

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4769946号
(P4769946)

(45) 発行日 平成23年9月7日(2011.9.7)

(24) 登録日 平成23年7月1日(2011.7.1)

(51) Int. Cl. F I
G06F 12/02 (2006.01) G O 6 F 12/02 5 3 0 C
G06F 12/00 (2006.01) G O 6 F 12/00 5 9 1

請求項の数 11 (全 31 頁)

<p>(21) 出願番号 特願2007-26111 (P2007-26111) (22) 出願日 平成19年2月5日 (2007.2.5) (65) 公開番号 特開2008-191952 (P2008-191952A) (43) 公開日 平成20年8月21日 (2008.8.21) 審査請求日 平成22年1月14日 (2010.1.14)</p> <p>早期審査対象出願</p>	<p>(73) 特許権者 504132272 国立大学法人京都大学 京都府京都市左京区吉田本町36番地1</p> <p>(74) 代理人 100084375 弁理士 板谷 康夫</p> <p>(74) 代理人 100121692 弁理士 田口 勝美</p> <p>(74) 代理人 100125221 弁理士 水田 慎一</p> <p>(72) 発明者 鶴川 始陽 京都府京都市左京区吉田本町 国立大学法人京都大学 大学院情報学研究科内</p> <p>(72) 発明者 湯淺 太一 京都府京都市左京区吉田本町 国立大学法人京都大学 大学院情報学研究科内</p> <p style="text-align: right;">最終頁に続く</p>
--	--

(54) 【発明の名称】 メモリ管理方法、メモリ管理装置、及びメモリ管理プログラムが記録されている記録媒体

(57) 【特許請求の範囲】

【請求項1】

メモリ中に、動的にデータを生成するプログラム（以下、主プログラム）の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ（以下、オブジェクト）を記録するためのヒープ領域とが設けられ、

主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域（以下、利用可能領域）を生成すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理方法において、

前記オブジェクトは、オブジェクトの複製先の位置又は複製元の位置を記録するための複製位置ポインタをさらに含み、

前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、

前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定ステップと、

前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、該複製元オブジェクト及び該複製先オブジェクトの複製位置ポインタの参照先を互いのオブジェクトの位置に設定するオブジェクト複製ステップと、

前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブ

10

20

ジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第1ポインタ更新ステップと、

前記限定範囲の領域を利用可能領域にする限定範囲利用可能化ステップと、

前記オブジェクト複製ステップを実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断ステップと、

前記複製判断ステップにより書込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書込みを行う複製書込みステップと、を含み、

前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、

前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第3ポインタ更新ステップと、

前記第3ポインタ更新ステップによるオブジェクト位置ポインタの更新処理を中断するときに、該第3ポインタ更新ステップによる処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定する障壁設定ステップと、をさらに備え、

前記第1ポインタ更新ステップは、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記障壁が、関数の実行完了後、前記第3ポインタ更新ステップによる処理を再開させることを特徴とするメモリ管理方法。

【請求項2】

メモリ中に、動的にデータを生成するプログラム(以下、主プログラム)の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ(以下、オブジェクト)を記録するためのヒープ領域とが設けられ、

主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域(以下、利用可能領域)を生成すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理方法において、

前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、

前記メモリ中に、オブジェクトの位置を記録するオブジェクト位置記録領域がさらに設けられ、

前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定ステップと、

前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、前記オブジェクト位置記録領域に、該複製元オブジェクトの位置及び該複製先オブジェクトの位置を記録するオブジェクト複製ステップと、

前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第1ポインタ更新ステップと、

前記限定範囲の領域を利用可能領域にする限定範囲利用可能化ステップと、

前記オブジェクト複製ステップを実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断ステップと、

10

20

30

40

50

前記複製判断ステップにより書込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書込みを行う複製書込みステップと、を含み、

前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、

前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第3ポインタ更新ステップと、

前記第3ポインタ更新ステップによるオブジェクト位置ポインタの更新処理を中断するときに、該第3ポインタ更新ステップによる処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定する障壁設定ステップと、をさらに備え、

前記第1ポインタ更新ステップは、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記障壁が、関数の実行完了後、前記第3ポインタ更新ステップによる処理を再開させることを特徴とするメモリ管理方法。

【請求項3】

前記オブジェクトは、前記データ要素として前記オブジェクト位置ポインタをさらに含み、

前記複製元オブジェクトの位置を参照する前記ヒープ領域内のオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第2ポインタ更新ステップと、

前記第2ポインタ更新ステップを実行中に、前記主プログラムが前記ヒープ領域に前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの書込み処理を行うとき、又は書込み処理を行った後、該オブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に変更する参照先変更ステップと、

前記第2ポインタ更新ステップの実行中に、前記主プログラムが2つのオブジェクト位置ポインタの参照先が同じオブジェクトの位置であるかどうかの検査処理を行うとき、該2つのオブジェクト位置ポインタのうち的一方が前記複製元オブジェクトの位置を、他方が前記複製先オブジェクトの位置を参照している場合、同一のオブジェクトを参照していると判断する同一判断ステップと、をさらに備え、

前記複製判断ステップは、前記オブジェクト複製ステップ、又は前記第2ポインタ更新ステップを実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断することを特徴とする請求項1又は請求項2に記載のメモリ管理方法。

【請求項4】

前記ヒープ領域内に、前記スタック領域を除く前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項1乃至請求項3のいずれかに記載のヒープ領域とし、

前記範囲設定ステップは、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、

前記第1ポインタ更新ステップは、

前記限定範囲内の前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の前記データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、

10

20

30

40

50

前記限定範囲及び前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新することを特徴とする請求項1乃至請求項3のいずれかに記載のメモリ管理方法。

【請求項5】

前記ヒープ領域内に、前記スタック領域を含む前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項1乃至請求項3のいずれかに記載のヒープ領域とし、

前記範囲設定ステップは、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、

前記第1ポインタ更新ステップは、

前記限定範囲内の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内に割当てられた、前記スタック領域を除く該データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、

前記限定範囲外の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記第3ポインタ更新ステップは、

前記ヒープ領域内に割当てられた前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、前記限定範囲内の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の該関数フレームを、該ヒープ領域内の利用可能領域に移動させ、

前記限定範囲外の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記障壁設定ステップは、前記第3ポインタ更新処理による処理を中断して、前記主プログラムを実行するとき、前記障壁を、該第3ポインタ更新処理による処理が最後に行われた前記関数フレームに設定することを特徴とする請求項1乃至請求項3のいずれかに記載のメモリ管理方法。

【請求項6】

メモリ中に、動的にデータを生成するプログラム（以下、主プログラム）の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ（以下、オブジェクト）を記録するためのヒープ領域とが設けられ、

主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域（以下、利用可能領域）を生成すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理装置において、

前記オブジェクトは、オブジェクトの複製先の位置又は複製元の位置を記録するための複製位置ポインタをさらに含み、

前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、

前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定手段と、

前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、該複製元オブジェクト及び該複製先オブジェクトの複製位置ポインタの参照先を互いのオブジェクトの位置に設定するオブジェクト複製手段と、

前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブ

10

20

30

40

50

ジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第1ポインタ更新手段と、

前記限定範囲の領域を利用可能領域にする限定範囲利用可能化手段と、

前記オブジェクト複製手段を実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断手段と、

前記複製判断手段により書込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書込みを行う複製書込み手段と、を備え、

前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、

前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第3ポインタ更新手段と、

前記第3ポインタ更新手段によるオブジェクト位置ポインタの更新処理を中断するときに、該第3ポインタ更新手段による処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定する障壁設定手段と、をさらに備え、

前記第1ポインタ更新手段は、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記障壁が、関数の実行完了後、前記第3ポインタ更新手段による処理を再開させることを特徴とするメモリ管理装置。

【請求項7】

メモリ中に、動的にデータを生成するプログラム（以下、主プログラム）の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ（以下、オブジェクト）を記録するためのヒープ領域とが設けられ、

主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域（以下、利用可能領域）を生成すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理装置において、

前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、

前記メモリ中に、オブジェクトの位置を記録するオブジェクト位置記録領域がさらに設けられ、

前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定手段と、

前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、前記オブジェクト位置記録領域に、該複製元オブジェクトの位置及び該複製先オブジェクトの位置を記録するオブジェクト複製手段と、

前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第1ポインタ更新手段と、

前記限定範囲の領域を利用可能領域にする限定範囲利用可能化手段と、

前記オブジェクト複製手段を実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断手段と、

10

20

30

40

50

前記複製判断手段により書込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書込みを行う複製書込み手段と、を備え、

前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、

前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第3ポインタ更新手段と、

10

前記第3ポインタ更新手段によるオブジェクト位置ポインタの更新処理を中断するとき、該第3ポインタ更新手段による処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定する障壁設定手段と、をさらに備え、

前記第1ポインタ更新手段は、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記障壁が、関数の実行完了後、前記第3ポインタ更新手段による処理を再開させることを特徴とするメモリ管理装置。

【請求項8】

前記オブジェクトは、前記データ要素として前記オブジェクト位置ポインタをさらに含み、

20

前記複製元オブジェクトの位置を参照する前記ヒープ領域内のオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第2ポインタ更新手段と、

前記第2ポインタ更新手段を実行中に、前記主プログラムが前記ヒープ領域に前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの書込み処理を行うとき、又は書込み処理を行った後、該オブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に変更する参照先変更手段と、

前記第2ポインタ更新手段の実行中に、前記主プログラムが2つのオブジェクト位置ポインタの参照先が同じオブジェクトの位置であるかどうかの検査処理を行うとき、該2つのオブジェクト位置ポインタのうち的一方が前記複製元オブジェクトの位置を、他方が前記複製先オブジェクトの位置を参照している場合、同一のオブジェクトを参照していると判断する同一判断手段と、をさらに備え、

30

前記複製判断手段は、前記オブジェクト複製手段、又は前記第2ポインタ更新手段を実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断することを特徴とする請求項6又は請求項7に記載のメモリ管理装置。

【請求項9】

前記ヒープ領域内に、前記スタック領域を除く前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項6乃至請求項8のいずれかに記載のヒープ領域とし、

40

前記範囲設定手段は、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、

前記第1ポインタ更新手段は、

前記限定範囲内の前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の前記データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、

