

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5458065号
(P5458065)

(45) 発行日 平成26年4月2日(2014.4.2)

(24) 登録日 平成26年1月17日(2014.1.17)

(51) Int.Cl. F I
G06F 9/44 (2006.01) G06F 9/06 620K

請求項の数 11 (全 29 頁)

(21) 出願番号	特願2011-162519 (P2011-162519)	(73) 特許権者	511180640
(22) 出願日	平成23年7月25日 (2011.7.25)		石田 伸一
(65) 公開番号	特開2013-25716 (P2013-25716A)		神奈川県小田原市曾我岸102番地
(43) 公開日	平成25年2月4日 (2013.2.4)	(73) 特許権者	593225736
審査請求日	平成24年4月9日 (2012.4.9)		松崎 務
			東京都稲城市平尾3丁目7番地の5 平尾 住宅67棟501号
		(73) 特許権者	504346938
			アイ・システム株式会社
			東京都千代田区九段南4丁目8番13号
		(74) 代理人	100117341
			弁理士 山崎 拓哉
		(74) 代理人	100148840
			弁理士 松本 健志

最終頁に続く

(54) 【発明の名称】 モジュールの構造解析を支援する装置及びプログラム

(57) 【特許請求の範囲】

【請求項1】

プログラム言語によって記述された1以上の文からなるモジュールを1以上の二次的文からなる二次的モジュールに変換し、これにより複数の箇所分散された機能を1つに纏めて把握するための前記二次的モジュールを作成する装置であって、

前記モジュールはモジュール名によって特定可能であり、前記モジュールには、1以上の前記文から構成される処理単位であって処理単位名によって特定可能な前記処理単位を記述することが可能であり、

前記モジュールには、前記文として、前記処理単位を前記処理単位名によって特定して実行するための処理単位実行文と、前記モジュールを前記モジュール名によって特定して実行するためのモジュール呼び出し文と、条件式が成立するか成立しないかによって実行すべき前記文を選択するための条件文とを記述可能であり、前記条件文によって選択される前記文として更に前記条件文が記述可能であり、これにより前記条件文による前記文の階層化が可能であり、

前記装置は、1以上の前記モジュールが記憶された記憶手段と、前記記憶手段に記憶された前記モジュールのうちの所定の対象モジュールを前記二次的モジュールに変換して前記記憶手段に記憶する変換手段とを備えており、

前記変換手段は、前記対象モジュールを前記記憶手段から読み込む読込機能と、前記対象モジュールの所定の処理開始箇所から所定の処理終了箇所までに記述された前記文の夫々を前記二次的文として前記二次的モジュールに順次出力すると共に、前記文が前記処理

単位実行文又は前記モジュール呼び出し文である場合、実行対象の前記処理単位又は呼び出し対象の前記モジュールに記述されている前記文を展開することで前記二次的文を生成して前記二次的モジュールに出力する変換機能であって、前記展開される前記文における前記処理単位実行文又は前記モジュール呼び出し文についても前記展開を更に行う前記変換機能と、前記二次的モジュールを前記記憶手段に記憶する書込機能とを有しているモジュールの構造解析を支援する装置。

【請求項 2】

請求項 1 記載の装置であって、

前記変換手段は、展開制御情報を取得する制御情報取得機能を有しており、前記展開制御情報は、実行対象の前記処理単位又は呼び出し対象の前記モジュールに記述されている前記文の前記展開を初回のみを制限することを指示する展開制限指示を指定することが可能なものであり、

10

前記変換手段は、前記変換機能において、前記展開制御情報に前記展開制限指示が指定されている場合、同一の前記処理単位については、実行対象の前記処理単位に記述されている前記文の前記展開を初回の 1 回のみ行い、同一の前記モジュールについては、呼び出し対象の前記モジュールに記述されている前記文の前記展開を初回の 1 回のみ行うモジュールの構造解析を支援する装置。

【請求項 3】

請求項 1 又は請求項 2 記載の装置であって、

前記変換手段は、前記変換機能において、前記文が前記条件文である場合、前記条件文の前記条件式に基づいて前記二次的文の一つである二次的条件文を生成し、前記二次的条件文によって前記二次的文を階層化して前記二次的モジュールに出力するものであり、

20

前記二次的条件文は、2 つの評価対象オペランド間の関係を 1 つの論理演算子のみによって評価する二次的条件式が成立するか成立しないかによって実行すべき前記二次的文を選択するものである

モジュールの構造解析を支援する装置。

【請求項 4】

請求項 3 記載の装置であって、

前記変換手段は、前記二次的モジュールに出力する前記二次的文の少なくとも一部について、前記二次的文がいずれの前記階層に位置しているかを示す階層識別を付加する階層識別付加機能と、前記二次的モジュールに出力する前記二次的文のうち定数又は変数からなる 2 つのオペランドと 1 つの演算子とを含むものについて、2 つの前記オペランドと 1 つの前記演算子とからなるオペランド情報を前記二次的文に付加するオペランド付加機能とのうちの少なくとも一方を有している

30

モジュールの構造解析を支援する装置。

【請求項 5】

請求項 3 記載の装置によって出力される二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールを加工する加工手段を備えており、

40

前記加工手段は、読込機能と、階層識別付加機能とオペランド付加機能とのうちの少なくとも一方と、書込機能とを有しており、

前記読込機能は、前記二次的モジュールを前記記憶手段から読み込む機能であり、

前記階層識別付加機能は、前記二次的モジュールの前記二次的文の少なくとも一部について、前記二次的文がいずれの前記階層に位置しているかを示す階層識別を付加する機能であり、

前記オペランド付加機能は、前記二次的モジュールの前記二次的文のうち定数又は変数からなる 2 つのオペランドと 1 つの演算子とを含むものについて、2 つの前記オペランドと 1 つの前記演算子とからなるオペランド情報を前記二次的文に付加する機能であり、

前記書込機能は、前記階層識別又は前記オペランド情報が付加された前記二次的文から

50

なる前記二次的モジュールを前記記憶手段に記憶する機能である
モジュールの構造解析を支援する装置。

【請求項 6】

請求項 4 又は請求項 5 記載の装置によって出力される二次的モジュールであって、少なくとも階層識別が付加された二次的文からなる前記二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールの前記二次的文の一部を選択する選択手段を備えており、

前記選択手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、前記二次的文のうち選択対象とすべきものの前記階層識別を特定する階層識別特定情報を取得する特定情報取得機能と、前記二次的モジュールの前記二次的文のうち前記階層識別が前記階層識別特定情報によって特定される前記階層識別と同一であるものを選択して選択後の二次的モジュールに出力する選択機能と、前記選択後の二次的モジュールを前記記憶手段に記憶する書込機能とを有している

モジュールの構造解析を支援する装置。

【請求項 7】

請求項 4 又は請求項 5 記載の装置によって出力される二次的モジュールであって、少なくとも階層識別が付加された二次的文からなる前記二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールから表形式のフローチャートを作成するフローチャート作成手段を備えており、

前記フローチャート作成手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、前記二次的モジュールの前記二次的文のうち前記階層識別によって示される階層が同一である前記二次的文について、前記二次的条件式が成立した場合に実行される前記二次的文と前記二次的条件式が成立しない場合に実行される前記二次的文とを表の左右に並べた前記フローチャートを作成するフローチャート作成機能と、前記フローチャートを前記記憶手段に記憶する書込機能とを有している

モジュールの構造解析を支援する装置。

【請求項 8】

請求項 4 又は請求項 5 記載の装置によって出力される二次的モジュールであって、少なくともオペランド情報が付加された二次的文からなる前記二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールの前記二次的文に所定の表示属性を設定する表示属性設定手段を備えており、

前記表示属性設定手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、初期検索文字を取得する初期検索文字取得機能と、前記二次的モジュールの前記二次的文のうち前記オペランド情報の前記オペランドに前記初期検索文字を含んでいるものの前記オペランドを追加検索文字とし、前記二次的文のうち前記オペランド情報の前記オペランドに前記追加検索文字を含んでいるものの前記オペランドを更に前記追加検索文字として、前記二次的文のうち前記オペランド情報の前記オペランドに前記初期検索文字又は前記追加検索文字を含んでいるものに前記所定の表示属性を設定する表示属性設定機能と、前記二次的文に前記所定の表示属性が設定された前記二次的モジュールを前記記憶手段に記憶する書込機能とを有している

モジュールの構造解析を支援する装置。

【請求項 9】

請求項 3 記載の装置によって出力される二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールから複数の列と 1 以上の行とを組み合わせた表形式のマトリックスを作成するマトリックス作成手段を備えており、

、

10

20

30

40

50

前記マトリックス作成手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、前記二次的モジュールの前記二次的文を前記マトリックスの所定の列の各行に出力する二次的文出力機能と、前記二次的モジュールの1以上の前記二次的条件式について、その全てが成立する場合と、その一部が成立する場合と、その全てが成立しない場合とを含む全てのケースを生成するケース生成機能と、生成した前記ケースの夫々に前記列のいずれかを関連表示列として対応させ、前記行に出力された前記二次的文のうち前記関連表示列と対応する前記ケースにおいて実行対象となるものが出力されている前記行に、実行対象となることを示す表示情報を出力するマトリックス作成機能と、前記マトリックスを前記記憶手段に記憶する書込機能とを有しているモジュールの構造解析を支援する装置。

10

【請求項10】

請求項9記載の装置であって、

前記マトリックス作成手段は、前記マトリックス作成機能において、前記記憶手段に記憶された前記二次的モジュールの前記二次的文の一部を選択し、前記関連表示列のうち前記選択された前記二次的文が実行対象となる前記ケースに対応するもののみを前記記憶対象として抽出するモジュールの構造解析を支援する装置。

【請求項11】

コンピュータを、請求項1乃至請求項10のいずれかに記載の装置として機能させるためのプログラム。

20

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、プログラム言語によって記述された1以上の文からなるモジュール(ソースプログラム)の構造解析を支援する装置に関する。

【背景技術】

【0002】

プログラムを作成する場合、構造化プログラミングが強く推奨される。具体的には、例えば、比較的少数の文からなる処理単位を組み合わせることでモジュールを作成するように求められる。また、複数のモジュールについての共通的な処理もモジュール化することが求められる。従って、例えばプログラムの合計行数が100万行を越すような大きなシステムは、夫々が数十の処理単位からなる数千のモジュールを備えていることも多い。

30

【0003】

上述のような大規模なシステムは、開発後しばらくの期間においては、開発に携わった担当者、即ちモジュール間の関連及び夫々のモジュールの構造を熟知した者がシステムのメンテナンスを行うことが多い。しかしながら、このようなシステムは数十年に亘ってメンテナンスを繰り返しながら使用されることも多い。時が経つにつれて開発に携わった担当者がいなくなり、モジュールの構造を知らない者がシステムのメンテナンスを行うことになる。このような場合、構造化プログラミングがなされていることに起因して、膨大なモジュールに分散して記述されている機能を把握しなければならず、メンテナンスは困難を極める。即ち、モジュールの構造解析が困難になる。

40

【0004】

例えば、特許文献1及び特許文献2には、モジュールの構造解析を支援するための技術が開示されている。特許文献1には、モジュール関連図を自動的に作成するモジュール構成図自動生成装置が開示されている。特許文献2には、フローチャート及びテスト項目を自動的に作成するテスト項目自動生成装置が開示されている。

【先行技術文献】

【特許文献】

【0005】

【特許文献1】特開平11-110198号公報

50

【特許文献2】特開2006-209521号公報

【発明の概要】

【発明が解決しようとする課題】

【0006】

しかしながら、システムのメンテナンスを目的としてモジュールの構造解析を行う場合、各モジュールの詳細な内部処理を把握する必要がある。モジュール関連図やモジュール毎のフローチャートを利用して、多数のモジュールに分散された機能の全体像を把握することは容易ではない。

【0007】

そこで、本発明は、多数のモジュールに分散された機能を1つの二次的モジュールに纏めることによって機能を把握しやすくし、これによってモジュールの構造解析を支援する装置を提供することを目的とする。

【課題を解決するための手段】

【0008】

本発明は、第1の装置として、

プログラム言語によって記述された1以上の文からなるモジュールを1以上の二次的文からなる二次的モジュールに変換する装置であって、

前記モジュールはモジュール名によって特定可能であり、前記モジュールには、1以上の前記文から構成される処理単位であって処理単位名によって特定可能な前記処理単位を記述することが可能であり、

前記モジュールには、前記文として、前記処理単位を前記処理単位名によって特定して実行するための処理単位実行文と、前記モジュールを前記モジュール名によって特定して実行するためのモジュール呼び出し文と、条件式が成立するか成立しないかによって実行すべき前記文を選択するための条件文とを記述可能であり、前記条件文によって選択される前記文として更に前記条件文が記述可能であり、これにより前記条件文による前記文の階層化が可能であり、

前記装置は、1以上の前記モジュールが記憶された記憶手段と、前記記憶手段に記憶された前記モジュールのうちの所定の対象モジュールを前記二次的モジュールに変換して前記記憶手段に記憶する変換手段とを備えており、

