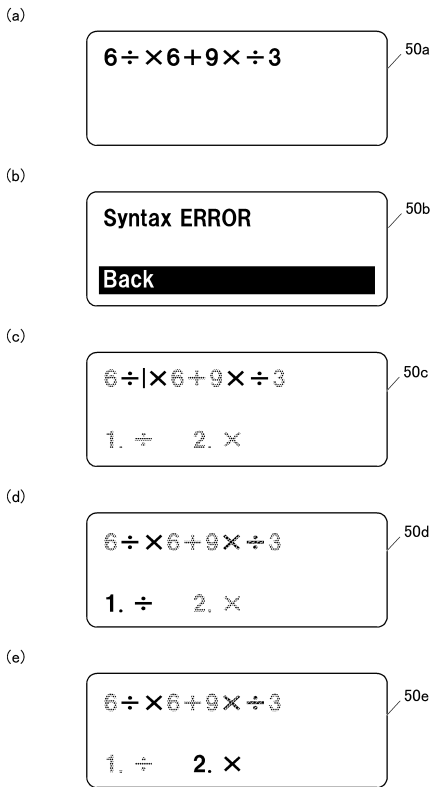
【図 6】



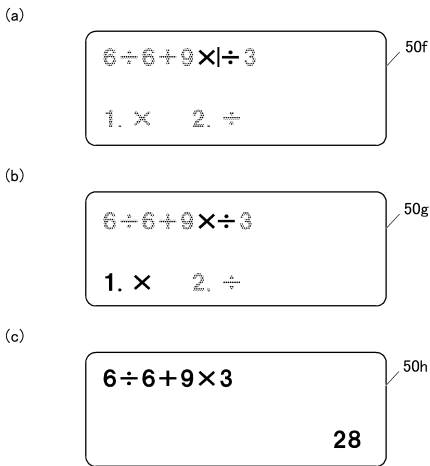
10

20

【図 7】



【図 8】

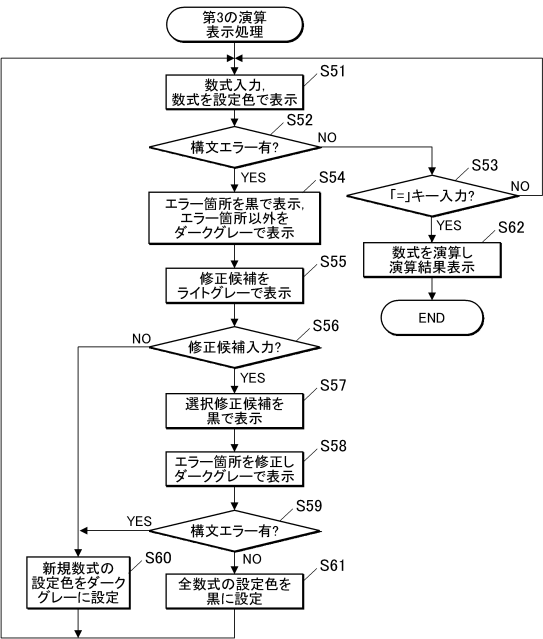


30

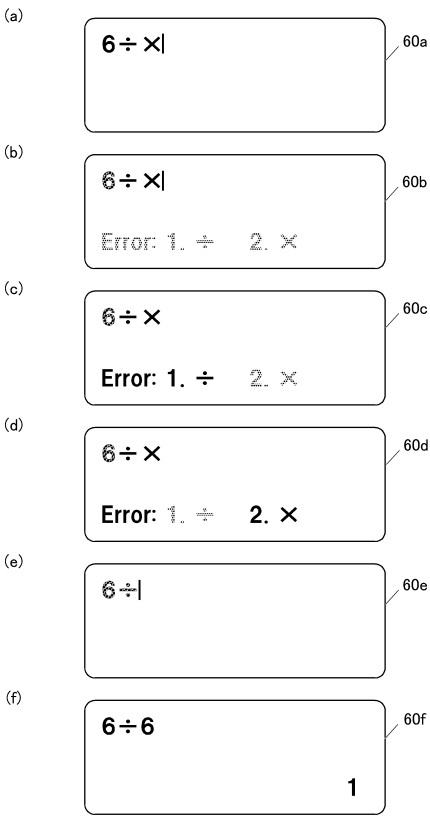
40

50

【図 9】



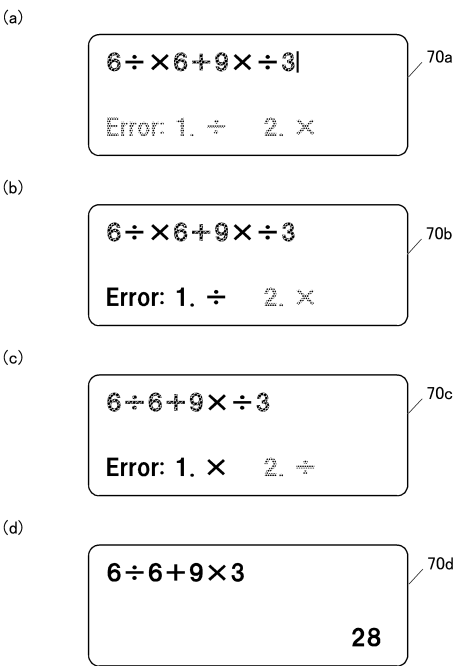
【図 10】



10

20

【図 11】



30

40

50

フロントページの続き

- (56)参考文献 特開 2 0 1 1 - 1 8 0 7 1 0 (J P , A)
特開平 1 0 - 1 8 7 2 3 3 (J P , A)
特表 2 0 2 0 - 5 2 5 8 6 8 (J P , A)
特開 2 0 1 1 - 1 8 6 7 7 1 (J P , A)
特開 2 0 1 1 - 1 8 6 5 6 7 (J P , A)
特開 2 0 1 4 - 0 6 7 3 9 0 (J P , A)
- (58)調査した分野 (Int.Cl. , D B 名)
G 0 6 F 3 / 0 1 - 3 / 0 2 7
G 0 6 F 3 / 0 4 8 - 3 / 0 4 8 9 5
G 0 6 F 1 5 / 0 2