前記限定範囲及び前記スタック領域を除く前記データ記録領域に記録されている、前記

50

複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新することを特徴とする請求項 6 乃至請求項 8 のいずれかに記載のメモリ管理装置。

【請求項 10】

前記ヒープ領域内に、前記スタック領域を含む前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項 6 乃至請求項 8 のいずれかに記載のヒープ領域とし、

前記範囲設定手段は、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、

前記第 1 ポインタ更新手段は、

前記限定範囲内の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内に割当てられた、前記スタック領域を除く該データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、

前記限定範囲外の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記第 3 ポインタ更新手段は、

前記ヒープ領域内に割当てられた前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、前記限定範囲内の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の該関数フレームを、該ヒープ領域内の利用可能領域に移動させ、

前記限定範囲外の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、

前記障壁設定手段は、前記第 3 ポインタ更新処理による処理を中断して、前記主プログラムを実行するとき、前記障壁を、該第 3 ポインタ更新処理による処理が最後に行われた前記関数フレームに設定することを特徴とする請求項 6 乃至請求項 8 のいずれかに記載のメモリ管理装置。

【請求項 11】

請求項 1 乃至請求項 5 のいずれか一項に記載のメモリ管理方法をコンピュータに実行させるためのメモリ管理プログラムが記録されているコンピュータ読取可能な記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、メモリ管理方法、メモリ管理装置、及びメモリ管理プログラムが記録されている記録媒体に関する。

【背景技術】

【0002】

Java（登録商標）言語等の動的にデータを生成するプログラミング言語を用いて作成されたプログラム（以下、主プログラム）は、生成したデータを動的にメモリ領域に記録するので、主プログラムの実行が進むことにより不要となったメモリ領域のデータを消去しなければ主プログラムが利用可能な空き領域（以下、利用可能領域という）が不足してしまう。そのため、従来から、メモリ領域内の不要なデータを消去し、メモリ領域を再利用する方法が知られている。

【0003】

メモリ領域内の不要なデータを消去し、その領域を再利用できるようにする処理はガーベッジコレクション（以下、GC）と呼ばれる。GC 処理によって再利用可能となった利用可能領域を一箇所に集めて連続した領域にする処理はコンパクションと呼ばれる。メモ

10

20

30

40

50

り領域を再利用できるようにするこれら一連の処理は、主プログラムを実行する場合に不可欠な処理である。主プログラムが実時間性を必要とする処理を実行する場合、GC処理及びコンパクション処理は、主プログラムの処理と実質的に並行して実行される必要がある。

【0004】

コンパクション処理は、利用可能領域を一箇所に集めるために、メモリ領域内の必要なデータを移動させた後、移動元データの位置を参照するポインタの参照先を、移動先データの位置に更新する処理を行う。コンパクション処理がデータを移動した後、ポインタの参照先を更新する前に、主プログラムが移動元データを参照したとき、ここにデータが存在しないため、主プログラムが処理異常を発生する。そのため、移動元データの位置を参照する全てのポインタの参照先が、移動先データの位置に更新されるまで、コンパクション処理を中断させることができない。その結果、主プログラムの実時間性が損なわれる。

10

【0005】

GC処理は、本発明者の一人により特許第3530887号に提案された発明によって、主プログラムの実時間性を損なわずに行うことができる。しかし、この発明は、連続した利用可能領域を生成するコンパクション処理を中断することができず、コンパクション処理が終了するまで主プログラムが処理を行うことができない。

【0006】

ところで、従来から、小型機器である携帯電話等には、組込み用Java（登録商標）処理系としてKVM（*K virtual machine*）が多く使われている。KVMには、上述したGC処理及びコンパクション処理を行うプログラム（以下、メモリ管理プログラム）が採用されており、このプログラムは、通常、メモリ領域内の利用可能領域が不足してきたときに、自動的に実行される。その結果、メモリ管理プログラムの処理を中断させることができないので、主プログラムの実行が数ミリ秒～数秒の間中断させられることがある。

20

【0007】

これに対して、従来の携帯電話は、主プログラムが実時間性を必要とする処理、例えば、なめらかな動作が必要なアニメーションやゲームのキャラクタを画面上で動かす処理などを行う場合、主プログラムの実行中にメモリ管理プログラムが自動的に実行されないように、実時間性を必要とする処理を行う直前に、強制的にメモリ管理プログラムを実行させるものがある。これにより、利用可能領域のサイズが大きくなり、しばらくの間はメモリ管理プログラムを実行する必要がないので、主プログラムは、実時間性を損なわずに処理を行うことができる。

30

【0008】

しかし、実時間性を必要とする処理の内容の違いや、携帯電話の機種によるメモリ領域のサイズや主プログラムを処理する実行処理速度の違いにより、強制的にメモリ管理プログラムを実行させるタイミングが異なることがある。そのため、開発者は、これらの違いを考慮して、メモリ管理プログラムを強制的に実行させるタイミングを設定しなければならず、開発に手間がかかり、開発効率が低下するという問題がある。

【0009】

ところで、非特許文献1には、GC処理後のメモリ領域全体に対して、例えば16分割したメモリ領域のうちの1つの領域を限定領域とし、この限定領域内のデータを、限定領域外のメモリ領域に一括して移動させた後、限定領域内のデータを参照するポインタの参照先を一括して更新するコンパクション処理が開示されている。しかし、このコンパクション処理では、データを移動した後に移動元を参照する全てのポインタの参照先を更新するまでの処理を一括して実行する必要があるため、コンパクション処理を中断して主プログラムの処理を行うことができない。そのため、主プログラムの実時間性が損なわれてしまう。

40

【0010】

また、非特許文献2には、メモリ領域を任意の数に分割し、このうちの1つをデータ移

50

動先領域として設定した後、主プログラムがデータ移動先領域を除くメモリ領域にデータを記録する。そして、データ移動先領域を除くメモリ領域のうちの1つをデータ移動元領域とし、このデータ移動元領域のデータをデータ移動先領域に移動させることにより連続した利用可能領域を生成する処理が開示されている。しかし、この処理も、非特許文献1と同様に、コンパクション処理を中断して主プログラムを実行することができないので、主プログラムの実時間性が損なわれてしまう。また、この処理では、主プログラムがメモリ領域にデータが記録する前に、データ移動先領域を設定しなければならないので、メモリ領域に記録されるデータの記録状態等を加味した後に、データ移動先領域を設定することができず、効率良く利用可能領域を生成することができない。

【0011】

10

ところで、一般的に、主プログラムがメモリ領域からデータを読み出す回数は、主プログラムがメモリ領域にデータを書込む回数よりもはるかに多い。そのため、データの読み出し処理に特別な処理を追加すると、データの読み出し時にオーバヘッドが掛かり、主プログラムの処理の実行効率が悪くなることがある。上述した非特許文献2の処理において、各データの先頭に最新のデータの位置を指すフォワーディングポインタを付加し、主プログラムがデータ要素を読み出すとき、フォワーディングポインタを辿って最新のデータの位置からデータ要素を読み出す方法が知られている。しかし、この処理でも、データの読み出し時にオーバヘッドが掛かり、主プログラムの処理の実行効率が悪くなることがある。

【非特許文献1】Parallel Incremental Compaction (US 2004/0128329 A1)

【非特許文献2】Incremental Incrementally Compacting Garbage Collection(SIGPLAN Notice 22(7) pp.253-263)

20

【発明の開示】

【発明が解決しようとする課題】

【0012】

本発明は、上述した問題点を解決するためになされたものであり、連続した利用可能領域を生成する処理を中断して、主プログラムが処理を行うことができるメモリ管理方法、メモリ管理装置、及びメモリ管理プログラムが記録されている記録媒体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】

【0013】

30

上記目的を達成するために請求項1の発明は、メモリ中に、動的にデータを生成するプログラム(以下、主プログラム)の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ(以下、オブジェクト)を記録するためのヒープ領域とが設けられ、主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域(以下、利用可能領域)を生成すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理方法において、前記オブジェクトは、オブジェクトの複製先の位置又は複製元の位置を記録するための複製位置ポインタをさらに含み、前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定ステップと、前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、該複製元オブジェクト及び該複製先オブジェクトの複製位置ポインタの参照先を互いのオブジェクトの位置に設定するオブジェクト複製ステップと、前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第1ポインタ更新ステップと、前記限定範囲の領域を利用可能領域にする限定範囲利用可能化ステップと、前記オブジェクト複製ステップを実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書き込み処理を行うとき、又は書き込み処理を行った後、データの書き込み先の記録領域が、前記複製元オブジェクトの記録

40

50

領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断ステップと、前記複製判断ステップにより書込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書込みを行う複製書込みステップと、を含み、前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第3ポインタ更新ステップと、前記第3ポインタ更新ステップによるオブジェクト位置ポインタの更新処理を中断するときに、該第3ポインタ更新ステップによる処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定する障壁設定ステップと、をさらに備え、前記第1ポインタ更新ステップは、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記障壁が、関数の実行完了後、前記第3ポインタ更新ステップによる処理を再開させるものである。

10

【0014】

請求項2の発明は、メモリ中に、動的にデータを生成するプログラム（以下、主プログラム）の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ（以下、オブジェクト）を記録するためのヒープ領域とが設けられ、主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域（以下、利用可能領域）を生成すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理方法において、前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、前記メモリ中に、オブジェクトの位置を記録するオブジェクト位置記録領域がさらに設けられ、前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定ステップと、前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、前記オブジェクト位置記録領域に、該複製元オブジェクトの位置及び該複製先オブジェクトの位置を記録するオブジェクト複製ステップと、前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第1ポインタ更新ステップと、前記限定範囲の領域を利用可能領域にする限定範囲利用可能化ステップと、前記オブジェクト複製ステップを実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断ステップと、前記複製判断ステップにより書込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書込みを行う複製書込みステップと、を含み、前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、

20

30

40

前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第3ポインタ更新ステップと、前記第3ポインタ更新ステップによるオブジェクト位置ポインタの更新処理を中断するときに、該第3ポインタ更新ステップによる処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定す

50

る障壁設定ステップと、をさらに備え、前記第1ポインタ更新ステップは、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記障壁が、関数の実行完了後、前記第3ポインタ更新ステップによる処理を再開させるのである。