前記変換手段は、前記対象モジュールを前記記憶手段から読み込む読込機能と、前記対象モジュールの所定の処理開始箇所から所定の処理終了箇所までに記述された前記文の夫々を前記二次的文として前記二次的モジュールに順次出力すると共に、前記文が前記処理単位実行文又は前記モジュール呼び出し文である場合、実行対象の前記処理単位又は呼び出し対象の前記モジュールに記述されている前記文を展開することで前記二次的文を生成して前記二次的モジュールに出力する変換機能であって、前記展開される前記文における前記処理単位実行文又は前記モジュール呼び出し文についても前記展開を更に行う前記変換機能と、前記二次的モジュールを前記記憶手段に記憶する書込機能とを有しているモジュールの構造解析を支援する装置を提供する。

【0009】

本発明は、第2の装置として、第1の装置であって、

前記変換手段は、展開制御情報を取得する制御情報取得機能を有しており、前記展開制御情報は、実行対象の前記処理単位又は呼び出し対象の前記モジュールに記述されている前記文の前記展開を初回のみ制限することを指示する展開制限指示を指定することが可能なものであり、

前記変換手段は、前記変換機能において、前記展開制御情報に前記展開制限指示が指定されている場合、同一の前記処理単位については、実行対象の前記処理単位に記述されている前記文の前記展開を初回の1回のみ行い、同一の前記モジュールについては、呼び出し対象の前記モジュールに記述されている前記文の前記展開を初回の1回のみ行うモジュールの構造解析を支援する装置を提供する。

【0010】

10

20

30

40

50

本発明は、第3の装置として、第1又は第2の装置であって、

前記変換手段は、前記変換機能において、前記文が前記条件文である場合、前記条件文の前記条件式に基づいて前記二次的文の一つである二次的条件文を生成し、前記二次的条件文によって前記二次的文を階層化して前記二次的モジュールに出力するものであり、

前記二次的条件文は、2つの評価対象オペランド間の関係を1つの論理演算子のみによって評価する二次的条件式が成立するか成立しないかによって実行すべき前記二次的文を選択するものである

モジュールの構造解析を支援する装置を提供する。

【0011】

本発明は、第4の装置として、第3の装置であって、

前記変換手段は、前記二次的モジュールに出力する前記二次的文の少なくとも一部について、前記二次的文がいずれの前記階層に位置しているかを示す階層識別を付加する階層識別付加機能と、前記二次的モジュールに出力する前記二次的文のうち定数又は変数からなる2つのオペランドと1つの演算子とを含むものについて、2つの前記オペランドと1つの前記演算子とからなるオペランド情報を前記二次的文に付加するオペランド付加機能とのうちの少なくとも一方を有している

モジュールの構造解析を支援する装置を提供する。

【0012】

本発明は、第5の装置として、第3の装置によって出力される二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールを加工する加工手段を備えており、

前記加工手段は、読込機能と、階層識別付加機能とオペランド付加機能とのうちの少なくとも一方と、書込機能とを有しており、

前記読込機能は、前記二次的モジュールを前記記憶手段から読み込む機能であり、

前記階層識別付加機能は、前記二次的モジュールの前記二次的文の少なくとも一部について、前記二次的文がいずれの前記階層に位置しているかを示す階層識別を付加する機能であり、

前記オペランド付加機能は、前記二次的モジュールの前記二次的文のうち定数又は変数からなる2つのオペランドと1つの演算子とを含むものについて、2つの前記オペランドと1つの前記演算子とからなるオペランド情報を前記二次的文に付加する機能であり、

前記書込機能は、前記階層識別又は前記オペランド情報が付加された前記二次的文からなる前記二次的モジュールを前記記憶手段に記憶する機能である

モジュールの構造解析を支援する装置を提供する。

【0013】

本発明は、第6の装置として、第4又は第5の装置によって出力される二次的モジュールであって、少なくとも階層識別が付加された二次的文からなる前記二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールの前記二次的文の一部を選択する選択手段を備えており、

前記選択手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、前記二次的文のうち選択対象とすべきものの前記階層識別を特定する階層識別特定情報を取得する特定情報取得機能と、前記二次的モジュールの前記二次的文のうち前記階層識別が前記階層識別特定情報によって特定される前記階層識別と同一であるものを選択して選択後の二次的モジュールに出力する選択機能と、前記選択後の二次的モジュールを前記記憶手段に記憶する書込機能とを有している

モジュールの構造解析を支援する装置を提供する。

【0014】

本発明は、第7の装置として、請求項4又は請求項5記載の装置によって出力される二次的モジュールであって、少なくとも階層識別が付加された二次的文からなる前記二次的

10

20

30

40

50

モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールから表形式のフローチャートを作成するフローチャート作成手段を備えており、

前記フローチャート作成手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、前記二次的モジュールの前記二次的文のうち階層識別によって示される階層が同一である前記二次的文について、前記二次的条件式が成立した場合に実行される前記二次的文と前記二次的 condition 式が成立しない場合に実行される前記二次的文とを表の左右に並べた前記フローチャートを作成するフローチャート作成機能と、前記フローチャートを前記記憶手段に記憶する書込機能とを有している

モジュールの構造解析を支援する装置を提供する。

10

【0015】

本発明は、第8の装置として、第4又は第5の装置によって出力される二次的モジュールであって、少なくともオペランド情報が付加された二次的文からなる前記二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールの前記二次的文に所定の表示属性を設定する表示属性設定手段を備えており、

前記表示属性設定手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、初期検索文字を取得する初期検索文字取得機能と、前記二次的モジュールの前記二次的文のうち前記オペランド情報の前記オペランドに前記初期検索文字を含んでいるものの前記オペランドを追加検索文字とし、前記二次的文のうち前記オペランド情報の前記オペランドに前記追加検索文字を含んでいるものの前記オペランドを更に前記追加検索文字として、前記二次的文のうち前記オペランド情報の前記オペランドに前記初期検索文字又は前記追加検索文字を含んでいるものに前記所定の表示属性を設定する表示属性設定機能と、前記二次的文に前記所定の表示属性が設定された前記二次的モジュールを前記記憶手段に記憶する書込機能とを有している

20

モジュールの構造解析を支援する装置を提供する。

【0016】

本発明は、第9の装置として、第3の装置によって出力される二次的モジュールが記憶された記憶手段を備える装置であって、

前記装置は、前記記憶手段に記憶された前記二次的モジュールから複数の列と1以上の行とを組み合わせた表形式のマトリックスを作成するマトリックス作成手段を備えており、

30

前記マトリックス作成手段は、前記二次的モジュールを前記記憶手段から読み込む読込機能と、前記二次的モジュールの前記二次的文を前記マトリックスの所定の列の各行に出力する二次的文出力機能と、前記二次的モジュールの1以上の前記二次的 condition 式について、その全てが成立する場合と、その一部が成立する場合と、その全てが成立しない場合とを含む全てのケースを生成するケース生成機能と、生成した前記ケースの夫々に前記列のいずれかを関連表示列として対応させ、前記行に出力された前記二次的文のうち前記関連表示列と対応する前記ケースにおいて実行対象となるものが出力されている前記行に、実行対象となることを示す表示情報を出力するマトリックス作成機能と、前記マトリックスを前記記憶手段に記憶する書込機能とを有している

40

モジュールの構造解析を支援する装置を提供する。

【0017】

本発明は、第10の装置として、第9の装置であって、

前記マトリックス作成手段は、前記マトリックス作成機能において、前記記憶手段に記憶された前記二次的モジュールの前記二次的文の一部を選択し、前記関連表示列のうち前記選択された前記二次的文が実行対象となる前記ケースに対応するもののみを前記記憶対象として抽出する

モジュールの構造解析を支援する装置を提供する。

【0018】

50

本発明は、第1のプログラムとして、コンピュータを、第1乃至第10のいずれかの装置として機能させるためのプログラムを提供する。

【発明の効果】

【0019】

本発明によれば、1つのモジュールの所定の処理開始箇所から所定の処理終了箇所までの間に実行対象の文が展開されるため、多数のモジュールに分散された機能を容易に把握することが可能になる。

【0020】

また、同じ処理単位又は同じモジュールについて実行対象の文を1回のみ展開することにすれば重複した展開を避けることができ、これにより展開される量を小さくすることができる。

10

【図面の簡単な説明】

【0021】

【図1】本発明の実施の形態によるモジュールの構造解析を支援する装置を示すブロック構成図である。

【図2】図1の装置の記憶装置に記憶されているモジュールの模式的なファイル構成図である。

【図3】図2のモジュールを例示する図である。

【図4】図1の記憶装置に記憶されている二次的モジュールの模式的なファイル構成図である。

20

【図5】図1の二次的モジュールを部分的に例示する図である。なお、二次的モジュールの一部（破線Aの部分）を実線Aの部分に拡大して示している。

【図6】図1の記憶装置に記憶されているフローチャートを部分的に例示する図である。

【図7】図1の記憶装置に記憶されているマトリックスを部分的に例示する図である。

【図8】図1の装置の変換手段の機能を示すフローチャートである。

【図9】図8の変換手段の機能の一部をより詳細に示すフローチャートである。

【図10】図1の装置の加工手段の機能を示すフローチャートである。

【図11】図1の装置の選択手段の機能を示すフローチャートである。

【図12】図1の装置の表示属性設定手段の機能を示すフローチャートである。

【図13】図1の装置のフローチャート作成手段の機能を示すフローチャートである。

30

【図14】図1の装置のマトリックス作成手段の機能を示すフローチャートである。

【図15】図1の記憶装置に記憶されているマトリックスを部分的に例示する図である。

【図16】図3のモジュールのうちの一つを変更した際の差分を例示する図である。

【図17】図14のマトリックス作成手段の機能の変形例を示すフローチャートである。

【図18】図16のモジュールが変更された場合のマトリックスを部分的に例示する図である。

【図19】図16のモジュールが変更された場合のフローチャートを部分的に例示する図である。

【図20】図1の装置の変形例を示すシステム構成図である。

【発明を実施するための形態】

40

【0022】

図1に示されるように、本実施の形態による装置（モジュールの構造解析を支援する装置）10は、装置本体20と、記憶装置（記憶手段）30と、入力装置40と、表示装置50とを備えている。装置10は、プログラム言語によって記述された1以上の文からなるモジュール（ソースプログラム）を、1以上の二次的文からなる二次的モジュール（二次的ソースプログラム）に変換するものである。

【0023】

本実施の形態によるプログラム言語はCOBOLである。しかしながら、本発明はCOBOL以外のプログラム言語にも適用可能である。

【0024】

50

本実施の形態によるモジュールは、文字、数字、記号等を組み合わせたモジュール名によって特定可能である。

【 0 0 2 5 】

モジュールには、1以上の文から構成される処理単位を記述することができる。本実施の形態による処理単位は、COBOLにおけるセクションであり、セクションの夫々は、セクション名（処理単位名）によって特定することができる。なお、他のプログラム言語における処理単位は、例えば関数名（処理単位名）によって特定可能な関数である。

【 0 0 2 6 】

モジュールは、他のモジュール又は当該モジュール自身を呼び出すことができる。また、モジュールのセクションは、他のセクション又は当該セクション自身を実行することができる。より具体的には、モジュールには、文として、セクションをセクション名によって特定して実行するための処理単位実行文と、モジュールの一つをモジュール名によって特定して実行するためのモジュール呼び出し文とを記述することができる。また、モジュールには、文として、条件式が成立するか成立しないかによって実行すべき文を選択するための条件文を記述することができる。なお、他のプログラム言語における処理単位実行文及びモジュール呼び出し文は、例えば関数名を指定した文である。モジュール呼び出し文が関数名を指定した文である場合、モジュール呼び出し文によって実行されるのはモジュールのうちの指定された関数部分（即ち、モジュールの一部）である。また、この場合、モジュール呼び出し文においてモジュール名が明示的に記述されない場合（即ち、関数名によってモジュール名が間接的に特定される場合）がある。

【 0 0 2 7 】

本実施の形態においては、処理単位実行文はPERFORM文であり、モジュール呼び出し文はCALL文である。また、条件文はIF文及びEVALUATE文である。IF文等の条件文によって選択される文として更に条件文が記述可能であり、これにより条件文による文の階層化が可能である。詳しくは、例えばIF文のTHENについて更にIF文を記述することができる。

【 0 0 2 8 】

図1に示されるように、記憶装置30は、例えば磁気ディスク装置であり、フォルダ（図示せず）等によって階層化された様々なファイルを記憶することができる。記憶装置30は、装置本体20と通信可能に接続されており、装置本体20からの指示に従ってファイルの読込、書込（記憶）等を行う。本実施の形態による記憶装置30には、1以上のモジュール31が記憶されている。換言すれば、本実施の形態によるモジュール31は、記憶装置30に記憶されたモジュールである。モジュール31は、例えばプログラミングツール（図示せず）を使用して作成される。モジュール31の夫々が一つのファイルであってもよいし、複数のモジュール31が一つのファイルとして記憶されていてもよい。記憶装置30は、二次的モジュール32と、フローチャート33と、マトリックス34とを、モジュール31と同様に記憶することができる。本実施の形態によるフローチャート33及びマトリックス34は、二次的モジュール32から夫々作成することができる。