【0015】

請求項3の発明は、請求項1又は請求項2に記載のメモリ管理方法において、前記オブジェクトは、前記データ要素として前記オブジェクト位置ポインタをさらに含み、前記複製元オブジェクトの位置を参照する前記ヒープ領域内のオブジェクト位置ポインタの参照先を、前記オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する第2ポインタ更新ステップと、前記第2ポインタ更新ステップを実行中に、前記主プログラムが前記ヒープ領域に前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの書込み処理を行うとき、又は書込み処理を行った後、該オブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に変更する参照先変更ステップと、前記第2ポインタ更新ステップの実行中に、前記主プログラムが2つのオブジェクト位置ポインタの参照先が同じオブジェクトの位置であるかどうかの検査処理を行うとき、該2つのオブジェクト位置ポインタのうち一方が前記複製元オブジェクトの位置を、他方が前記複製先オブジェクトの位置を参照している場合、同一のオブジェクトを参照していると判断する同一判断ステップと、をさらに備え、前記複製判断ステップは、前記オブジェクト複製ステップ、又は前記第2ポインタ更新ステップを実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断するものである。

【0016】

請求項4の発明は、請求項1乃至請求項3のいずれかに記載のメモリ管理方法において、前記ヒープ領域内に、前記スタック領域を除く前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項1乃至請求項3のいずれかに記載のヒープ領域とし、前記範囲設定ステップは、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、前記第1ポインタ更新ステップは、前記限定範囲内の前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の前記データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、前記限定範囲及び前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新するものである。

【0017】

請求項5の発明は、請求項1乃至請求項3のいずれかに記載のメモリ管理方法において、前記ヒープ領域内に、前記スタック領域を含む前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項1乃至請求項3のいずれかに記載のヒープ領域とし、前記範囲設定ステップは、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、前記第1ポインタ更新ステップは、前記限定範囲内の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内に割当てられた、前記スタック領域を除く該データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、前記限定範囲外の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記第3ポインタ更新ステップは、前記ヒープ領域内に割当てられた前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、前記限定範囲内の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照す

るオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の該関数フレームを、該ヒープ領域内の利用可能領域に移動させ、前記限定範囲外の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記障壁設定ステップは、前記第3ポインタ更新処理による処理を中断して、前記主プログラムを実行するとき、前記障壁を、該第3ポインタ更新処理による処理が最後に行われた前記関数フレームに設定するものである。

【0018】

請求項6の発明は、メモリ中に、動的にデータを生成するプログラム（以下、主プログラム）の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ（以下、オブジェクト）を記録するためのヒープ領域とが設けられ、主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域（以下、利用可能領域）を生成すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理装置において、前記オブジェクトは、オブジェクトの複製先の位置又は複製元の位置を記録するための複製位置ポインタをさらに含み、前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定手段と、前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、該複製元オブジェクト及び該複製先オブジェクトの複製位置ポインタの参照先を互いのオブジェクトの位置に設定するオブジェクト複製手段と、前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第1ポインタ更新手段と、前記限定範囲の領域を利用可能領域にする限定範囲利用可能化手段と、前記オブジェクト複製手段を実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書き込み処理を行うとき、又は書き込み処理を行った後、データの書き込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断手段と、前記複製判断手段により書き込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書き込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書き込みを行う複製書き込み手段と、を備え、前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第3ポインタ更新手段と、前記第3ポインタ更新手段によるオブジェクト位置ポインタの更新処理を中断するとき、該第3ポインタ更新手段による処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定する障壁設定手段と、をさらに備え、前記第1ポインタ更新手段は、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記障壁が、関数の実行完了後、前記第3ポインタ更新手段による処理を再開させるものである。

【0019】

請求項7の発明は、メモリ中に、動的にデータを生成するプログラム（以下、主プログラム）の実行等に利用されるデータが記録されたデータ記録領域と、主プログラムが実行中に使用するデータ要素を含むデータ（以下、オブジェクト）を記録するためのヒープ領域とが設けられ、主プログラムの実行が進むことにより不要となった前記ヒープ領域のオブジェクトを消去し、該ヒープ領域に利用可能な空き領域（以下、利用可能領域）を生成

10

20

30

40

50

すると共に、該ヒープ領域に連続した利用可能領域を生成するために、該ヒープ領域に記録されているオブジェクトを移動させるメモリ管理装置において、前記データ記録領域に、前記オブジェクトの位置を参照するオブジェクト位置ポインタが記録され、前記メモリ中に、オブジェクトの位置を記録するオブジェクト位置記録領域がさらに設けられ、前記ヒープ領域に連続した利用可能領域を生成する限定範囲を設定する範囲設定手段と、前記限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、前記オブジェクト位置記録領域に、該複製元オブジェクトの位置及び該複製先オブジェクトの位置を記録するオブジェクト複製手段と、前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第1ポインタ更新手段と、前記限定範囲の領域を利用可能領域にする限定範囲利用可能化手段と、前記オブジェクト複製手段を実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断する複製判断手段と、前記複製判断手段により書込み先が前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであると判断されたとき、前記主プログラムが書込み処理を行うデータを、前記複製元オブジェクトの記録領域及び前記複製先オブジェクトの記録領域の両方に書込みを行う複製書込み手段と、を備え、前記データ記録領域中に、前記主プログラムが実行する関数の実行結果等を一時的に記録する関数フレームが積み上げられて配置されるスタック領域が設けられ、前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、この順に該スタック領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第3ポインタ更新手段と、前記第3ポインタ更新手段によるオブジェクト位置ポインタの更新処理を中断するときに、該第3ポインタ更新手段による処理が最後に行われた前記関数フレームに、関数の実行を制御する障壁を設定する障壁設定手段と、をさらに備え、前記第1ポインタ更新手段は、前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記障壁が、関数の実行完了後、前記第3ポインタ更新手段による処理を再開させるものである。

【0020】

請求項8の発明は、請求項6又は請求項7に記載のメモリ管理装置において、前記オブジェクトは、前記データ要素として前記オブジェクト位置ポインタをさらに含み、前記複製元オブジェクトの位置を参照する前記ヒープ領域内のオブジェクト位置ポインタの参照先を、前記オブジェクト複製手段により複製された複製先オブジェクトの位置に更新する第2ポインタ更新手段と、前記第2ポインタ更新手段を実行中に、前記主プログラムが前記ヒープ領域に前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの書込み処理を行うとき、又は書込み処理を行った後、該オブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に変更する参照先変更手段と、前記第2ポインタ更新手段の実行中に、前記主プログラムが2つのオブジェクト位置ポインタの参照先が同じオブジェクトの位置であるかどうかの検査処理を行うとき、該2つのオブジェクト位置ポインタのうち一方が前記複製元オブジェクトの位置を、他方が前記複製先オブジェクトの位置を参照している場合、同一のオブジェクトを参照していると判断する同一判断手段と、をさらに備え、前記複製判断手段は、前記オブジェクト複製手段、又は前記第2ポインタ更新手段を実行中に、前記主プログラムが前記ヒープ領域に記録されたオブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、前記複製元オブジェクトの記録領域又は前記複製先オブジェクトの記録領域のいずれかであるかを判断するものである。

【0021】

10

20

30

40

50

請求項 9 の発明は、請求項 6 乃至請求項 8 のいずれかに記載のメモリ管理装置において、前記ヒープ領域内に、前記スタック領域を除く前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項 6 乃至請求項 8 のいずれかに記載のヒープ領域とし、前記範囲設定手段は、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、前記第 1 ポインタ更新手段は、前記限定範囲内の前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の前記データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、前記限定範囲及び前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新するものである。

10

【 0 0 2 2 】

請求項 10 の発明は、請求項 6 乃至請求項 8 のいずれかに記載のメモリ管理装置において、前記ヒープ領域内に、前記スタック領域を含む前記データ記録領域の一部又は全部を割当て、該ヒープ領域内の該データ記録領域外の領域を、請求項 6 乃至請求項 8 のいずれかに記載のヒープ領域とし、前記範囲設定手段は、前記ヒープ領域内の前記データ記録領域が割当てられた領域に対して限定範囲を設定することができ、前記第 1 ポインタ更新手段は、前記限定範囲内の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内に割当てられた、前記スタック領域を除く該データ記録領域を、前記ヒープ領域内の利用可能領域に移動させ、前記限定範囲外の前記スタック領域を除く前記データ記録領域に記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記第 3 ポインタ更新手段は、前記ヒープ領域内に割当てられた前記スタック領域内の前記関数フレームを上位から下位に向かって走査し、前記限定範囲内の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新すると共に、該限定範囲内の該関数フレームを、該ヒープ領域内の利用可能領域に移動させ、前記限定範囲外の前記関数フレームに記録されている、前記複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、前記複製先オブジェクトの位置に更新し、前記障壁設定手段は、前記第 3 ポインタ更新処理による処理を中断して、前記主プログラムを実行するときに、前記障壁を、該第 3 ポインタ更新処理による処理が最後に行われた前記関数フレームに設定するものである。

20

30

【 0 0 2 3 】

請求項 11 の発明は、請求項 1 乃至請求項 5 のいずれか一項に記載のメモリ管理方法をコンピュータに実行させるためのメモリ管理プログラムが記録されているコンピュータ読取可能な記録媒体である。

【 発明の効果 】

【 0 0 2 8 】

請求項 1 の発明によれば、主プログラムが複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、書込み処理を行うデータを、複製元データの記録領域及び複製先データの記録領域の両方に書込むことができる。そのため、複製元データと複製先データとのデータの整合性を保つことができ、連続した利用可能領域を生成する処理を中断して、主プログラムが処理を行うことができる。従って、主プログラムの処理の実時間性を損なわずに、ヒープ領域に連続した利用可能領域を生成することができる。また、請求項 1 の発明によれば、オブジェクト位置ポインタの更新処理が最後に行われた関数フレームに障壁を設定し、関数の実行完了後、障壁により、オブジェクト位置ポインタの更新処理が再開される。そのため、スタック領域内のオブジェクト位置ポインタの更新処理を中断して、主プログラムが関数を実行することができ、主プログラムの処理の実時間性を損なわずに、スタック領域のオブ