【 0 0 2 9 】

入力装置40は、例えばキーボードやマウスである。入力装置40は、装置本体20と通信可能に接続されており、キーボードから入力された文字やマウスによって指示された位置を装置本体20に通知する。

【 0 0 3 0 】

表示装置50は、例えばディスプレイやプリンタである。表示装置50は、装置本体20と通信可能に接続されており、装置本体20から転送された文字や図形等を表示又は印刷する。

【 0 0 3 1 】

装置本体20は、例えばPC（Personal Computer）の本体である。装置本体20は、CPU（Central Processing Unit）、主記憶装置等を備えている（図示せず）。記憶装置30には様々なプログラムの実行形式ファイルが記憶されている（図示せず）。CPUは、実行形式ファ

10

20

30

40

50

イルを主記憶装置にローディングし、実行形式ファイル内の命令語を実行することで様々な機能を実現する。またCPUが命令語を実行する過程において、記憶装置30に記憶された所定のファイルが、主記憶装置内のバッファ領域を経由してプログラムの変数領域に読み込まれる。また、プログラムの変数領域に読み込まれたファイルの内容は、必要に応じて主記憶装置内のバッファ領域を経由して記憶装置30に書き込まれる。上記の説明から理解されるように、記憶装置30と主記憶装置とは、本実施の形態における読み込み及び書き込み可能な記憶手段として夫々機能する。以降の説明における記憶手段としては記憶装置30を例示する。

【0032】

本実施の形態による記憶装置30には、変換プログラムと、加工プログラムと、選択プログラムと、表示属性設定プログラムと、フローチャート作成プログラムと、マトリックス作成プログラムとが記憶されている(図示せず)。上記のプログラムは、主記憶装置にローディングされ、CPUによって実行され、これによってコンピュータは、変換手段21、加工手段22、選択手段23、表示属性設定手段24、フローチャート作成手段25、及びマトリックス作成手段26として夫々機能する。即ち、上記のプログラムは、コンピュータをモジュールの構造解析を支援する装置10として機能させるためのものである。換言すれば、本実施の形態による装置10の装置本体20は、変換手段21と、加工手段22と、選択手段23と、表示属性設定手段24と、フローチャート作成手段25と、マトリックス作成手段26とを備えている。

【0033】

本実施の形態による変換手段21は、記憶装置30に記憶されたモジュール31のうち所定の対象モジュール31を二次的モジュール32に変換して記憶装置30に記憶するものである。加工手段22は、記憶装置30に記憶された二次的モジュール32を加工するものである。選択手段23は、記憶装置30に記憶された二次的モジュール32の二次的文の一部を選択するものである。表示属性設定手段24は、記憶装置30に記憶された二次的モジュール32の二次的文に所定の表示属性を設定するものである。フローチャート作成手段25は、記憶装置30に記憶された二次的モジュール32から表形式のフローチャート33を作成するものである。マトリックス作成手段26は、記憶装置30に記憶された二次的モジュール32から複数の列と1以上の行とを組み合わせた表形式のマトリックス34を作成するものである。

【0034】

図2に示されるように、本実施の形態によるモジュール31は、 n 個($n \geq 1$)の行310から構成される。行310は、一連番号(行識別)311と文312とから構成される。一連番号311は、行310を識別するためのものであり、数字だけでなく文字等を含むことができる。文312には、文(データ定義文、実行文、コメント文等)が記述されている。文312には、一つの完結した文全体が記述されていてもよいし、一つの文の一部が記述されていてもよい。例えば、MOVE文が2つの行310の文312に分けて記述されていてもよい。

【0035】

図3に示されるように、記憶装置30には、例えばモジュール名がXXXであるモジュール31(XXXモジュール)と、モジュール名がAAAであるモジュール31(AAAモジュール)と、モジュール名がBBBであるモジュール31(BBBモジュール)と、モジュール名がCCCであるモジュール31(CCCモジュール)とが記憶されている。モジュール31の夫々は、一連番号311と文312とからなる複数の行310から構成されている。

【0036】

XXXモジュールはメインモジュール(即ち、処理の起点となるモジュール)であり、AAAモジュールとBBBモジュールとをモジュール名によって特定して呼び出すためのCALL文が夫々記述された行310を有している。AAAモジュールは、CCCモジュールを呼び出すCALL文が記述された行310を有している。

【 0 0 3 7 】

図 3 に例示されたモジュール 3 1 の夫々は、1 以上のセクションから構成されている。例えば、X X X モジュールは、" PROCEDURE DIVISION." が記述された行 3 1 0 の次の行から、" STOP RUN." が記述された行 3 1 0 までのメインセクション（モジュールにおける処理の起点となる主たる処理単位）と、" AAA-SUB SECTION." が記述された行 3 1 0 から、" EXIT." が記述された行 3 1 0 までの従たるセクション（従たる処理単位）を有している。なお、上述の従たるセクションのセクション名（処理単位名）は A A A - S U B であり、メインセクションは、A A A - S U B セクションをセクション名によって特定して実行する行 3 1 0 を備えている。より具体的には、メインセクションは、" PERFORM AAA-SUB." が記述された行 3 1 0 を備えている。

10

【 0 0 3 8 】

図 3 に例示されたセクションの一部は、条件文が記述された行 3 1 0 を備えている。例えば、X X X モジュールのメインセクションは、5 つの WHEN を有する EVALUATE 文が記述された行 3 1 0 と、THEN および ELSE を有する IF 文が記述された行 3 1 0 とを備えている。また、C C C モジュールのメインセクションは、IF 文の ELSE において更に IF 文が記述された行 3 1 0 を備えており、これによって条件文による文 3 1 2 の階層化が行われている。

【 0 0 3 9 】

図 4 に示されるように、本実施の形態による二次的モジュール 3 2 は、m 個（m 1）の行 3 2 0 から構成される。本実施の形態においては、二次的モジュール 3 2 は、MICROSOFT EXCEL（登録商標）のワークシートであるが、テキストファイル等であってもよい。行 3 2 0 は、モジュール識別 3 2 1 と、行番号 3 2 2 と、階層識別 3 2 3 と、命令識別 3 2 4 と、オペランド情報 3 2 5 と、一連番号（行識別）3 2 6 と、二次的文 3 2 7 と、表示属性 3 2 8 とから構成される。オペランド情報 3 2 5 は、第 1 オペランド 3 2 5 f と、演算子 3 2 5 o と、第 2 オペランド 3 2 5 s とを備えている。二次的モジュール 3 2 は、1 以上のモジュール 3 1 から実行文等が記述された行 3 1 0 を集めて加工することによって生成されている。

20

【 0 0 4 0 】

図 5 に示されるように、本実施の形態による二次的モジュール 3 2 は、例えば、X X X モジュール、A A A モジュール、B B B モジュール及び C C C モジュールの PROCEDURE DIVISION の行 3 1 0 を集めて加工したものである。

30

【 0 0 4 1 】

図 5 に例示されているように、行 3 2 0 の二次的文 3 2 7 において、行 3 1 0 の文 3 1 2 に記述されている CALL 文はコメント化されて（コメント文に変更されて）いる。また、CALL 文によって呼び出される A A A モジュールの PROCEDURE DIVISION の文 3 1 2 が、行 3 2 0 の二次的文 3 2 7 として展開されている。更に、A A A モジュールから CALL 文によって呼び出される C C C モジュールの PROCEDURE DIVISION の文 3 1 2 が、行 3 2 0 の二次的文 3 2 7 として展開されている。同様に、行 3 1 0 の文 3 1 2 に記述されている PERFORM 文は、行 3 2 0 の二次的文 3 2 7 においてコメント化されている。また、PERFORM 文によって実行される A A A - S U B セクションの文 3 1 2 が、行 3 2 0 の二次的文 3 2 7 として展開されている。

40

【 0 0 4 2 】

図 5 に例示されているように、文 3 1 2 に記述された条件文のうち IF 文以外のもの（本実施の形態においては EVALUATE 文）は、二次的文 3 2 7 において IF 文に変換されている。例えば、" EVALUATE WK-A-0 "、" WHEN CS-1 "、" MOVE CS-1 TO WK-B-1 " が夫々記述された 3 つの文 3 1 2 は、" *#E EVALUATE WK-A-0 "（コメント文）、" *#E WHEN CS-1 "（コメント文）、" IF (WK-A-0 = CS-1) "、" THEN "、" MOVE

50

CS-1 TO WK-B-1” が夫々記述された5つの二次的文3 2 7に変換される。上記の変換に加え、IF文におけるTHEN、ELSE及びEND IFの欠落を補うための二次的文3 2 7が生成される。また、否定の条件式は肯定の条件式に変換される。更に、ANDやORで結合された条件式を有するIF文は複数のIF文に分解されて、2つの評価対象オペランド間の関係を1つの論理演算子のみによって評価する単純な二次的条件式を有するIF文に変換される。

【0043】

以上に加えて、複数の文3 1 2に分かち書きされているMOVE文等は、1つの二次的文3 2 7に纏められている。即ち、二次的モジュール3 2 の行3 2 0の夫々は、モジュール3 1 の行3 1 0に基づいて生成されており、二次的文3 2 7は、文3 1 2をそのままコピーしたものであるか、又は文3 1 2を加工・展開することによって生成されたものである。

10

【0044】

表示属性3 2 8には、行3 2 0の二次的文3 2 7等を表示装置5 0に表示する際の表示色、背景色、書体等、表示に関する属性が設定される。例えば、文3 1 2をコメント化した二次的文3 2 7を赤色で表示することで、コメント化したことを視覚的に把握することが可能になる。

【0045】

本実施の形態によるモジュール識別3 2 1は、夫々の行3 2 0がいずれのモジュール3 1に基づくものであるかを示すものである。より具体的には、モジュール識別3 2 1には、XXX等のモジュール名が設定される。なお、モジュール名に代えて、モジュール3 1を一意に識別可能な番号等を設定してもよい。

20

【0046】

一連番号3 2 6には、二次的文3 2 7が文3 1 2のコピーである行3 2 0においては、コピー元の行3 1 0の一連番号3 1 1が設定される。また、二次的文3 2 7が文3 1 2をコメント化したものである等、行3 1 0と1対1に対応する行3 2 0については、一連番号3 2 6には、対応する行3 1 0の一連番号3 1 1が設定される。一方、生成された(追加された)行3 2 0の一連番号3 2 6には、生成された行3 2 0であることを識別可能な記号等が設定される。このようにすることで、システムのメンテナンスを行う際に、行3 2 0と行3 1 0を容易に対応付けることができる。

【0047】

本実施の形態による行番号3 2 2には、1から始まる通番が行3 2 0の並び順に沿って設定されている。しかしながら、通番である必要はなく、行番号3 2 2に記号等を設定してもよい。

30

【0048】

本実施の形態による階層識別3 2 3は、分岐階層と分岐の深さとから構成されている。分岐階層は、条件文による階層を識別するためのものである。本実施の形態による分岐階層は、1を1行目の行3 2 0における出発値としており、IF文のTHENが記述された行3 2 0において1 . 1に変わり、ELSEが記述された行3 2 0において1 . 2に変わる。そしてIF文のEND IFが記述された行3 2 0の次の行から1に戻る。一般化すると、THENが記述された行3 2 0においては、直前の行3 2 0の分岐階層の末尾に「. 1」が加えられ、ELSEが記述された行3 2 0においては、直前の行3 2 0の分岐階層の末尾「. 1」が「. 2」に変更され、END IFが記述された行3 2 0の次の行においては、END IFが記述された行3 2 0の分岐階層の末尾「. 2」が削除されている。上記の場合を除き、分岐階層には、直前の行3 2 0と同一の値が設定されている。

40

【0049】

本実施の形態による分岐の深さは、1を1行目の行3 2 0における出発値としており、IF文のTHENが記述された行3 2 0において1 . 1に変わり、END IFが記述された行3 2 0の次の行から1に戻る。また、その後のIF文のTHENが記述された行3 2 0において1 . 2に変わり、END IFが記述された行3 2 0の次の行から1に戻る。一般化すると、分岐の深さは、1を出発値としており、THENが記述された行3 2 0において末尾に「. p」(pは1以上の整数)なる桁が加えられ、END

50

IFが記述された行320の次の行において末尾の「.p」なる桁が削除される。「.p」におけるpの値は、桁の夫々について1を出発値としている。詳しくは、1桁目の「.p」（例えば1.2の「.2」）におけるpの値は、1を出発値としており、付加される毎にインクリメントされる。一方、2桁目以降の「.p」（例えば1.2.1の「.1」）におけるpの値は、当該桁を除いた分岐の深さが同一値である行320が連続する範囲において1を出発値としており、当該連続する範囲において付加される毎にインクリメントされる。

【0050】

以上の説明から理解されるように、階層識別323によって、条件文による階層の二次的モジュール32における場所を特定することができる。なお、階層識別323の設定内容は上述の説明には限られない。条件文による階層の二次的モジュール32における場所を特定することができる限り、階層識別323にどのような値が設定されていてもよい。

10

【0051】

命令識別324は、二次的文327に記述されている文の種別を示すものである。本実施の形態による命令識別324は、命令名と、条件命令であるか否かを示す区分（条件命令当否）と、処理命令であるか否かを示す区分（処理命令当否）とからなる。二次的文327にIF文の一部が記述されている場合（IF、THEN、ELSE又はEND-IFが記述されている場合）、条件命令当否には が設定される。また、二次的文327にIF文以外の実行文が記述されている場合（例えばMOVE文やCOMPUTE文が記述されている場合）、処理命令当否には が設定される。条件命令当否又は処理命令当否に が設定されている場合、命令名には二次的文327に記述されている命令名（IF、THEN、END-IF、MOVE等）が設定される。

20

【0052】

オペランド情報325は、命令識別324に命令名が設定された二次的文327のうち（即ち、実行文が記述されている二次的文327のうち）2つのオペランドと1つの演算子とからなる二次的文327について設定される。例えば、「MOVE 9 TO WK-C-1」が記述されている二次的文327については、第1オペランド325f、演算子325o及び第1オペランド325sに、9、TO及びWK-C-Iが夫々設定される。オペランド情報325を参照することによって、変数および定数間の代入・比較がどのように行われているかを、より容易に把握することができる。

30

【0053】

なお、行320は、以上に説明した項目に加えて様々な項目を備えることができる。例えば、対応する行310のセクション名を備えることにより、行320と行310との対応を、より容易に把握することができる。

【0054】

本実施の形態によれば、記憶装置30に記憶された二次的モジュール32に基づいて様々なファイルを作成し、作成したファイルをモジュール31の構文解析を支援するための資料として表示装置50に表示・印刷することができる。具体的には、例えば二次的モジュール32に基づいてフローチャート33及びマトリックス34を作成することができる。

【0055】

図6及び図7から理解されるように、本実施の形態によるフローチャート33及びマトリックス34は、MICROSOFT EXCEL（登録商標）のワークシートであり、二次的モジュール32から生成される。但し、フローチャート33及びマトリックス34は他の形式のファイル等であっても構わない。

40

【0056】

図6から理解されるように、フローチャート33は、二次的モジュール32の二次的文327をフローチャート風に編集したものである。より具体的には、IF文と関連しない二次的文327は、二次的モジュール32における行320の並び順に沿って、第2列目に並べられている。一方、二次的文327のうち、IF文と関連するもの（即ち、IFからEND-IFまでの文が記述されたもの）については、IF文における条件式が成立する場合に実行さ

50

れる二次的文 3 2 7 と、IF文における条件式が成立しない場合に実行される二次的文 3 2 7 とが左右の列に並べられている。

【 0 0 5 7 】

フローチャート 3 3 の第 1 列目には第 2 列目に並べられた二次的文 3 2 7 の行番号 3 2 2 が設定されており、これによりフローチャート 3 3 の行を二次的モジュール 3 2 の行 3 2 0 に夫々対応付けることができる。なお、行番号 3 2 2 に限らず、行 3 2 0 の任意の項目をフローチャート 3 3 に設定することができる。

【 0 0 5 8 】

以上の説明から理解されるように、本実施の形態によるフローチャート 3 3 は、IF文による階層が深くなるにつれて列が増加する。換言すれば、二次的モジュール 3 2 のうち、より複雑な条件文が記述されている箇所は、表の列方向に延びるようにして表示される。従って、フローチャート 3 3 を表示装置 5 0 に表示させ、左右の列に並べて表示された実行文を見比べることで、条件式の成否による処理の相違を把握することができる。フローチャート 3 3 における処理の塊を色付けして表示装置 5 0 に表示させることもできる。例えば、フローチャート 3 3 のIF及びEND-IFが設定された欄に所定の背景色 A を設定し、THENの場合に実行される二次的文 3 2 7 に所定の背景色 B を設定し、ELSEの場合に実行される二次的文 3 2 7 に所定の背景色 C を設定することができる。このようにすることで、フローチャート 3 3 を表示装置 5 0 に表示させた際に、IF文による階層構造を背景色の組み合わせによるパターンとして視覚的に把握することが可能になる。

【 0 0 5 9 】

図 7 に示されるように、本実施の形態によるマトリックス 3 4 は、行番号の列（第 1 列目）と、一連番号の列（第 2 列目）と、二次的文の列（第 3 列目）と、1 以上の関連表示列（第 4 列目以降）とからなる表である。列の夫々は 1 以上の行を備えている。

【 0 0 6 0 】

行番号の列、一連番号の列、及び二次的文の列には、二次的モジュール 3 2 の行番号 3 2 2、一連番号 3 2 6、及び二次的文 3 2 7 が夫々設定され、表示属性 3 2 8 に基づく表示属性が設定される。なお、上記の項目に代えて又は上記の項目に加えて、行 3 2 0 の任意の項目をマトリックス 3 4 に設定することができる。

【 0 0 6 1 】

関連表示列の列数は、二次的モジュール 3 2 の二次的文 3 2 7 に記述されているIF文によって決まる。例えば、THENとELSEとを夫々 1 つだけ有するIF文が 1 つのみ記述されている場合、IF文による分岐数（即ち、IF文において実行されるケースの数）は 2 なので、関連表示列の列数は 2 である。また、例えば、THENとELSEとの夫々にTHENとELSEとを有するIF文が記述されている場合、実行されるケースの数は 4（ 2×2 ）なので、関連表示列の列数は 4 である。例えば、実行されるケースの数が 5 のIF文が 2 つ記述されている場合、関連表示列の列数は、2 つのケースの数の最大値である 5 としてもよいし、夫々のケースを組み合わせた数である 2 5（ 5×5 ）としてもよい。図 7 に例示されている関連表示列の列数は、夫々のIF文において実行されるケースの数の最大値となっている。詳しくは、図 7 のマトリックス 3 4 は、ケース C 1 乃至ケース C 5 の 5 つのケースに対応する 5 列の関連表示列を備えている。

【 0 0 6 2 】

関連表示列には、二次的文の列（第 3 列目）との関連が設定される。より具体的には、二次的文の列に記述されている実行文のうち、関連表示列と対応するケースにおいて実行されるものについて、関連表示列の対応する行に I、T、E、EI 又は が設定される。一方、実行されないものについては - が設定される。ここで、I、T、E、EI 及び は、実行文がIF文のIF、THEN、ELSE、END-IF及びIF文以外の実行文であることを夫々示している。なお、実行文以外の文（例えばコメント文）と対応する行には * が設定される。

【 0 0 6 3 】

以上の説明から理解されるように、本実施の形態によるマトリックス 3 4 を表示装置 5 0 に表示させることにより、二次的文の列に表示される実行文と、関連表示列に表示され

10

20

30

40

50

るケースとの関連を把握することができる。また、メンテナンスにおいて実行文を変更した場合に必要なテストケースを把握することができる。

【 0 0 6 4 】

以下、以上のように構成された装置 1 0 の機能及び動作について詳細に説明する。

【 0 0 6 5 】

図 8 から理解されるように、本実施の形態による変換手段 2 1 は、展開制御情報を取得する制御情報取得機能 (S 8 0 0) と、モジュール 3 1 のうち処理対象とする対象モジュール 3 1 を記憶装置 3 0 から読み込む読込機能 (S 8 0 2、 S 8 0 4) と、対象モジュール 3 1 を二次的モジュール 3 2 に変換する変換機能 (S 8 0 6 ~ S 8 1 2) と、二次的モジュール 3 2 を記憶装置 3 0 に記憶する書込機能 (S 8 1 4) とを有している。

10

【 0 0 6 6 】

より具体的には、図 8 に示されるように、変換手段 2 1 は、例えば入力装置 4 0 から入力された起動指示によって起動され、例えば起動指示の一部として入力された展開制御情報を取得する (S 8 0 0)。ここで、展開制御情報は、展開制限指示を指定することが可能なものであり、展開制限指示は、実行対象のセクション又は呼び出し対象のモジュール 3 1 の文 3 1 2 に記述されている文の展開を初回のみを制限することを指示するものである。

【 0 0 6 7 】

次に、変換手段 2 1 は、対象モジュール 3 1 を特定するための対象モジュール名を取得する (S 8 0 2)。対象モジュール名は、展開制御情報と同様な方法で取得することができる。変換手段 2 1 は、取得した対象モジュール名によって特定される対象モジュール 3 1 を記憶装置 3 0 から読み込む (S 8 0 4)。このとき、例えば所定のフォルダに記憶されたモジュール 3 1 のみを読み込み対象としてもよい。なお対象モジュール 3 1 が記憶装置 3 0 に記憶されていない場合、変換手段 2 1 は、処理を終了する。

20

【 0 0 6 8 】

次に、変換手段 2 1 は、対象モジュール 3 1 の処理開始箇所の文 3 1 2 を取得する (S 8 0 6)。本実施の形態による処理開始箇所は、メインセクションの先頭である。変換手段 2 1 は、取得した文 3 1 2 について二次的文出力処理を行う (S 8 0 8)。次に、変換手段 2 1 は、処理終了箇所の文 3 1 2 を処理済みであるか判定する (S 8 1 0)。本実施の形態による処理終了箇所は、メインセクションの末尾である。処理済みの場合 (S 8 1 0 において Y E S の場合)、変換手段 2 1 は、作成した二次的モジュール 3 2 を記憶装置 3 0 に書き込み (S 8 1 4)、処理を終了する。一方、処理済みでない場合 (S 8 1 0 において N O の場合)、変換手段 2 1 は、次に処理する文 3 1 2 (後続する行 3 1 0 の文 3 1 2) を取得し (S 8 1 2)、取得した文 3 1 2 について二次的文出力処理を行う (S 8 0 8)。

30

【 0 0 6 9 】

図 9 は、図 8 に示した変換手段 2 1 の処理 (機能) のうち、二次的文出力処理 (S 8 0 8) をより詳細に示すフローチャートである。図 9 に示されるように、変換手段 2 1 は、二次的文出力処理において文 3 1 2 を二次的文 3 2 7 として出力する。より具体的には、本実施の形態においては、文 3 1 2 に記述されている実行文等をワークシートの所定の行・列にコピーする。このとき、前述したように、PERFORM文をコメント文にする等の加工が行われる。また、変換手段 2 1 は、モジュール識別 3 2 1 と、行番号 3 2 2 と、一連番号 3 2 6 と、表示属性 3 2 8 とに前述したような値を出力する。

40

【 0 0 7 0 】

次に、変換手段 2 1 は、文 3 1 2 に条件文が記述されているか判定する (S 9 0 2)。文 3 1 2 に記述されている実行文が条件文である場合 (S 9 0 2 で Y E S の場合)、変換手段 2 1 は、前述したように IF 文以外の条件文を IF 文に変換する等の処理を行う。より具体的には、変換手段 2 1 は、条件文の条件式に基づいて、二次的文の一つである二次的条件文 (IF 文) が記述された二次的文 3 2 7 を生成し、二次的条件文によって二次的文 3 2 7 を階層化して二次的モジュールに出力し (S 9 0 4)、二次的文出力処理を終了する。

50

ここで、二次的条件文は、2つの評価対象オペランド間の関係を1つの論理演算子のみによって評価する二次的条件式が成立するか成立しないかによって実行すべき二次的文327を選択するものである。

【0071】

文312に記述されている実行文が条件文でない場合(S902でNOの場合)、変換手段21は、文312に記述されている実行文が処理単位実行文(PERFORM文)であるか判定する(S906)。処理単位実行文である場合(S906でYESの場合)、変換手段21は、実行対象の処理単位(セクション)の処理開始箇所の文312を取得する。本実施の形態においては、処理単位の処理開始箇所はセクションの先頭である。

【0072】

処理単位実行文でない場合(S906でNOの場合)、変換手段21は、文312に記述されている実行文がモジュール呼び出し文(CALL文)であるか判定する(S910)。モジュール呼び出し文である場合(S910でYESの場合)、変換手段21は、CALL文によって特定される呼び出し対象のモジュール31を記憶装置30から読み込み、呼び出し対象のモジュール31の処理開始箇所の文312を取得する。このとき、例えば対象モジュール31が記憶されている所定のフォルダに記憶されたモジュール31のみを読み込み対象としてもよい。本実施の形態においては、呼び出し対象のモジュール31の処理開始箇所は、当該モジュール31のメインセクションの先頭である。一方、モジュール呼び出し文でない場合(S910でNOの場合)、変換手段21は二次的文出力処理を終了する。

【0073】

変換手段21は、実行対象のセクションの文312又は呼び出し対象のモジュール31の文312について、再帰的に二次的文出力処理を行う(S916)。次に、変換手段21は、処理終了箇所の文312を処理済みであるか判定する(S918)。本実施の形態による処理終了箇所は、実行対象のセクションについては当該セクションの末尾であり、呼び出し対象のモジュール31についてはメインセクションの末尾である。処理済みの場合(S918においてYESの場合)、変換手段21は、二次的文出力処理を終了する。一方、処理済みでない場合(S918においてNOの場合)、変換手段21は、次に処理する文312(後続する行310の文312)を取得し(S920)、取得した文312について再帰的に二次的文出力処理を行う(S916)。