40

50

ジェクト位置ポインタを更新することができる。

【0029】

請求項2の発明によれば、主プログラムが複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、書込み処理を行うデータを、複製元データの記録領域及び複製先データの記録領域の両方に書込むことができるので、請求項1と同様の効果を得られる。また、オブジェクト位置記録領域を参照して、複製元オブジェクトの位置及び複製先オブジェクトの位置や、複製元オブジェクトの記録領域及び複製先オブジェクトの記録領域を取得することができるので、オブジェクトは複製位置ポインタを有する必要がない。そのため、ヒープ領域の使用効率が良くなる。また、請求項2の発明によれば、オブジェクト位置ポインタの更新処理が最後に行われた関数フレームに障壁を設定し、関数の実行完了後、障壁により、オブジェクト位置ポインタの更新処理が再開される。そのため、スタック領域内のオブジェクト位置ポインタの更新処理を中断して、主プログラムが関数を実行することができ、主プログラムの処理の実時間性を損なわずに、スタック領域のオブジェクト位置ポインタを更新することができる。

10

【0030】

請求項3の発明によれば、主プログラムがヒープ領域に複製元オブジェクトの位置を参照するオブジェクト位置ポインタの書込み処理を行うとき、又は書込み処理を行った後、オブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に変更することができる。そのため、これらの処理（コンパクション処理）終了後にオブジェクト位置ポインタの参照先にデータが存在しなくなることを防ぐことができるので、主プログラムが処理異常を発生することを防ぐことができ、連続した利用可能領域を生成する処理を中断して、主プログラムが処理を行うことができる。従って、主プログラムの処理の実時間性を損なわずに、ヒープ領域に連続した利用可能領域を生成することができる。

20

【0031】

請求項4の発明によれば、ヒープ領域内に、スタック領域を除くデータ記録領域を割当て、ヒープ領域内のデータ記録領域外の領域を、請求項1乃至請求項3のいずれかに記載のヒープ領域とするので、オブジェクト位置ポインタの更新処理が最後に行われた関数フレームに障壁を設定し、関数の実行完了後、障壁により、オブジェクト位置ポインタの更新処理が再開される。そのため、スタック領域内のオブジェクト位置ポインタの更新処理を中断して、主プログラムが関数を実行することができ、主プログラムの処理の実時間性を損なわずに、スタック領域のオブジェクト位置ポインタを更新することができる。

30

【0032】

請求項5の発明によれば、ヒープ領域内に、スタック領域を含むデータ記録領域を割当て、ヒープ領域内のデータ記録領域外の領域を、請求項1乃至請求項3のいずれかに記載のヒープ領域とするので、オブジェクト位置ポインタの更新処理が最後に行われた関数フレームに障壁を設定し、関数の実行完了後、障壁により、オブジェクト位置ポインタの更新処理が再開される。そのため、スタック領域内のオブジェクト位置ポインタの更新処理を中断して、主プログラムが関数を実行することができ、主プログラムの処理の実時間性を損なわずに、スタック領域のオブジェクト位置ポインタを更新することができる。また、限定範囲内のスタック領域をヒープ領域内の利用可能領域に移動させる処理を中断して、主プログラムが処理を行うことができる。そのため、主プログラムの処理の実時間性を損なわずに、限定範囲内のスタック領域をヒープ領域内の利用可能領域に移動させる処理を行うことができる。

40

【0033】

請求項6の発明によれば、請求項1の発明と同様の効果を得ることができる。

【0034】

請求項7の発明によれば、請求項2の発明と同様の効果を得ることができる。

【0035】

請求項8の発明によれば、請求項3の発明と同様の効果を得ることができる。

50

【 0 0 3 6 】

請求項 9 の発明によれば、ヒープ領域内に、スタック領域を除くデータ記録領域を割当て、ヒープ領域内のデータ記録領域外の領域を、請求項 6 乃至請求項 8 のいずれかに記載のヒープ領域とするので、請求項 4 の発明と同様の効果を得ることができる。

【 0 0 3 7 】

請求項 1 0 の発明によれば、ヒープ領域内に、スタック領域を含むデータ記録領域を割当て、ヒープ領域内のデータ記録領域外の領域を、請求項 6 乃至請求項 8 のいずれかに記載のヒープ領域とするので、請求項 5 の発明と同様の効果を得ることができる。

【 0 0 3 8 】

請求項 1 1 の発明によれば、コンピュータにメモリ管理プログラムを読み取らせて、コンピュータ上でこのプログラムを実行することにより、請求項 1 乃至請求項 5 のいずれか一項に記載の発明と同様の効果が得られる。

【発明を実施するための最良の形態】

【 0 0 4 3 】

以下、本発明の第 1 実施形態に係る携帯電話（メモリ管理装置）について、図 1 乃至図 3 を参照して説明する。以下の実施形態では、本発明のメモリ管理装置を携帯電話に適用した場合について説明する。図 1 は、携帯電話 1 の構成を示し、図 2 はフラッシュメモリ（記録媒体）1 7 の構成を示し、図 3 は、データ記録領域 2 1 の構成を示す。携帯電話 1 は、動的なデータを生成する J a v a（登録商標）言語を用いて作成されたプログラム（以下、主プログラム）9 1、及び本発明のメモリ管理プログラム 9 3 の実行が可能である。

【 0 0 4 4 】

携帯電話 1 は、音声を入力するためのマイク等の入力部 1 2 と、音声を出力するためのスピーカ等の出力部 1 3 と、ユーザによって操作されるキーを有する操作部 1 4 と、音声データ等の送受信を行うための通信部 1 5 と、メニューや通信状態等を表示するための表示部 1 6 とを備える、また、携帯電話 1 は、プログラム及びデータの情報が記録されたフラッシュメモリ 1 7 と、各種情報を一時的に記録するメモリ部（メモリ）1 8 と、装置各部の制御を行うための C P U 1 1 と、を備える。

【 0 0 4 5 】

フラッシュメモリ 1 7 には、図 2 に示すように、主プログラム 9 1 と、主プログラム 9 1 の実行に必要な J a v a（登録商標） V M（J a v a（登録商標） V i r t u a l M a c h i n e）等の実行環境プログラム 9 2 と、実行環境プログラム 9 2 の一部であるメモリ管理プログラム 9 3 と、各種設定情報等のデータ 9 4 等と、が記録されている。携帯電話 1 は、メモリ管理プログラム 9 3 をメモリ部 1 8 に読み取らせて、C P U 1 1 で実行することにより、本発明のメモリ管理装置として動作する。なお、メモリ管理プログラム 9 3 は、フラッシュメモリ 1 7 中に、主プログラム 9 1 の一部として記録されていてもよいし、独立したプログラムとして記録されていてもよい。

【 0 0 4 6 】

メモリ部 1 8 には、主プログラム 9 1 の実行等に利用されるデータを記録するためのデータ記録領域 2 1 と、主プログラム 9 1 が実行中に使用するデータ要素を含むデータ（以下、オブジェクト）を記録するためのヒープ領域 2 2 と、フリーリスト 2 3 と、が設けられている。データ記録領域 2 1 は、特別なオブジェクト等の位置を参照するオブジェクト位置ポインタや、スタック領域 2 7 の位置を示すポインタや、プログラムの実行位置を示すデータ等が記録されている固定データ記録領域（図示せず）や、一時的なデータを記録する一時データ記録領域（図示せず）などを記録している。

【 0 0 4 7 】

ヒープ領域 2 2 は、主プログラム 9 1 の実行によりオブジェクトが記録され、主プログラム 9 1 の実行過程において利用可能な空き領域（以下、利用可能領域）が次第に不足する。そのため、ヒープ領域 2 2 は、メモリ管理プログラム 9 3 により、主プログラム 9 1 の実行が進むことにより不要になったオブジェクトを消去するガーベッジコレクション（

10

20

30

40

50

以下、GC)処理と、GC処理の後に必要に応じて連続した利用可能領域を生成するためのコンパクション処理とを含むメモリ管理処理が行われる。フリーリスト23は、ヒープ領域22内の利用可能領域の位置が記録されている。

【0048】

次に、スタック領域27の構成について、図4を参照して説明する。図4は、スタック領域27に記録された関数フレームを概念的に示したものである。関数フレームは、主プログラム91の実行に必要な関数(Java(登録商標)言語ではメソッドと呼ばれる)を記録する領域である。スタック領域27は、主プログラム91が実行する関数の実行結果等を一時的に記録する関数フレーム60a~60cが積み上げられて配置されている。実行中の関数フレームはカレントフレームと呼ばれ、図4では、関数フレーム60aがカレントフレーム70となる。

10

【0049】

主プログラム91の関数は、実行中の関数が、実行すべき関数を呼び出して処理を行う。関数が関数を呼び出すとき、スタック領域27内に確保された実行中の関数の関数フレームに、呼び出し元の関数は、新たな関数フレームを積み上げる形で確保し、呼び出し先の関数を実行する。そして、呼び出し先の関数の実行完了後、呼び出し元の関数の関数フレームは不要であるため、積み上げられている不要な関数フレームを破棄する。なお、スタック領域27内に積み上げられる関数フレームは、それぞれの位置がデータ記録領域21内の記録領域において隣接している必要はない。

【0050】

20

次に、ヒープ領域22に記録されるオブジェクトの構成について、図5を参照して説明する。図5は、ヒープ領域22に記録されるオブジェクト30の一例を概念的に示すものである。オブジェクト30は、複製位置ポインタ50の参照先アドレス111と、データ要素121~123と、データ要素131と、を含むものから構成される。データ要素121~123は、数値や文字などの値が記録されているものであり、データ要素131はオブジェクト位置ポインタ40の参照先アドレスが記録されているものである。

【0051】

オブジェクト位置ポインタ40は、データ要素として、他のオブジェクトを参照するものであり、データ要素121~123とオブジェクト位置ポインタ40とは、主プログラム91によって使用される。複製位置ポインタ50は、オブジェクト30の複製先の位置又は複製元の位置を記録するためのものである。複製位置ポインタ50の参照先アドレス111は、オブジェクト30の先頭に記録されており、主プログラム91によって使用されない。オブジェクトが複製されていない場合、複製位置ポインタ50の参照先アドレス111には、「オブジェクトが複製されていない」という旨を示す特別な値が記録される。

30

【0052】

次に、メモリ管理プログラム93によるメモリ管理処理について図6を参照して説明する。携帯電話1は、ヒープ領域22の利用可能領域が不足したときに、メモリ管理プログラム93を実行することによりメモリ管理処理を行う。携帯電話1は、ヒープ領域22の利用可能領域がある基準値以下になったとき、ヒープ領域22の利用可能領域が不足していると判断する。なお、ヒープ領域22の利用可能領域の不足を判断する基準値は、任意に設定可能である。

40

【0053】

まず、メモリ管理プログラム93は、ヒープ領域22内の不要なオブジェクトを消去し、利用可能領域を生成するGC処理を行った後(S1)、ヒープ領域22内の利用可能領域をフリーリスト23に登録する(S2)。なお、GC処理は、本発明者により特許第3530887号に提案された発明によって、主プログラムの実時間性を損なわずに行うことができるが、これに限られず、主プログラムの実時間性を損なわずにGC処理を行うことができる他の方法を用いてGC処理を行ってもよい。

【0054】

50

そして、上記 S 2 処理の後、ヒープ領域 2 2 内に連続した利用可能領域を生成するためにコンパクション処理を行う (S 3)。コンパクション処理の詳細については後述する。上記 S 3 処理の後、メモリ管理プログラム 9 2 は、データ記録領域 2 1 内のオブジェクト位置ポインタ更新処理を行う (S 4)。オブジェクト位置ポインタ更新処理の詳細については後述する。上記 S 4 処理の後、ヒープ領域 2 2 内の限定範囲 (図 7 参照) を連続した利用可能領域としてフリーリスト 2 3 に登録し (S 5 : 限定範囲利用可能化ステップ)、メモリ管理処理を終了する。

【 0 0 5 5 】

次に、上記 S 3 のコンパクション処理について、図 7 を参照して説明する。まず、メモリ管理プログラム 9 3 は、ヒープ領域 2 2 中に連続した利用可能領域を生成する限定範囲を設定し (S 1 1 : 範囲設定ステップ)、この限定範囲内に含まれるオブジェクト (以下、複製元オブジェクト) をこの限定範囲外の利用可能領域に複製先のオブジェクト (以下、複製先オブジェクト) として複製すると共に、複製元オブジェクト及び複製先オブジェクトの複製位置ポインタの参照先を、互いのオブジェクトの位置に設定する (S 1 2 : オブジェクト複製ステップ)。なお、複製元オブジェクト及び複製先オブジェクトは、オブジェクトの一種であり、便宜上、上記のように呼ぶ。

10

【 0 0 5 6 】

そして、ヒープ領域 2 2 内の複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、上記 S 1 2 により複製された複製先オブジェクトの位置に更新し (S 1 3 : 第 2 ポインタ更新ステップ)、コンパクション処理を終了する。なお、本処理では、上記 S 1 1 により限定範囲を設定した後に、上記 S 1 2 の処理を実行しているが、これに限らず、上記 S 1 2 の処理中に限定範囲を設定してもよい。また、上記 S 1 1 による限定範囲設定処理は、メモリ管理プログラム 9 3 に含まれない独立したプログラムであってもよい。上記 S 1 1 による限定範囲設定処理が独立したプログラムである場合、このプログラムは、メモリ管理プログラム 9 3 の実行前に実行される。

20

【 0 0 5 7 】

次に、上記 S 4 のオブジェクト位置ポインタ更新処理について、図 8 を参照して説明する。メモリ管理プログラム 9 3 は、データ記録領域 2 1 に記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新し (S 2 1 : 第 1 ポインタ更新ステップ)、処理を終了する。データ記録領域 2 1 内のスタック領域 2 7 のオブジェクト位置ポインタ更新処理については、メモリ管理プログラム 9 3 が、スタック領域 2 7 の関数フレームを上位から下位に向かって走査する。そして、この順にスタック領域 2 7 に記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、オブジェクト複製ステップにより複製された複製先オブジェクトの位置に更新する (S 2 1 : 第 3 ポインタ更新ステップ)。

30

【 0 0 5 8 】

次に、メモリ管理処理中のヒープ領域 2 2 の状態について、図 9 を参照して説明する。図 9 (a) (b) (c) は、メモリ管理処理を行った際のヒープ領域 2 2 の状態を概念的に示したものであり、図 9 (a) は上記 S 1 1 の処理を、図 9 (b) は上記 S 1 2 の処理を、図 9 (c) は上記 S 1 3 の処理を行った際の状態に相当する。

40

【 0 0 5 9 】

図 9 において、3 1 ~ 3 7、3 4 a、3 5 a はオブジェクト、4 1 ~ 4 4 はオブジェクト位置ポインタ、5 1 ~ 5 4 は複製位置ポインタ、R は限定範囲を示す。なお、3 4 と 3 5 は複製元オブジェクトになっており、3 4 a と 3 5 a は複製先オブジェクトになっている。

【 0 0 6 0 】

図 9 (a) に示すように、限定範囲 R には、複製元オブジェクト 3 4、3 5 が含まれ、オブジェクト 3 1 のオブジェクト位置ポインタ 4 1 は複製元オブジェクト 3 4 の位置を参照し、オブジェクト 3 7 のオブジェクト位置ポインタ 4 2 は複製元オブジェクト 3 5 の位

50

置を参照している。オブジェクト32～36は、オブジェクト位置ポイントを保持していない。また、オブジェクト31～37のオブジェクトは複製されておらず、オブジェクト31～37の複製位置ポイントは他のオブジェクトの位置を参照していない。

【0061】

複製元オブジェクト34、35を複製すると、図9(b)に示すように、ヒープ領域22内の利用可能領域に複製先オブジェクト34a、35aとして複製される。このとき、複製先オブジェクト34aの複製位置ポイント51が複製元オブジェクト34の位置を参照し、複製元オブジェクト34の複製位置ポイント53が複製先オブジェクト34aの位置を参照するように複製位置ポイント51及び複製位置ポイント53が設定される。また、複製先オブジェクト35aの複製位置ポイント52が複製元オブジェクト35の位置を参照し、複製元オブジェクト35の複製位置ポイント54が複製先オブジェクト35aの位置を参照するように複製位置ポイント52及び複製位置ポイント54が設定される。

10

【0062】

そして、ヒープ領域22を走査して、オブジェクト31～37のオブジェクト位置ポイントが、限定範囲R内の複製元オブジェクト34、35を参照している場合、このオブジェクト位置ポイントの参照先を複製先オブジェクト34a、35aの位置に更新する。ここでは、図9(c)に示すように、オブジェクト31のオブジェクト位置ポイント43の参照先を複製先オブジェクト34aの位置に、オブジェクト37のオブジェクト位置ポイント44の参照先を複製先オブジェクト35aの位置に更新している。

【0063】

20

次に、上述したコンパクション処理の上記S12及び上記S13の処理中に、主プログラム91がヒープ領域22にデータの書込みを行う際の処理について、図10を参照して説明する。まず、メモリ管理プログラム93は、主プログラム91がヒープ領域22にデータの書込み処理を行うとき、又は書込み処理を行った後、データの書込み先の記録領域が、複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域のいずれかであるかを判断する(S31:複製判断ステップ)。そして、データの書込み先が複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域のいずれかであると判断されたとき(S32でYES)、メモリ管理プログラム93は、主プログラム91が書込むデータを、複製元オブジェクトの記録領域及び複製先オブジェクトの記録領域の両方に書込む(S33:複製書込みステップ)。

30

【0064】

一方、上記S31により、データの書込み先が複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域のいずれかでもない判断されたとき(S32でNO)、上記S33の処理を行わず、主プログラム91がデータを書込む(S34)。なお、上記S31乃至上記S33の処理は、主プログラム91が、ヒープ領域22に記録されたオブジェクトの記録領域にデータの書込み処理を行った後、メモリ管理プログラム93は、主プログラム91により複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域のいずれかにだけデータが書込まれたかどうかを判断し(上記S31に相当)、この判断の結果、データの書込みがあった場合(上記S32でYESに相当)、メモリ管理プログラム93は、データが書込まれていない複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域のいずれかに、主プログラム91により書込まれたデータを書込む(上記S33に相当)処理により実現されてもよい。

40

【0065】

次に、上述したコンパクション処理の上記S13の処理中に、主プログラム91がヒープ領域22に複製元オブジェクトの位置を参照するオブジェクト位置ポイントの書込みを行う際の処理について、図11を参照して説明する。まず、メモリ管理プログラム93は、主プログラム91が、ヒープ領域22にオブジェクト位置ポイントの書込み処理を行うとき、又は書込み処理を行った後、このオブジェクト位置ポイントの参照先が、複製元オブジェクトの位置であるかどうかを判断する(S41)。そして、このオブジェクト位置ポイントの参照先が、ヒープ領域22の複製元オブジェクトの位置である場合(S42で

50

YES)、このオブジェクト位置ポイントの参照先を、複製先オブジェクトの位置に変更し(S 4 3)、参照先を変更したオブジェクト位置ポイントを上記図 1 0 の処理手順に従って書込む(S 4 4)。このとき、参照先を変更したオブジェクト位置ポイントは、上述した図 1 0 の処理中のデータに相当する。

【 0 0 6 6 】

一方、上記 S 4 1 により、オブジェクト位置ポイントの参照先が複製元オブジェクトの位置でない場合(S 4 2 で NO)、上記 S 4 3 の処理は行わず、主プログラム 9 1 がオブジェクト位置ポイントをヒープ領域 2 2 に、上記図 1 0 の処理手順に従って書込む(S 4 4)。なお、図 1 0 及び図 1 1 の処理は、主プログラム 9 1 又は実行環境プログラム 9 2 が行ってもよい。

10

【 0 0 6 7 】

上述したように携帯電話 1 は、主プログラム 9 1 が複製元オブジェクトの記録領域又は複製先オブジェクトの記録領域にデータの書込み処理を行うとき、又は書込み処理を行った後、書込み処理を行うデータを、複製元データの記録領域及び複製先データの記録領域の両方に書込むことができる。そのため、複製元オブジェクトと複製先オブジェクトとのデータの整合性を保つことができる。従って、オブジェクトの複製処理を中断して主プログラム 9 1 が処理を行うことができる。