【0074】

なお、変換手段21は、再帰的な(自分自身を実行する)PERFORM文については、実行対象のセクションの文312の展開を行わない。同様に、再帰的な(自分自身を呼び出す)CALL文については、呼び出し対象のモジュール31の文312の展開を行わない。更に、変換手段21は、展開制御情報に展開制限指示が指定されている場合、既に二次的文出力処理を行ったセクションとセクション名が同一であるセクションを実行するPERFORM文については、二次的文出力処理を行わない(S906)。同様に、変換手段21は、展開制御情報に展開制限指示が指定されている場合、変換手段21は、既に二次的文出力処理を行ったモジュール31とモジュール名が同一であるモジュール31を呼び出すCALL文については、二次的文出力処理を行わない(S910)。

【0075】

以上の説明から理解されるように、変換手段21の変換機能は、対象モジュール31の所定の処理開始箇所から所定の処理終了箇所までに記述された文312の夫々を二次的文327として二次的モジュールに順次出力する。更に、文312が処理単位実行文又はモジュール呼び出し文である場合、実行対象の前記処理単位又は呼び出し対象のモジュール31に記述されている文312を展開することで二次的文327を生成して二次的モジュール32に出力する。更に、展開される文312における処理単位実行文又はモジュール呼び出し文についても展開を更に行う(即ち、再帰的に展開する)。従って、多数の処理単位やモジュール31に分散された機能を、対象モジュール31のメインセクションに手繰り寄せるようにして集めることができる。なお、モジュール31についての所定の処理

10

20

30

40

50

開始箇所及び所定の処理終了箇所は、夫々、PROCEDURE DIVISIONの先頭及び末尾としてもよい。また、例えば、メインセクションの途中とすることも可能である。但し、不要な展開を避け、且つ十分な展開結果を得るためには本実施の形態のようにするのが好ましい。

【0076】

また、以上の説明から理解されるように、変換手段21の変換機能は、展開制御情報に展開制限指示が指定されている場合、同一の処理単位については、実行対象の処理単位に記述されている文312の展開を初回の1回のみ行い、同一のモジュール31については、呼び出し対象のモジュール31に記述されている文312の展開を初回の1回のみ行う。

【0077】

以上に説明したように、変換手段21によって、二次的文327からなる二次的モジュール32が記憶装置30に記憶される。換言すれば、変換手段21を備えた装置10は、二次的文327からなる二次的モジュール32を記憶装置30に出力することができる。

【0078】

図10から理解されるように、本実施の形態による加工手段22は、読込機能(S1000、S1002)と、階層識別付加機能(S1006)とオペランド付加機能(S1010)とのうちの少なくとも一方と、書込機能(S1014)とを有している。読込機能は、処理対象の二次的モジュール32を記憶装置30から読み込む機能である。階層識別付加機能は、二次的モジュール32の二次的文327の少なくとも一部について、二次的文327がいずれの階層に位置しているかを示す階層識別323を付加する機能である。オペランド付加機能は、二次的モジュール32の二次的文327のうち定数又は変数からなる2つのオペランドと1つの演算子とを含むものについて、2つのオペランド(第1オペランド325fと第2オペランド325s)と1つの演算子325oとからなるオペランド情報325を二次的文327に付加する機能である。書込機能は、階層識別323又はオペランド情報325が付加された二次的文327からなる二次的モジュール32を記憶装置30に記憶する機能である。

【0079】

より具体的には、図10に示されるように、加工手段22は、例えば変換手段21と同様に起動され、処理対象の二次的モジュール32を特定するための二次的モジュール名を取得する(S1000)。加工手段22は、取得した二次的モジュール名によって特定される二次的モジュール32を記憶装置30から読み込む(S1002)。なお処理対象の二次的モジュール32が記憶装置30に記憶されていない場合、加工手段22は、処理を終了する。

【0080】

次に、加工手段22は、処理対象の二次的モジュール32の処理開始箇所の二次的文327を取得する(S1004)。本実施の形態による処理開始箇所は、二次的モジュール32の先頭である。

【0081】

加工手段22は、取得した二次的文327について階層識別323を付加する(S1006)。より具体的には、本実施の形態による加工手段22は、ワークシートにおいて行320の階層識別323と対応する行・列に、前述したような分岐階層と分岐の深さとを設定する。次に、加工手段22は、取得した二次的文327について命令識別324を付加する(S1008)。より具体的には、本実施の形態による加工手段22は、ワークシートにおいて行320の命令識別324と対応する行・列に、前述したような命令名と、条件命令当否と、処理命令当否とを設定する。次に、加工手段22は、取得した二次的文327についてオペランド情報325を付加する(S1010)。より具体的には、本実施の形態による加工手段22は、ワークシートにおいて行320のオペランド情報325と対応する行・列に、前述したような第1オペランド325fと、演算子325oと、第2オペランド325sとを設定する。

【0082】

10

20

30

40

50

次に、加工手段 2 2 は、処理終了箇所の二次的文 3 2 7 を処理済みであるか判定する (S 1 0 1 2)。本実施の形態による処理終了箇所は、二次的モジュール 3 2 の末尾である。処理済みの場合 (S 1 0 1 2 において Y E S の場合)、加工手段 2 2 は、加工した二次的モジュール 3 2 を記憶装置 3 0 に書き込み (S 1 0 1 4)、処理を終了する。一方、処理済みでない場合 (S 1 0 1 2 において N O の場合)、加工手段 2 2 は、次に処理する二次的文 3 2 7 (後続する行 3 2 0 の二次的文 3 2 7) を取得し (S 1 0 1 6)、取得した二次的文 3 2 7 について階層識別 3 2 3 の付加等を行う (S 1 0 0 6 ~ S 1 0 1 0)。

【 0 0 8 3 】

以上に説明したように、加工手段 2 2 によって、階層識別 3 2 3、命令識別 3 2 4 及びオペランド情報 3 2 5 が付加された二次的文 3 2 7 からなる二次的モジュール 3 2 が記憶装置 3 0 に記憶される。換言すれば、加工手段 2 2 を備えた装置 1 0 は、階層識別 3 2 3、命令識別 3 2 4 及びオペランド情報 3 2 5 が付加された二次的文 3 2 7 からなる二次的モジュール 3 2 を記憶装置 3 0 に出力することができる。

10

【 0 0 8 4 】

なお、階層識別 3 2 3 の付加 (S 1 0 0 6)、命令識別 3 2 4 の付加 (S 1 0 0 8) 及びオペランド情報 3 2 5 の付加 (S 1 0 1 0) のうちの一部を行うようにしてもよい。例えば、加工手段 2 2 を起動する際に、入力装置 4 0 から階層識別 3 2 3 の付加を行うか否かの指示を入力するようにしてもよい。

【 0 0 8 5 】

また、階層識別 3 2 3 の付加、命令識別 3 2 4 の付加及びオペランド情報 3 2 5 の付加のうちの一部を変換手段 2 1 が行うようにしてもよい。例えば、変換手段 2 1 が、二次的モジュール 3 2 に出力する二次的文 3 2 7 に階層識別 3 2 3 を付加する階層識別付加機能と、二次的文 3 2 7 にオペランド情報 3 2 5 を付加するオペランド付加機能とのうちの少なくとも一方を有するように構成してもよい。

20

【 0 0 8 6 】

図 1 1 から理解されるように、本実施の形態による選択手段 2 3 は、読込機能 (S 1 1 0 0、S 1 1 0 2) と、特定情報取得機能 (S 1 1 0 4) と、選択機能 (S 1 1 0 6) と、書込機能 (S 1 1 0 8) とを有している。読込機能は、処理対象の二次的モジュール 3 2 を記憶装置 3 0 から読み込む機能である。特定情報取得機能は、階層識別特定情報を取得する機能である。ここで、階層識別特定情報は、処理対象の二次的モジュール 3 2 の二次的文 3 2 7 のうち選択対象とすべきものの階層識別 3 2 3 を特定するものである。選択機能は、処理対象の二次的モジュール 3 2 の二次的文 3 2 7 のうち、階層識別 3 2 3 が階層識別特定情報によって特定される階層識別と同一であるものを選択して選択後の二次的モジュール 3 2 に出力する機能である。書込機能は、選択された二次的文 3 2 7 からなる選択後の二次的モジュール 3 2 を記憶装置 3 0 に記憶する機能である。

30

【 0 0 8 7 】

より具体的には、図 1 1 に示されるように、選択手段 2 3 は、例えば変換手段 2 1 と同様に起動され、処理対象の二次的モジュール 3 2 を特定するための二次的モジュール名を取得する (S 1 1 0 0)。選択手段 2 3 は、取得した二次的モジュール名によって特定される二次的モジュール 3 2 を記憶装置 3 0 から読み込む (S 1 1 0 2)。なお処理対象の二次的モジュール 3 2 が記憶装置 3 0 に記憶されていない場合、選択手段 2 3 は、処理を終了する。

40

【 0 0 8 8 】

次に、選択手段 2 3 は、階層識別特定情報を取得する (S 1 1 0 4)。本実施の形態においては、例えば処理対象の二次的モジュール 3 2 を表示装置 5 0 に表示させ、ワークシートの行 3 2 0 を表示した行がダブルクリックされた場合に、ダブルクリックされた行に表示されている階層識別 3 2 3 を階層識別特定情報として取得するようにできる。

【 0 0 8 9 】

次に、選択手段 2 3 は、処理対象の二次的モジュール 3 2 の先頭の行 3 2 0 から末尾の行 3 2 0 まで、取得した階層識別特定情報と階層識別 3 2 3 とを順次比較し、一致する行

50

3 2 0のうち先頭の行3 2 0から末尾の行3 2 0までの二次的文3 2 7（選択対象となる二次的文3 2 7）を選択する（S 1 1 0 6）。選択手段2 3は、選択対象となる二次的文3 2 7の行3 2 0からなる選択後の二次的モジュール3 2（選択後のワークシート）を作成する。

【0 0 9 0】

次に、選択手段2 3は、選択後の二次的モジュール3 2を記憶装置3 0に書き込み（S 1 1 0 8）、処理を終了する。

【0 0 9 1】

以上の説明から理解されるように、選択後の二次的モジュール3 2は、処理対象の二次的モジュール3 2のうち、指定された階層識別3 2 3を有する部分である。このように二次的モジュール3 2の一部を選択可能とすることにより、例えばメンテナンスを行う際、処理について確認したい箇所をコンパクトに表示装置5 0に表示することが可能になる。なお、階層識別3 2 3以外の項目によって二次的文3 2 7を選択するようにしてもよい。例えば、入力装置4 0によって行番号が入力されるようにして、入力された行番号と一致する行番号3 2 2を検索し、検索された行番号3 2 2の階層識別3 2 3によって二次的文3 2 7を選択するようにしてもよい。

10

【0 0 9 2】

図1 2から理解されるように、本実施の形態による表示属性設定手段2 4は、読込機能（S 1 2 0 0、S 1 2 0 2）と、初期検索文字取得機能（S 1 2 0 4）と、表示属性設定機能（S 1 2 0 6～S 1 2 1 0）と、書込機能（S 1 2 1 2）とを有している。

20

【0 0 9 3】

より具体的には、図1 2に示されるように、表示属性設定手段2 4は、例えば変換手段2 1と同様に起動され、処理対象の二次的モジュール3 2を特定するための二次的モジュール名を取得する（S 1 2 0 0）。表示属性設定手段2 4は、取得した二次的モジュール名によって特定される処理対象の二次的モジュール3 2を記憶装置3 0から読み込む（S 1 2 0 2）。なお処理対象の二次的モジュール3 2が記憶装置3 0に記憶されていない場合、表示属性設定手段2 4は、処理を終了する。

【0 0 9 4】

次に、表示属性設定手段2 4は、初期検索文字を取得する（S 1 2 0 4）。本実施の形態による初期検索文字は二次的文3 2 7において参照又は更新される変数名である。初期検索文字は、変数名の一部分であってもよいし、二次的文3 2 7において参照される定数であってもよい。

30

【0 0 9 5】

次に、表示属性設定手段2 4は、二次的文3 2 7の第1オペランド3 2 5 f又は第2オペランド3 2 5 sのうち初期検索文字を含むものを追加検索文字とする（S 1 2 0 6）。

【0 0 9 6】

より具体的には、表示属性設定手段2 4は、まず処理対象の二次的モジュール3 2の二次的文3 2 7のうち、オペランド情報3 2 5の第1オペランド3 2 5 f又は第2オペランド3 2 5 sに初期検索文字を含んでいるものを検索する。このとき、単純な文字検索を行うだけでなく、例えばREDEFINES句によって再定義された変数について、再定義される変数名によって文字検索してもよい。具体的には、例えば変数名V R B Lが変数名V V Vに再定義され、第1オペランド3 2 5 fに変数名V V Vが設定されており、初期検索文字がV R B Lである場合、第1オペランド3 2 5 fは初期検索文字を含んでいると判定してもよい。上述のREDEFINES句による例に限らず、実質的に初期検索文字を含んでいるか否かを判定してもよい。なお、このような判定を行うためには、変換手段2 1がモジュール3 1から二次的モジュール3 2を生成する際に、生成した二次的モジュール3 2と対応付けてモジュール3 1のDATA DIVISIONを記憶装置3 0に記憶すればよい。

40

【0 0 9 7】

なお、表示属性設定手段2 4は、第1オペランド3 2 5 f又は第2オペランド3 2 5 sが初期検索文字を含んでいる場合、第1オペランド3 2 5 fと第2オペランド3 2 5 sの

50

2つを追加検索文字とする。

【0098】

次に、表示属性設定手段24は、新たな追加検索文字が見つからなくなるまで、二次的文327の第1オペランド325f又は第2オペランド325sのうち追加検索文字を含むものを追加検索文字とする(S1208)。より具体的には、表示属性設定手段24は、二次的文327のうちオペランド情報325の第1オペランド325f又は第2オペランド325sに追加検索文字を含んでいるものを更に検索する。このときの検索方法は初期検索文字の検索方法と同じである。表示属性設定手段24は、第1オペランド325f又は第2オペランド325sが追加検索文字を含んでいる場合、第1オペランド325fと第2オペランド325sの2つを追加検索文字とする。

10

【0099】

次に、表示属性設定手段24は、二次的文327のうちオペランド情報325の第1オペランド325f又は第2オペランド325sに初期検索文字又は追加検索文字を含んでいる行320の表示属性328に所定の表示属性(例えば強調表示可能な背景色)を設定する。(S1210)。

【0100】

次に、表示属性設定手段24は、二次的文327等に所定の表示属性が設定された二次的モジュール32を記憶装置30に書き込む。即ち、処理対象の二次的モジュール32を更新して(S1212)、処理を終了する。

【0101】

以上の説明から理解されるように、表示属性設定後の二次的モジュール32の二次的文327のうち初期検索文字と関連する変数又は定数を参照・更新しているものには、所定の背景色等が設定される。従って、例えばメンテナンスを行う際、所定の変数に関連する処理が目立つようにして表示装置50に表示することが可能になる。また、例えば、初期検索文字を含む二次的文327の背景色と、追加検索文字を含む二次的文327の背景色を別の色とすることにより、関連する処理を、より視覚的に把握することができる。

20

【0102】

図13から理解されるように、本実施の形態によるフローチャート作成手段25は、読込機能(S1300、S1302)と、フローチャート作成機能(S1304)と、書込機能(S1306)とを有している。

30

【0103】

より具体的には、図13に示されるように、フローチャート作成手段25は、例えば変換手段21と同様に起動され、処理対象の二次的モジュール32を特定するための二次的モジュール名を取得する(S1300)。フローチャート作成手段25は、取得した二次的モジュール名によって特定される処理対象の二次的モジュール32を記憶装置30から読み込む(S1302)。なお処理対象の二次的モジュール32が記憶装置30に記憶されていない場合、フローチャート作成手段25は、処理を終了する。

【0104】

次に、フローチャート作成手段25は、二次的条件式の成否に対応した二次的文327を表の左右に並べた表形式のフローチャート33を作成する(S1304)。より具体的には、二次的モジュール32の二次的文327のうち階層識別323によって示される階層が同一である二次的文327について、二次的条件式が成立した場合に実行される二次的文327と二次的条件式が成立しない場合に実行される二次的文327とを表の左右に並べたフローチャート33を作成する。このとき、フローチャート33に、二次的文327と対応するように行番号322を設定する。なお、階層識別323の分岐の深さが同一であり、且つ分岐階層のうち末尾の1桁(1又は2)を除いた部分が同一である場合、階層識別323によって示される階層が同一であると判定する。

40

【0105】

次に、フローチャート作成手段25は、フローチャート33を記憶手段30に書き込み(S1306)、処理を終了する。

50

【 0 1 0 6 】

図 1 4 から理解されるように、本実施の形態によるマトリックス作成手段 2 6 は、読込機能 (S 1 4 0 0 、 S 1 4 0 2) と、二次的文出力機能 (S 1 4 0 4) と、ケース生成機能 (S 1 4 0 6) と、マトリックス作成機能 (S 1 4 0 8 、 S 1 4 1 0) と、書込機能 (S 1 4 1 2) とを有している。

【 0 1 0 7 】

より具体的には、図 1 4 に示されるように、マトリックス作成手段 2 6 は、例えば変換手段 2 1 と同様に起動され、処理対象の二次的モジュール 3 2 を特定するための二次的モジュール名を取得する (S 1 4 0 0)。マトリックス作成手段 2 6 は、取得した二次的モジュール名によって特定される処理対象の二次的モジュール 3 2 を記憶装置 3 0 から読み込む (S 1 4 0 2)。なお処理対象の二次的モジュール 3 2 が記憶装置 3 0 に記憶されていない場合、マトリックス作成手段 2 6 は、処理を終了する。

10

【 0 1 0 8 】

次に、マトリックス作成手段 2 6 は、二次的文 3 2 7 を表形式のマトリックス 3 4 の所定の列に出力する (S 1 4 0 4)。詳しくは、マトリックス作成手段 2 6 は、処理対象の二次的モジュール 3 2 の行番号 3 2 2、一連番号 3 2 6、及び二次的文 3 2 7 をマトリックス 3 4 の所定の列 (例えば、第 1 列目乃至第 3 列目) の各行に出力する。

【 0 1 0 9 】

次に、マトリックス作成手段 2 6 は、二次的条件式の成否についてのケースを生成する (S 1 4 0 6)。より具体的には、マトリックス作成手段 2 6 は、二次的モジュール 3 2 の二次的文 3 2 7 に記述されている 1 以上の二次的条件式について、その全てが成立する場合と、その一部が成立する場合と、その全てが成立しない場合とを含む全てのケースを生成する。このとき、階層識別 3 2 3 の分岐識別が 1 である二次的文 3 2 7 に記述された IF 文 (階層の最上位の IF 文) 毎にケースを作成してもよいし、階層の最上位の IF 文の夫々についてのケースを組み合わせることでケースを作成してもよい。更に、階層の最上位の IF 文のうちの一部についてケースを組み合わせることでケースを作成してもよい。

20

【 0 1 1 0 】

次に、マトリックス作成手段 2 6 は、生成したケースの夫々に、マトリックス 3 4 の列のいずれかを関連表示列として対応させる (S 1 4 0 8)。例えば、生成したケースの数が 5 であれば、第 1 のケース乃至第 5 のケースに、第 4 列目乃至第 8 列目を夫々対応させる。この場合、例えば第 3 のケースに対応する関連表示列は第 6 列目である。

30

【 0 1 1 1 】

次に、マトリックス作成手段 2 6 は、関連表示列の夫々につき、対応するケースにおいて実行対象となる二次的文 3 2 7 が出力されている行に、実行対象となることを示す表示情報を出力する (S 1 4 1 0)。具体的には、前述したように、、I 等の表示情報が出力される。

【 0 1 1 2 】

次に、マトリックス作成手段 2 6 は、マトリックス 3 4 を記憶手段 3 0 に書き込み (S 1 4 1 2)、処理を終了する。

【 0 1 1 3 】

マトリックス作成手段 2 6 によって、図 7 及び図 1 5 に示されるようなマトリックス 3 4 を作成することができる。図 7 に部分的に示されているマトリックス 3 4 は、階層の最上位の IF 文毎にケースを作成したものであり、ケースの数は 5 である。図 1 5 に部分的に示されているマトリックス 3 4 は、階層の最上位の IF 文の夫々についてのケースを組み合わせることでケースを作成したものであり、ケースの数は 1 8 0 である。

40

【 0 1 1 4 】

本実施の形態による装置 1 0 は、メンテナンスによってモジュール 3 1 が変更された場合、モジュール 3 1 の変更箇所を表示装置 5 0 に視覚的に表示することもできる。

【 0 1 1 5 】

例えば、図 1 6 に示されるように、CCC モジュールを変更した場合、変更前後の CC

50

Cモジュールを照合することで、CCCモジュールにおける変更前後の差分を得ることができる。変更前後の差分は、例えば、行毎にモジュール名(CCC)を付加して、テキストファイル(差分ファイル)として記憶装置30に記憶することができる。このとき、マトリックス作成手段26を、差分ファイルを参照してマトリックス34を作成するにすればよい。

【0116】

図17に示されるように、第1の変形例によるマトリックス作成手段26は、マトリックス作成機能において、記憶装置30に記憶された二次的モジュール32の二次的文327の一部を選択して、所定の表示属性を表示属性328に設定する(S1405)。より具体的には、第1の変形例によるマトリックス作成手段26は、記憶装置30から差分ファイルを読み込み、差分ファイルの各行について、モジュール識別321及び一連番号326が一致する行320を検索し、検索された行320の表示属性328に強調表示のための背景色等を設定する。

10

【0117】

また、第1の変形例によるマトリックス作成手段26は、関連表示列のうち選択された二次的文327が実行対象となるケースに対応するもののみを記憶対象として抽出する(S1411)。

【0118】

以上に説明した他は、第1の変形例によるマトリックス作成手段26は、前述した本実施の形態によるマトリックス作成手段26と同じ機能を備えている。第1の変形例によるマトリックス作成手段26により、モジュール31の変更前後の差分と関連するケースのみをマトリックス34に作成することができる。従って、メンテナンスによって変更した機能に関連するテストケースを、より容易に作成することができる。なお、マトリックス作成手段26は、強調表示のための背景色等の設定を行う一方、実行対象となるケースに対応する関連表示列の抽出を行わないようにしてもよい(第2の変形例)。

20

【0119】

図18から理解されるように、マトリックス作成手段26の第2の変形例によって作成されたマトリックス34を表示装置50に表示すると、変更された文312に対応する行320(変更された箇所)と、変更された文312から生成された二次的文327の行320(二次的に変更された箇所)とが、他の行320と異なる色で夫々表示される。従って、変更された箇所及び二次的に変更された箇所における関連表示列の表示内容を見ることで、関連するケースを選択することができる。より具体的には、変更された箇所及び二次的に変更された行の関連表示列に、I、T、E、EIのいずれかが表示されている場合、変更した機能に関連するケースであると判断することができる。

30

【0120】

図18に示されるように、CCCモジュールを前述のように変更した例においては、変更した機能に関連するケースは、C2乃至C4である。従って、第1の変形例によるマトリックス作成手段26によってマトリックス34を作成した場合、作成されたマトリックス34は、図18の第1列目乃至第3列目および第5列目乃至第7列目を有する一方、第4列目と第8列目とを有していない。

40

【0121】

図19から理解されるように、フローチャート作成手段25についても、マトリックス作成手段26の第2の変形例と同様に变形することで、変更された箇所と二次的に変更された箇所とを強調表示することができる。

【0122】

また、本実施の形態による装置10は、様々に变形することが可能である。例えば、図20に示されるように、装置10を、変換手段21を備える装置10と加工手段22を備える装置10と、選択手段23等を備える装置(図示せず)に分けてもよい。装置10と装置10との夫々は、通信回線80を経由して、ファイルサーバ60と通信可能に接続されている。通信回線80は、例えばLAN(Local Area Network)であってもよ

50

いし、インターネットであってもよい。ファイルサーバ60は、記憶装置（記憶手段）70を備えている。記憶装置70は、モジュール31と、二次的モジュール32と、フローチャート33と、マトリックス34とを記憶することができる。従って、装置10と装置10とは、記憶装置70からモジュール31等を読み込み、記憶装置70にモジュール31等を書き込むことができる。

【0123】

装置10は、装置本体20と、入力装置40と、表示装置50とを備えている。装置10は、装置本体20と、入力装置40と、表示装置50とを備えている。装置本体20は、変換手段21と記憶装置（記憶手段）30とを備えている。装置本体20は、加工手段22と記憶装置（記憶手段）30とを備えている。

10

【0124】

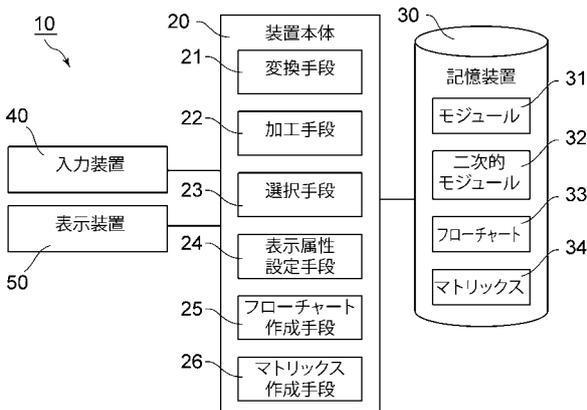
本変形例における記憶装置30と記憶装置30とは、主記憶装置によって構成することができる。例えば、変換手段21は、モジュール31を記憶装置30から読み込むことができる。

【符号の説明】

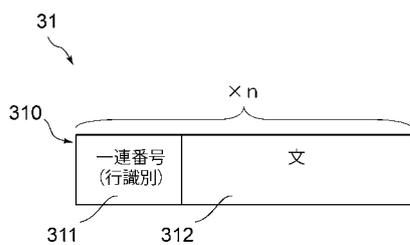
【0125】

10, 10, 10	装置（モジュールの構造解析を支援する装置）	
20, 20, 20	装置本体	
21	変換手段	
22	加工手段	20
23	選択手段	
24	表示属性設定手段	
25	フローチャート作成手段	
26	マトリックス作成手段	
30, 30, 30	記憶装置（記憶手段）	
31	モジュール（ソースプログラム）	
310	行	
311	一連番号（行識別）	
312	文	
32	二次的モジュール（二次的ソースプログラム）	30
320	二次的行	
321	モジュール識別	
322	行番号	
323	階層識別	
324	命令識別	
325	オペランド情報	
325f	第1オペランド（オペランド）	
325o	演算子	
325s	第2オペランド（オペランド）	
326	一連番号（行識別）	40
327	二次的文	
328	表示属性	
33	フローチャート	
34	マトリックス	
40	入力装置	
50	表示装置	
60	ファイルサーバ	
70	記憶装置（記憶手段）	
80	通信回線	