【 0 0 6 8 】

また、主プログラム 9 1 が、ヒープ領域 2 2 に複製元オブジェクトの位置を参照するオブジェクト位置ポイントの書込み処理を行うとき、又は書込み処理を行った後、オブジェクト位置ポイントの参照先を、複製先オブジェクトの位置に変更することができる。そのため、オブジェクト位置ポイントの参照先にデータが存在しないことを防ぐことができるので、主プログラム 9 1 が処理異常を発生することを防ぐことができる。従って、携帯電話 1 は、コンパクション処理を中断して、主プログラム 9 1 の処理を行うことができ、主プログラム 9 1 の処理の実時間性を損なわずに、ヒープ領域 2 2 に連続した利用可能領域を生成することができる。

20

【 0 0 6 9 】

次に、スタック領域 2 7 に対する上述したオブジェクト位置ポイント更新処理の上記 S 2 1 の処理中に、主プログラム 9 1 が関数を実行する際の処理について、図 1 2 を参照して説明する。まず、メモリ管理プログラム 9 3 がオブジェクト位置ポイントの更新処理を中断するときに、メモリ管理プログラム 9 3 は、オブジェクト位置ポイントの更新処理が最後に行われた関数フレームに、関数の実行を制御する障壁を設定する(S 5 1 : 障壁設定ステップ)。そして、主プログラム 9 1 が関数を実行し、この関数の実行が完了して関数フレームを破棄するとき(S 5 2)、主プログラム 9 1 が、この関数フレームに障壁が設定されているどうか判断する(S 5 3)。

30

【 0 0 7 0 】

上記 S 5 3 の結果、障壁が設定されている場合(S 5 4 で YES)、障壁がメモリ管理プログラム 9 3 による、スタック領域 2 7 のオブジェクト位置ポイントの更新処理を再開させ、メモリ管理プログラム 9 3 は、この関数フレームの下位に位置する関数フレーム内の複製元オブジェクトを参照するオブジェクト位置ポイントの参照先を、複製先オブジェクトの位置に更新する(S 5 5)。そして、オブジェクト位置ポイントの更新処理が最後に行われた関数フレームに、関数の実行を制御する障壁を再設定し、カレントフレームを下位の関数フレームに移動し、カレントフレームが移動した関数フレームの関数を実行する(S 5 6)。

40

【 0 0 7 1 】

一方、上記 S 5 3 により、関数フレームに障壁が設定されていない場合(S 5 4 で NO)、上記 S 5 5 の処理は行わず、上記 S 5 6 の処理を行う。なお、上述した障壁の設定処理、及び障壁が設定されているかどうかの判断処理は、実行環境プログラム 9 2 が行ってもよい。また、本処理によりスタック領域 2 7 のオブジェクト位置ポイントの更新が行われる場合、スタック領域 2 7 を除くデータ記録領域 2 1 に対して、上述した図 8 によるオ

50

プロジェクト位置ポインタの更新処理が行われる。

【 0 0 7 2 】

次に、上記図 1 2 に示したオブジェクト位置ポインタ処理中のスタック領域 2 7 の状態について、図 1 3 を参照して説明する。図 1 3 は、スタック領域 2 7 を概念的に示した図であり、(a) は、上記 S 5 2 の処理に相当し、(b) は上記 S 5 6 の処理に相当する。図 1 3 において、6 1 ~ 6 4 は関数フレーム、7 0 はカレントフレーム、8 0 は障壁を示す。

【 0 0 7 3 】

図 1 3 (a) に示すように、カレントフレーム 7 0 は関数フレーム 6 1 に位置し、このとき、関数フレーム 6 1 のオブジェクト位置ポインタは既に更新されている。この関数フレーム 6 1 の関数の実行が完了したとき、設定されている障壁 8 0 により、メモリ管理プログラム 9 3 が関数フレーム 6 2 のオブジェクト位置ポインタの更新処理を行う。そして、関数フレーム 6 2 のオブジェクト位置ポインタの更新後、図 1 3 (b) に示すように、メモリ管理プログラム 9 3 は、関数フレーム 6 2 に対して障壁 8 0 を設定し、カレントフレーム 7 0 を関数フレーム 6 2 に移動させ、主プログラム 9 1 が関数フレーム 6 2 の関数を実行する。なお、上記 S 5 5 において、複数の関数フレームのオブジェクト位置ポインタを更新してもよい。

【 0 0 7 4 】

上述したように、携帯電話 1 は、オブジェクト位置ポインタの更新処理が最後に行われた関数フレームに障壁を設定し、関数の実行完了後、障壁により、オブジェクト位置ポインタの更新処理を再開される。そのため、オブジェクト位置ポインタの更新処理を中断して、主プログラムが実行する関数を実行することができ、主プログラム 9 1 による関数の処理の実時間性を損なわずに、スタック領域のオブジェクト位置ポインタを更新することができる。また、スタック領域 2 7 内のオブジェクト位置ポインタの更新処理を分割して行うことができるので、主プログラム 9 1 の処理の実時間性を損なわない。

【 0 0 7 5 】

次に、上述したコンパクション処理の上記 S 1 3 の処理中における、2 つのオブジェクト位置ポインタの比較処理について、図 1 4 を参照して説明する。ここでは、主プログラム 9 1 が、ヒープ領域 2 2 から読み出し処理などを行うことにより、2 つのオブジェクト位置ポインタを得ているものとする。まず、主プログラム 9 1 は、2 つのオブジェクト位置ポインタを比較し (S 6 1)、同一のオブジェクトを参照している場合 (S 6 2 で Y E S)、同一のオブジェクトを参照しているものと判断し (S 6 5)、同一判断処理を終了する。

【 0 0 7 6 】

一方、同一のオブジェクトを参照していない場合 (S 6 2 で N O)、一方のオブジェクト位置ポインタが参照するオブジェクトの複製位置ポインタと他方のオブジェクト位置ポインタを比較する (S 6 3)。そして、これらのポインタが同一のオブジェクトを参照しているとき (S 6 4 で Y E S)、上記 S 6 5 の判断を行い、同一判断処理を終了する。また、上記 S 6 3 で、これらのポインタが同一のオブジェクトを参照していないとき (S 6 4 で N O)、異なるオブジェクトを参照しているものと判断し (S 6 6)、同一判断処理は終了する。なお、上記図 1 4 の処理は実行環境プログラム 9 2 が行ってもよい。

【 0 0 7 7 】

上記 S 6 1 の結果、2 つのオブジェクト位置ポインタのうち的一方が複製元オブジェクトの位置を、他方が複製先オブジェクトの位置を参照している場合 (S 6 2 で Y E S)、メモリ管理プログラム 9 3 は、2 つのオブジェクト位置ポインタが同一のオブジェクトを参照していると判断し (S 6 3 : 同一判断ステップ)、処理を終了する。一方、上記 S 6 1 の結果、2 つのオブジェクト位置ポインタのうち的一方が複製元オブジェクトの位置を、他方が複製先オブジェクトの位置を参照していない場合 (S 6 2 で N O)、上記 S 6 3 の処理は行わないで処理を終了する。これにより、携帯電話 1 は、2 つのオブジェクト位置ポインタが、同一のオブジェクトを参照しているかどうか判断することができる。

10

20

30

40

50

【 0 0 7 8 】

上記のように、本実施形態に係る携帯電話 1 によれば、ヒープ領域 2 2 内に連続した利用可能領域を生成する処理を中断して、主プログラム 9 1 が処理を行うことができるので、主プログラム 9 1 の処理の実時間性を損なわない。そのため、開発者が、従来のように、携帯電話の機種毎によるメモリ部 1 8 のサイズの違い等により、メモリ管理プログラム 9 3 を強制的に実行させるタイミングを設定する必要がないので、開発効率が良くなり、開発期間が短縮する。

【 0 0 7 9 】

また、主プログラム 9 1 がヒープ領域 2 2 にデータやオブジェクト位置ポインタの書込みを行うときに、書込み先の記録領域と書込むポインタの参照先を判断する処理を追加する。主プログラム 9 1 は、一般的に、ヒープ領域 2 2 への読出し処理よりも書込み処理を行う回数の方が少ないので、ヒープ領域 2 2 からオブジェクトを読出す際に処理を追加する場合よりも、主プログラム 9 1 の実行時にオーバヘッドが小さい。

10

【 0 0 8 0 】

また、GC 処理を行った後に、限定範囲の設定をすることができるので、利用可能領域や利用しているデータ量等のさまざまな要因を加味して限定範囲の設定をすることができるので、効率良く利用可能領域を生成することができる。

【 0 0 8 1 】

次に、本発明の第 2 実施形態に係る携帯電話について、図 1 5 を参照して説明する。図 1 5 は、本実施形態の携帯電話 1 の構成を示す。携帯電話 1 は、メモリ部 1 8 が位置記録リスト（オブジェクト位置記録領域）2 4 をさらに有していること、及びヒープ領域 2 2 に記録されるオブジェクトが複製位置ポインタを含んでいないこと以外は、第 1 実施形態に示した携帯電話 1 と同様の構成である。位置記録リスト 2 4 は、第 1 実施形態における複製位置ポインタに相当する、ヒープ領域 2 2 内のオブジェクトの位置を記録するためのリストである。ヒープ領域 2 2 内に記録されているオブジェクトは、データ要素と、オブジェクト位置ポインタと、を含むものから構成されている。

20

【 0 0 8 2 】

次に、位置記録リスト 2 4 について、図 1 6 を参照して説明する。図 1 6 は、複製元オブジェクトの位置及び複製先オブジェクトの位置が記録されている位置記録リスト 2 4 の一例を示す。図 1 6 に示すように、位置記録リスト 2 4 は、隣接する複数の複製元オブジェクトを複製元オブジェクト群として、隣接する複数の複製先オブジェクトを複製先オブジェクト群としてまとめて記録し、複製元オブジェクト群の起点及び終点の位置と、この複製元オブジェクト群に対応する複製先オブジェクト群の起点及び終点の位置が、ヒープ領域 2 2 内のアドレスを示す数値で記録されている。そのため、複製元オブジェクトの位置、複製先オブジェクトの位置、複製元オブジェクトの記録領域、又は複製先オブジェクトの記録領域の取得を行うとき、メモリ管理プログラム 9 3 は、位置記録リスト 2 4 を参照する。