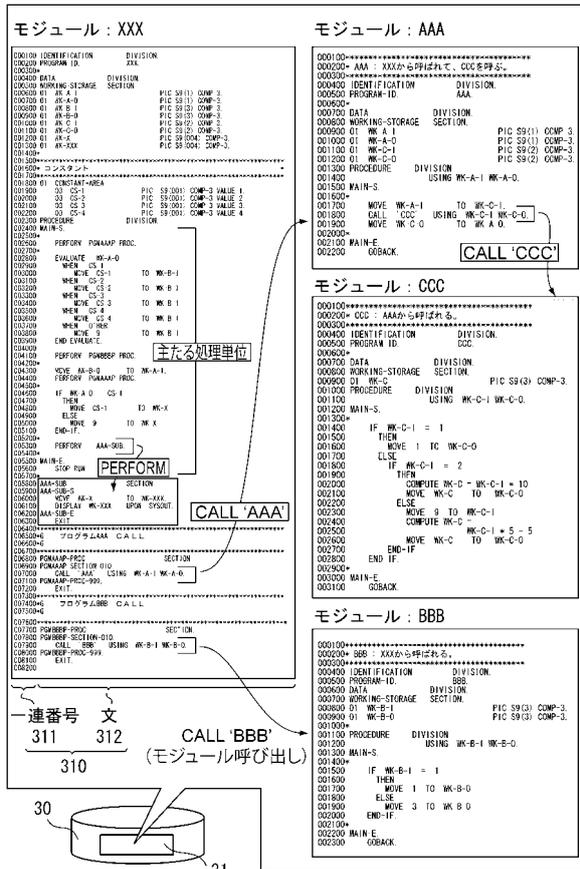
【図1】



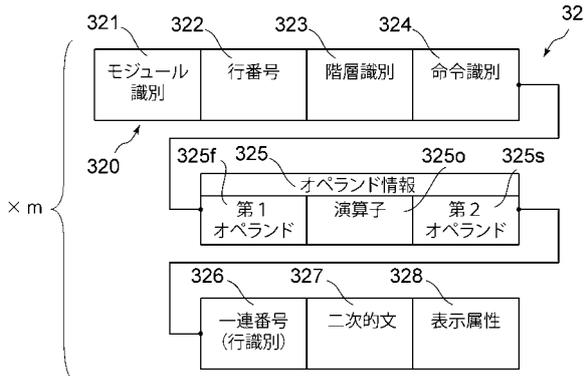
【図2】



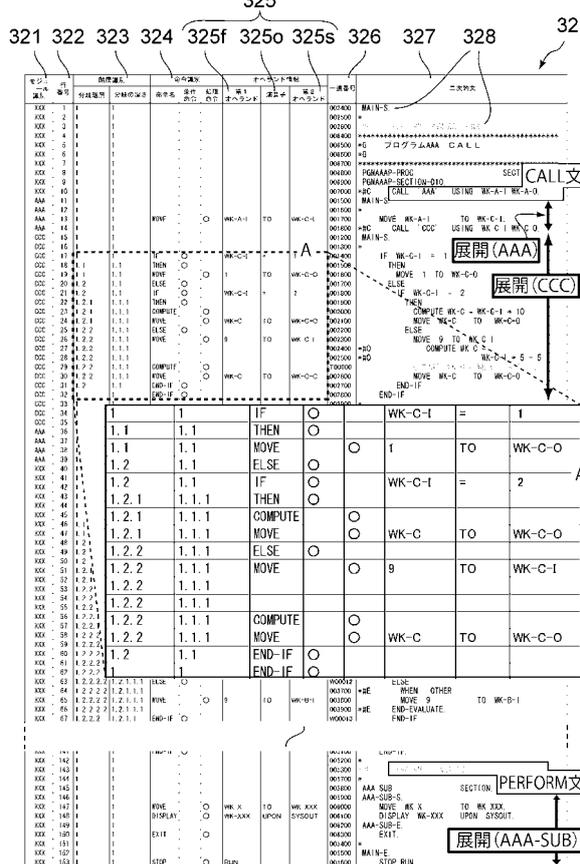
【図3】



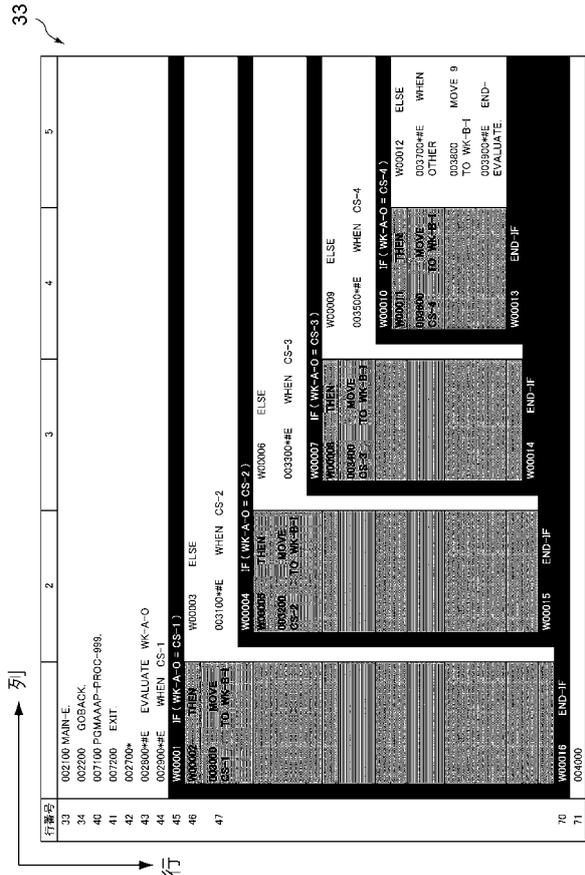
【図4】



【図5】



【図6】

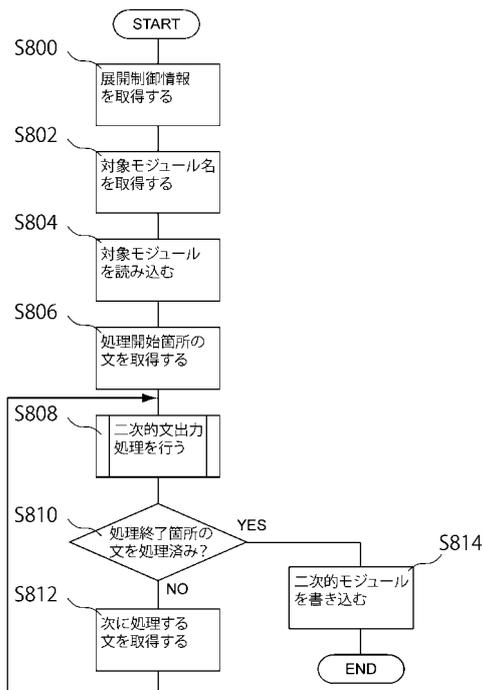


【図7】

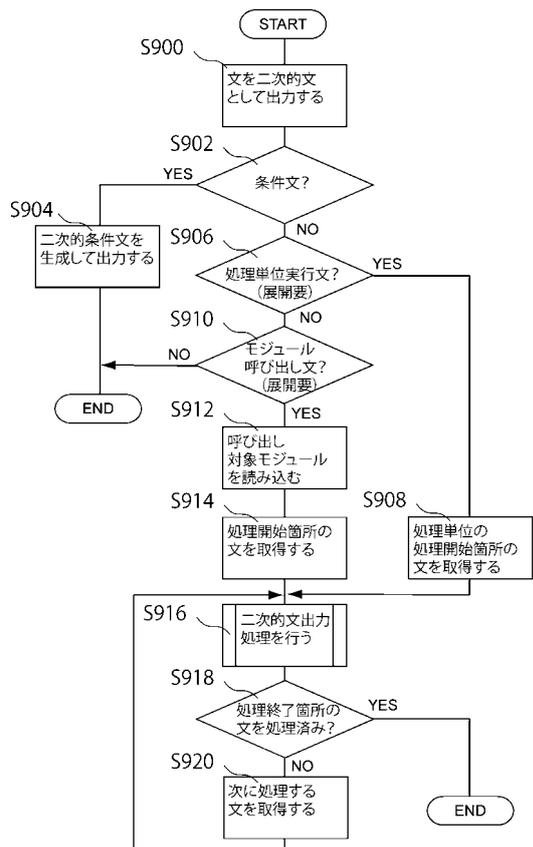
第1列～第8列

行番号	一次番号	二次的文	処理と実行ケースのマトリクス				
行			C1	C2	C3	C4	C5
1	002100	MAIN-S	*	*	*	*	*
2	002500	*	*	*	*	*	*
3	002600	*:E H1=999 PGMMAAP-PROC	*	*	*	*	*
4	002640	*****	*	*	*	*	*
5	002650	HG プログラムAAA CALL	*	*	*	*	*
6	002690	HG	*	*	*	*	*
7	002700	*****	*	*	*	*	*
8	002800	PGMMAAP-PROC SECTION	*	*	*	*	*
9	002890	PGMMAAP-SECTION-010	*	*	*	*	*
10	007000	HDC CALL AAA USING WK-A-1 WK-A-0	*	*	*	*	*
11	001500	MAIN-S	*	*	*	*	*
12	001500	MAIN-S	*	*	*	*	*
13	001700	MOVE WK-A-1 TO WK-C-1	*	*	*	*	*
14	001800	HDC CALL COCC USING WK-G-1 WK-C-0	*	*	*	*	*
15	001200	MAIN-S	*	*	*	*	*
16	001300	*	*	*	*	*	*
17	001400	IF WK-C-1 = 1	*	*	*	*	*
18	001500	THEN	*	*	*	*	*
19	001600	MOVE 1 TO WK-C-0	*	*	*	*	*
20	001700	ELSE	*	*	*	*	*
21	001800	IF WK-C-1 = 2	*	*	*	*	*
22	001900	THEN	*	*	*	*	*
23	002000	COMPUTE WK-C = WK-C-1 * 10	*	*	*	*	*
24	002100	MOVE WK-C TO WK-C-0	*	*	*	*	*
25	002200	ELSE	*	*	*	*	*
26	002300	MOVE 9 TO WK-C-1	*	*	*	*	*
27	002400	HDC COMPUTE WK-C =	*	*	*	*	*
28	002500	HDC WK-C-1 * 5 - 5	*	*	*	*	*
29	002600	END-IF	*	*	*	*	*
30	002600	END-IF	*	*	*	*	*
31	002700	END-IF	*	*	*	*	*
32	002800	GOBACK	*	*	*	*	*
33	002900	MAIN-E	*	*	*	*	*
34	003000	GOBACK	*	*	*	*	*
35	003100	MOVE WK-C-0 TO WK-A-0	*	*	*	*	*
36	003200	MAIN-E	*	*	*	*	*
37	002000	*	*	*	*	*	*
38	002100	MAIN-E	*	*	*	*	*
39	002200	GOBACK	*	*	*	*	*
40	007100	PGMMAAP-PROC-999	*	*	*	*	*
41	007200	EXIT	*	*	*	*	*
42	002700	*E EVALUATE WK-A-0	*	*	*	*	*
43	002800	*E WHEN CS-1	*	*	*	*	*
44	002900	*E IF (WK-A-0 = CS-1)	*	*	*	*	*
45	W00001	THEN	*	*	*	*	*
46	W00002	MOVE CS-1 TO WK-B-1	*	*	*	*	*
47	W00003	ELSE	*	*	*	*	*
48	W00004	IF (WK-A-0 = CS-2)	*	*	*	*	*
49	W00005	THEN	*	*	*	*	*
50	W00006	MOVE CS-2 TO WK-B-1	*	*	*	*	*
51	W00007	ELSE	*	*	*	*	*
52	W00008	IF (WK-A-0 = CS-3)	*	*	*	*	*
53	W00009	THEN	*	*	*	*	*
54	W00010	MOVE CS-3 TO WK-B-1	*	*	*	*	*
55	W00011	ELSE	*	*	*	*	*
56	W00012	IF (WK-A-0 = CS-4)	*	*	*	*	*
57	W00013	THEN	*	*	*	*	*
58	W00014	MOVE CS-4 TO WK-B-1	*	*	*	*	*
59	W00015	ELSE	*	*	*	*	*
60	W00016	IF (WK-A-0 = CS-5)	*	*	*	*	*
61	W00017	THEN	*	*	*	*	*
62	W00018	MOVE CS-5 TO WK-B-1	*	*	*	*	*
63	W00019	ELSE	*	*	*	*	*
64	W00020	IF (WK-A-0 = CS-6)	*	*	*	*	*
65	W00021	THEN	*	*	*	*	*
66	W00022	MOVE CS-6 TO WK-B-1	*	*	*	*	*
67	W00023	ELSE	*	*	*	*	*
68	W00024	IF (WK-A-0 = CS-7)	*	*	*	*	*
69	W00025	THEN	*	*	*	*	*
70	W00026	MOVE CS-7 TO WK-B-1	*	*	*	*	*
71	W00027	ELSE	*	*	*	*	*