30

【 0 0 8 3 】

次に、携帯電話 1 のコンパクション処理について、図 1 7 を参照して説明する。まず、メモリ管理プログラム 9 3 は、ヒープ領域 2 2 中に連続した利用可能領域を生成する限定範囲を設定し（S 7 1）、この限定範囲内に含まれる複製元オブジェクトをこの限定範囲外の利用可能領域に複製先オブジェクトとして複製すると共に、位置記録リスト 2 4 に、複製元オブジェクトの位置及び複製先オブジェクトの位置を記録する（S 7 2）。

40

【 0 0 8 4 】

そして、メモリ管理プログラム 9 3 は、位置記録リスト 2 4 を参照して、ヒープ領域 2 2 内の複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、上記 S 7 2 により複製された複製先オブジェクトの位置に更新し（S 7 3）、ヒープ領域 2 2 のコンパクション処理を終了する。

【 0 0 8 5 】

上記のように、本実施形態に係る携帯電話 1 によれば、第 1 実施形態に係る携帯電話と

50

同様の効果を得ることができる。また、位置記録リスト24を参照して、複製元オブジェクトの位置及び複製先オブジェクトの位置や、複製元オブジェクトの記録領域及び複製先オブジェクトの記録領域を取得することができるので、オブジェクトは複製位置ポインタを有する必要がない。そのため、ヒープ領域22の使用効率が良くなる。

【0086】

次に、本発明の第3実施形態に係る携帯電話について、図18及び図19を参照して説明する。図18は、本実施形態の携帯電話1の構成を示し、図19はヒープ領域25内のデータ記録領域26の割当て状態を示す。携帯電話1は、ヒープ領域25内にデータ記録領域26の一部が割り当てられていること以外は、第1実施形態に示した携帯電話1の構成と同様である。スタック領域27は、図19に示すように、ヒープ領域25内に割当てられていない。

10

【0087】

携帯電話1は、ヒープ領域25内のデータ記録領域26外の領域を、第1実施形態に係る携帯電話のヒープ領域とすることができる。メモリ管理プログラム93は、ヒープ領域25内の前記データ記録領域26が割当てられた領域に対しても限定範囲を設定することができる。

【0088】

次に、メモリ管理プログラム93による限定範囲内のデータ記録領域26のコンパクション処理について、図20を参照して説明する。まず、メモリ管理プログラム93は、限定範囲内のデータ記録領域26に記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に更新すると共に、限定範囲内のデータ記録領域26を、ヒープ領域25内の利用可能領域に移動させる(S81)。

20

【0089】

そして、限定範囲及びスタック領域27を除くデータ記録領域26に記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に更新し(S82)、処理を終了する。これにより、ヒープ領域25に割当てられているデータ記録領域26に対してもコンパクション処理を行うことができるので、ヒープ領域25内に連続した利用可能領域を効率良く生成することができる。

【0090】

上記のように、本実施形態に係る携帯電話1によれば、ヒープ領域25にデータ記録領域26の一部が割当てられている場合であっても、第1実施形態の携帯電話と同様の効果を得ることができる。なお、ここでは、ヒープ領域25にデータ記録領域26の一部が割当てられている場合を示したが、ヒープ領域25に、スタック領域27を除くデータ記録領域26の全部が割当てられてもよい。この場合であっても、携帯電話1は、第1実施形態の携帯電話と同様の効果を得ることができる。

30

【0091】

また、本実施形態に係る携帯電話1のメモリ部18が、オブジェクトの位置を記録するための位置記録リスト24をさらに有し、ヒープ領域25に記録されているオブジェクトが複製位置ポインタを有しない場合、本実施形態に係る携帯電話1は、第2実施形態に係る携帯電話と同様の効果を得ることができる。

40

【0092】

次に、本発明の第4実施形態に係る携帯電話について、図21を参照して説明する。図21は、ヒープ領域25内のデータ記録領域26の割当て状態を示す。携帯電話1は、ヒープ領域25内にデータ記録領域26の一部が割り当てられると共に、スタック領域27がヒープ領域25内に割当てられていること以外は、第3実施形態に示した携帯電話1の構成と同様である。

【0093】

次に、メモリ管理プログラム93による限定範囲内のデータ記録領域26のコンパクション処理について、図22を参照して説明する。まず、メモリ管理プログラム93は、限

50

定範囲内のスタック領域 27 を除くデータ記録領域 26 に記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に更新すると共に、限定範囲内に割当てられた、スタック領域 27 を除くデータ記録領域 26 を、ヒープ領域 25 内の利用可能領域に移動させる (S 91)。

【0094】

そして、限定範囲外のスタック領域 27 を除くデータ記録領域 26 に記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に更新し (S 92)、処理を終了する。これにより、ヒープ領域 25 に割当てられている、スタック領域 27 を除くデータ記録領域 26 に対してもコンパクション処理を行うことができるので、ヒープ領域 25 内に連続した利用可能領域を効率良く生成することができる。

10

【0095】

次に、メモリ管理プログラム 93 による限定範囲内のスタック領域 27 のコンパクション処理について、図 23 を参照して説明する。まず、メモリ管理プログラム 93 は、限定範囲内の関数フレームの移動先を決定し (S 101)、ヒープ領域 25 内に割当てられたスタック領域 27 内の関数フレームを上位から下位に向かって走査する (S 102)。そして、限定範囲内の関数フレームに記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に更新すると共に、限定範囲内の関数フレームを、ヒープ領域 25 内の利用可能領域に移動させる (S 103)。

20

【0096】

そして、限定範囲外の関数フレームに記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に更新し (S 104)、処理を終了する。これにより、ヒープ領域 25 内のスタック領域 27 に対してもコンパクション処理を行うことができるので、ヒープ領域 25 内に連続した利用可能領域を効率良く生成することができる。

【0097】

上記のように、本実施形態に係る携帯電話 1 によれば、スタック領域 27 がヒープ領域 25 内に割当てられている場合であっても、第 3 実施形態の携帯電話と同様の効果を得ることができる。なお、ここでは、ヒープ領域 26 にデータ記録領域 26 の一部が割当てられている場合を示したが、ヒープ領域 26 に、データ記録領域 26 の全部が割当てられてもよい。この場合であっても、携帯電話 1 は、第 3 実施形態の携帯電話と同様の効果を得ることができる。また、スタック領域 27 の一部がヒープ領域 25 外に割当てられている場合でもよく、この場合であっても、携帯電話 1 は、第 3 実施形態の携帯電話と同様の効果を得ることができる。

30

【0098】

次に、メモリ管理プログラム 93 をコンピュータ上で実行した場合の主プログラムの最大停止時間の実験結果について説明する。本実験で使用したコンピュータの性能は、CPU が「Xeon 3.2GHz」、メモリが 1GB である。実験方法は、このコンピュータに、ヒープ領域が 5MB である Java (登録商標) 実行環境を実装し、JAKLD (Java (登録商標) Application Kumikomi Lisp Driver: LISP 処理系: 情報処理学会論文誌 プログラミング 44 巻 p1~16 参照) 上で Boyer ベンチマークを実行することにより、メモリ管理プログラム 93 による主プログラムの最大停止時間を測定した。なお、メモリ管理プログラム 93 の GC 処理は、特許 3530887 号で本発明者が提案した処理を利用している。また、本実験では、データ記録領域及びスタック領域がヒープ領域内に割当てられている。

40

【0099】

まず、従来の一括してコンパクション処理を行う方法の場合、主プログラムの最大停止時間は 9.5ms であった。一方、本発明のメモリ管理プログラム 93 による主プログラムの最大停止時間は 0.7ms であった。

50

【 0 1 0 0 】

なお、本発明は、上記各実施形態の構成に限られず、発明の趣旨を変更しない範囲で種々の変形が可能である。例えば、上記各実施形態では、本発明のメモリ管理装置を携帯電話に適用した例を示したが、これに限られず、例えば、コンピュータに適用したものであってもよい。また、上記各実施形態の携帯電話 1 では、フラッシュメモリ 17 にメモリ管理プログラム 93 が記録されている例を示したが、これに限らず、他の記憶媒体にメモリ管理プログラム 93 が記録されていてもよく、また、通信部 15 を用いて、インターネット上からメモリ管理プログラム 93 をダウンロードして、携帯電話に実行させるものであってもよい。また、CPU 11 内のレジスタをデータ領域とし、メモリ管理プログラム 93 が、レジスタに対して処理を行ってもよい。

10

【 0 1 0 1 】

また、上記各実施形態のメモリ管理プログラム 93 は、データ記録領域に記録されている、複製元オブジェクトの位置を参照するオブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に更新する処理中に、主プログラム 91 がデータ記録領域に複製元オブジェクトの位置を参照するオブジェクト位置ポインタを書込む処理を行うとき、又は書込み処理を行った後、オブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に変更する処理と、主プログラム 91 がヒープ領域に、複製元オブジェクトの位置を参照するオブジェクト位置ポインタを書込む処理を行うとき、又は書込み処理を行った後、オブジェクト位置ポインタの参照先を、複製先オブジェクトの位置に変更する処理と、をさらに行うものであってもよい。