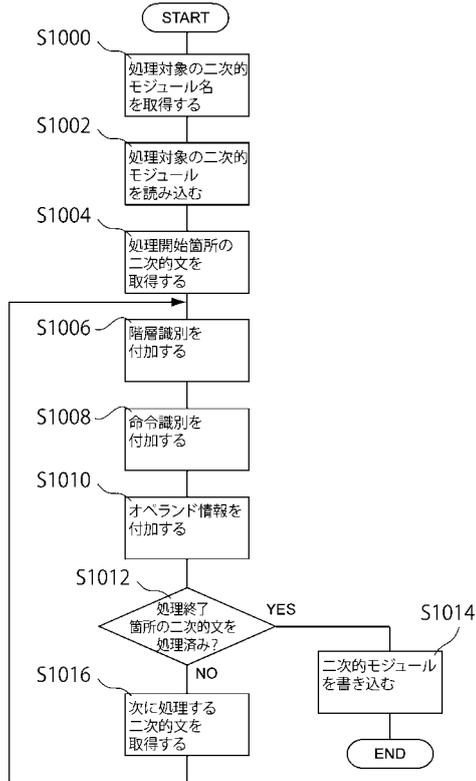
【図8】



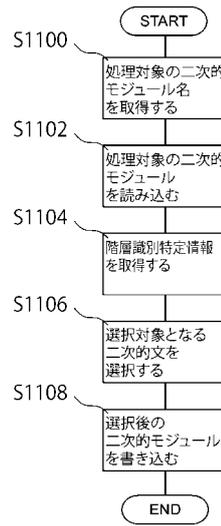
【図9】



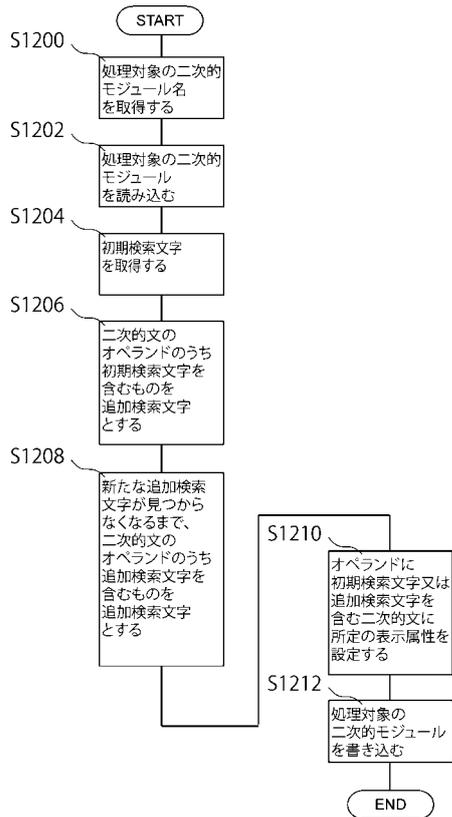
【図10】



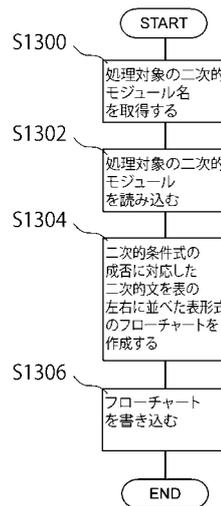
【図11】



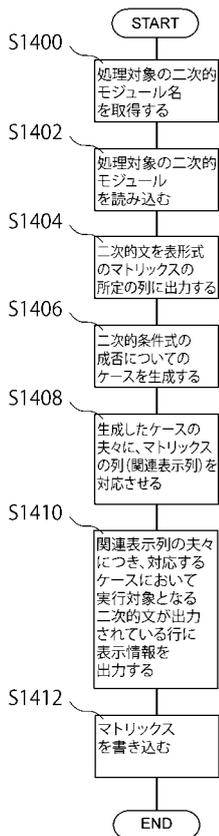
【図12】



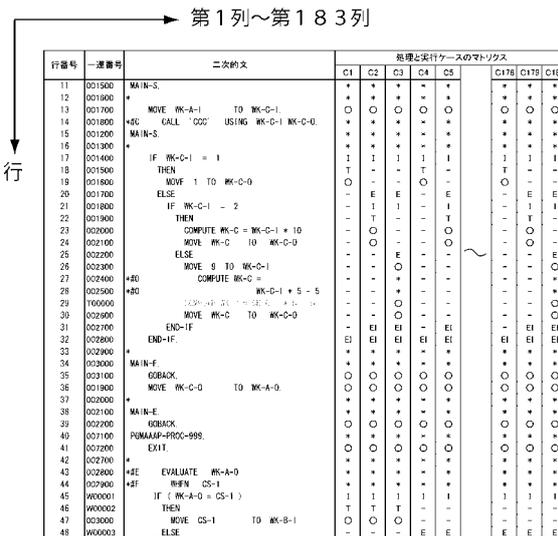
【図13】



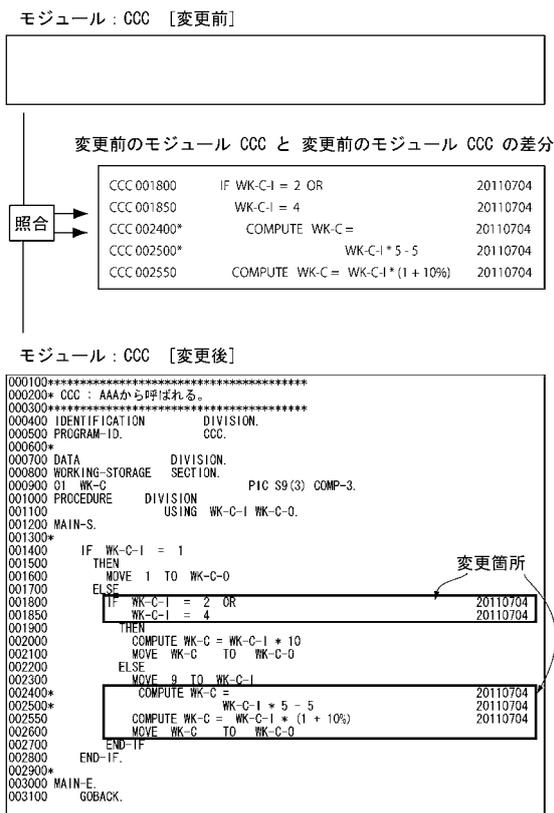
【図14】



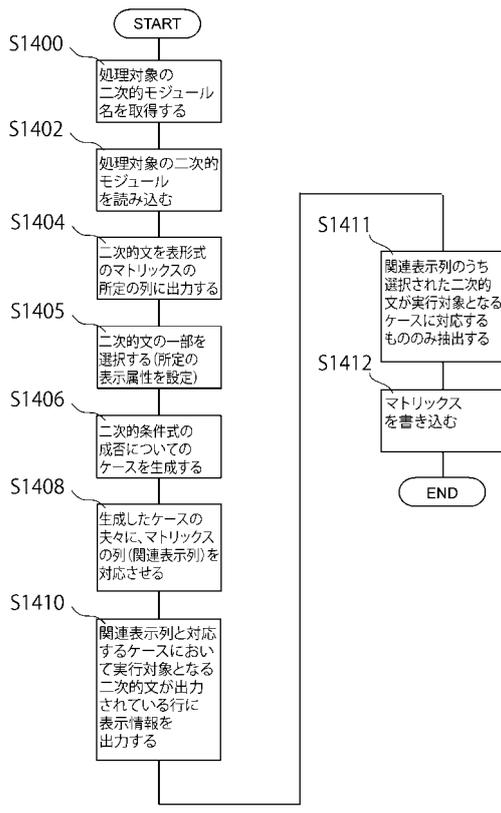
【図15】



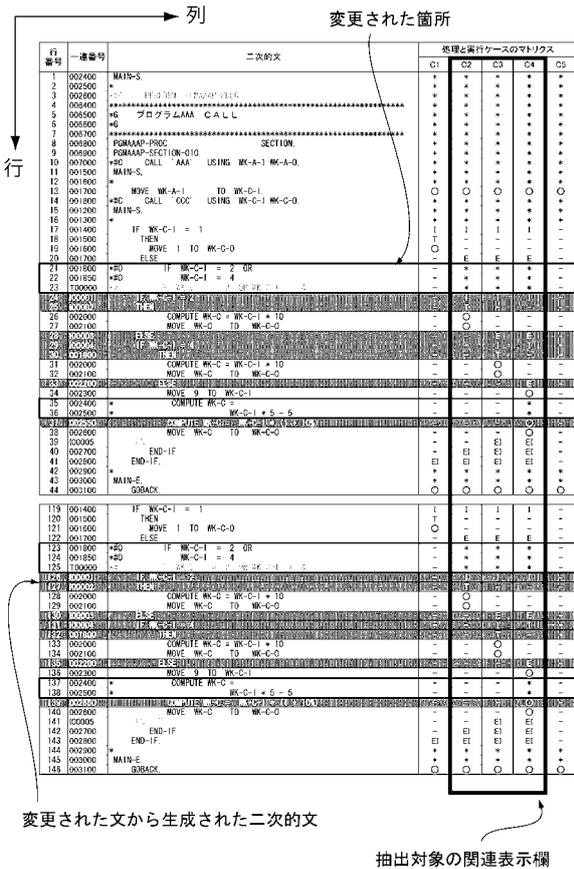
【図16】



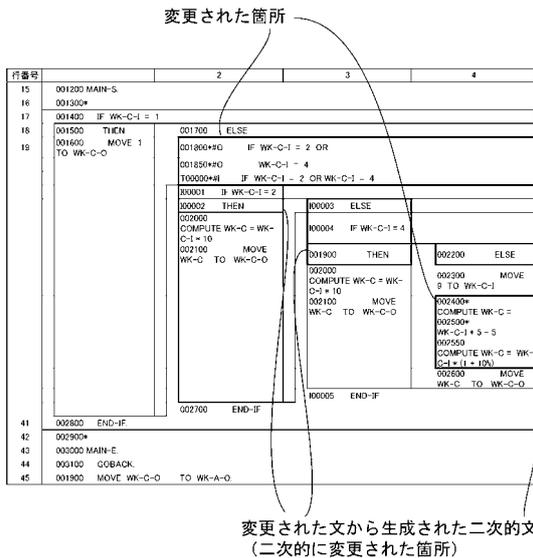
【図17】



【図18】



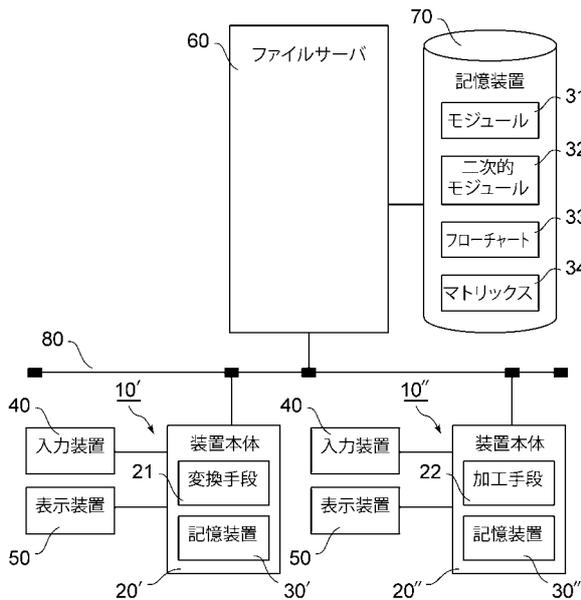
【図19】



変更された文から生成された二次的文

抽出対象の関連表示欄

【図20】



フロントページの続き

(72)発明者 石田 伸一
神奈川県小田原市曾我岸102番地

審査官 坂庭 剛史

(56)参考文献 特開2008-181311(JP,A)
特開平08-095763(JP,A)
特開平11-272503(JP,A)
特開平01-286026(JP,A)
特開2001-154835(JP,A)
特開平04-127235(JP,A)
特開平07-281883(JP,A)
特開2009-086922(JP,A)
特開平10-320190(JP,A)
特開平09-282150(JP,A)
特開平07-261990(JP,A)
特開平04-165425(JP,A)
特開平03-292532(JP,A)
特開平02-140828(JP,A)
臥待天祐,最新開発環境レポート:組み込みソフト向けコードレビュー支援ツール Development Assistant for C4.0, C MAGAZINE, 日本, ソフトバンククリエイティブ株式会社, 2005年12月 1日, 第17巻, 第12号(通巻195号), pp. 86~91

(58)調査した分野(Int.Cl., DB名)
G06F 9/44