20

【 図面の簡単な説明 】

【 0 1 0 2 】

【 図 1 】 本発明の第 1 実施形態に係る携帯電話の構成を示すブロック図。

【 図 2 】 上記携帯電話のフラッシュメモリの構成を示すブロック図。

【 図 3 】 上記携帯電話のデータ記録領域の構成を示すブロック図。

【 図 4 】 上記携帯電話のスタック領域の構成を説明する図。

【 図 5 】 上記携帯電話のヒープ領域に記録されるオブジェクトの構成を説明する図。

【 図 6 】 上記携帯電話におけるメモリ管理処理手順を示すフローチャート。

【 図 7 】 上記メモリ管理処理におけるコンパクション処理手順を示すフローチャート。

【 図 8 】 上記メモリ管理処理におけるオブジェクト位置ポインタ更新手順を示すフローチャート。

30

【 図 9 】 (a) (b) (c) は、上記携帯電話のヒープ領域におけるコンパクション処理を説明する図。

【 図 10 】 上記携帯電話のヒープ領域のコンパクション処理中にデータを書込む処理手順を示すフローチャート。

【 図 11 】 上記携帯電話のヒープ領域のコンパクション処理中にオブジェクト位置ポインタを書込む処理手順を示すフローチャート。

【 図 12 】 上記携帯電話のスタック領域におけるオブジェクト位置ポインタ更新処理手順を示すフローチャート。

【 図 13 】 (a) (b) は、上記携帯電話のスタック領域における障壁設定処理を説明する図。

40

【 図 14 】 上記携帯電話のオブジェクト位置ポインタの同一判断処理手順を示すフローチャート。

【 図 15 】 本発明の第 2 実施形態に係る携帯電話の構成を示すブロック図。

【 図 16 】 上記携帯電話の位置記録リストの一例を示す図。

【 図 17 】 上記携帯電話のヒープ領域のコンパクション処理手順を示すフローチャート。

【 図 18 】 本発明の第 3 の実施形態に係る携帯電話の構成を示すブロック図。

【 図 19 】 上記携帯電話のヒープ領域とデータ記録領域とスタック領域の状態を示す説明図。

【 図 20 】 上記携帯電話のデータ記録領域のコンパクション処理手順を示すフローチャー

50

ト。

【図 2 1】本発明の第 3 の実施形態に係る携帯電話のヒープ領域とデータ記録領域とスタック領域の状態を示す説明図。

【図 2 2】上記携帯電話のデータ記録領域のコンパクション処理手順を示すフローチャート。

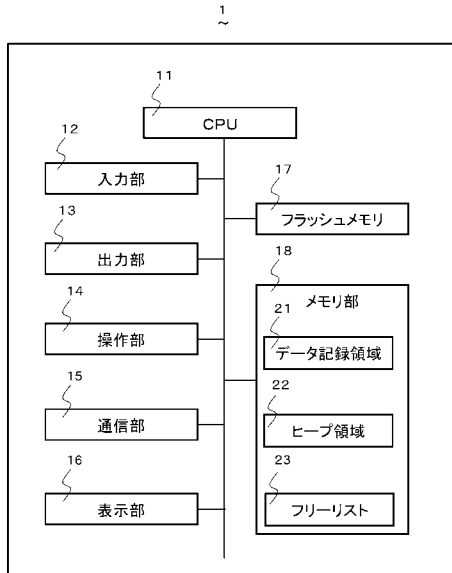
【図 2 3】上記携帯電話のスタック領域のコンパクション処理手順を示すフローチャート。

【符号の説明】

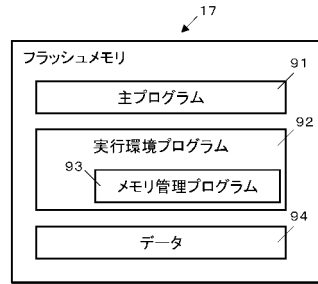
【 0 1 0 3 】

1	携帯電話 (メモリ管理装置)	10
1 1	C P U	
1 2	入力部	
1 3	出力部	
1 4	操作部	
1 5	通信部	
1 6	表示部	
1 7	フラッシュメモリ (記録媒体)	
1 8	メモリ部 (メモリ)	
2 1	データ記録領域	
2 2	ヒープ領域	20
2 4	位置記録リスト (オブジェクト位置記録領域)	
2 5	ヒープ領域	
2 6	データ記録領域	
2 7	スタック領域	
3 0	オブジェクト	
3 1 ~ 3 3	オブジェクト	
3 6	オブジェクト	
3 7	オブジェクト	
3 4	複製元オブジェクト	
3 5	複製元オブジェクト	30
3 4 a	複製先オブジェクト	
3 5 a	複製先オブジェクト	
4 0 ~ 4 4	オブジェクト位置ポインタ	
5 0 ~ 5 4	複製位置ポインタ	
6 0 a ~ 6 0 c	関数フレーム	
6 1 ~ 6 4	関数フレーム	
7 0	カレントフレーム	
8 0	障壁	
9 1	主プログラム	
9 3	メモリ管理プログラム	40
9 4	データ	
1 2 1 ~ 1 2 3	データ要素	
R	限定範囲	

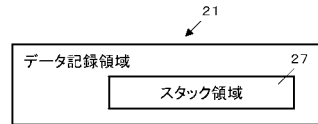
【図1】



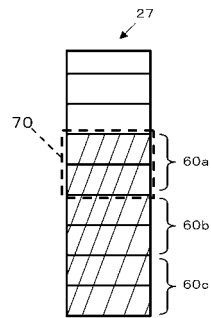
【図2】



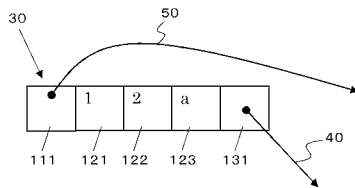
【図3】



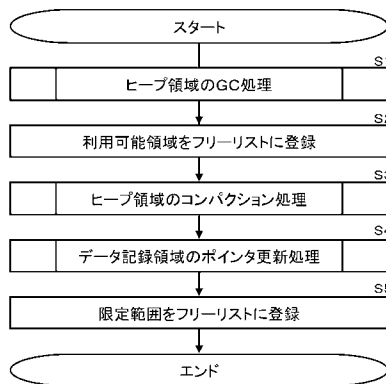
【図4】



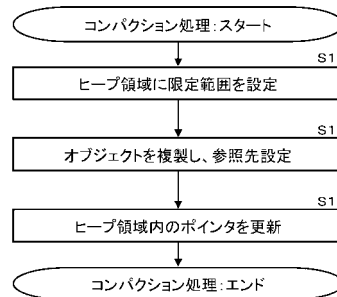
【図5】



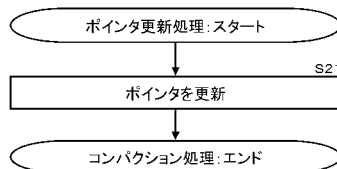
【図6】



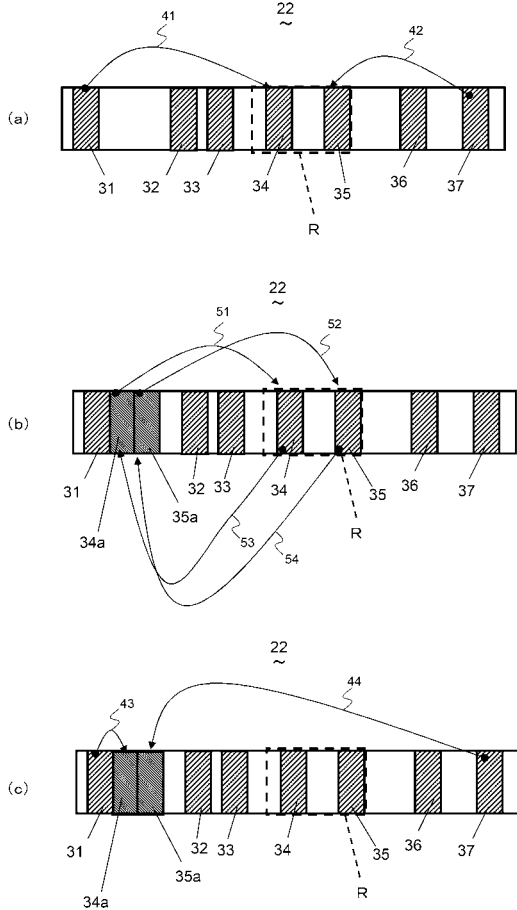
【図7】



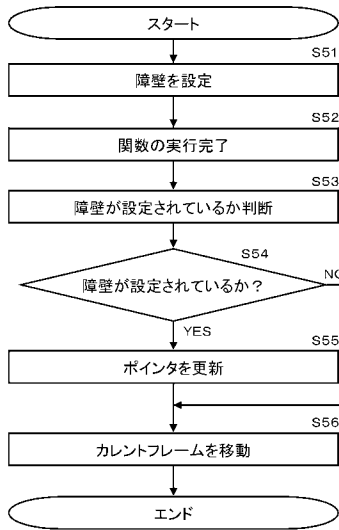
【図8】



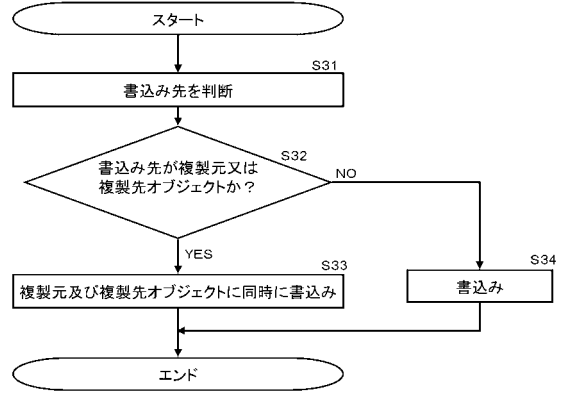
【図9】



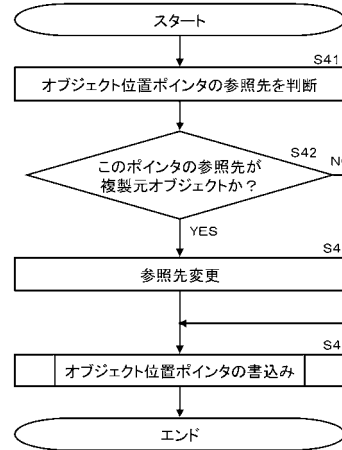
【図12】



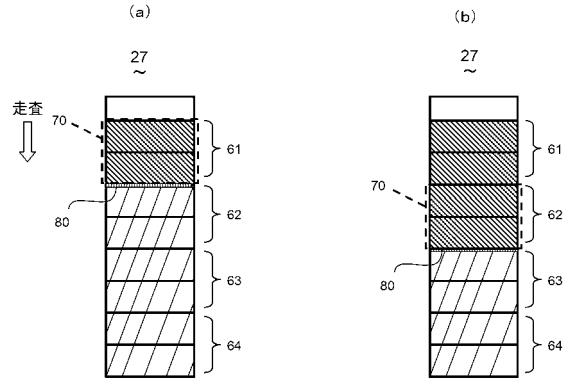
【図10】



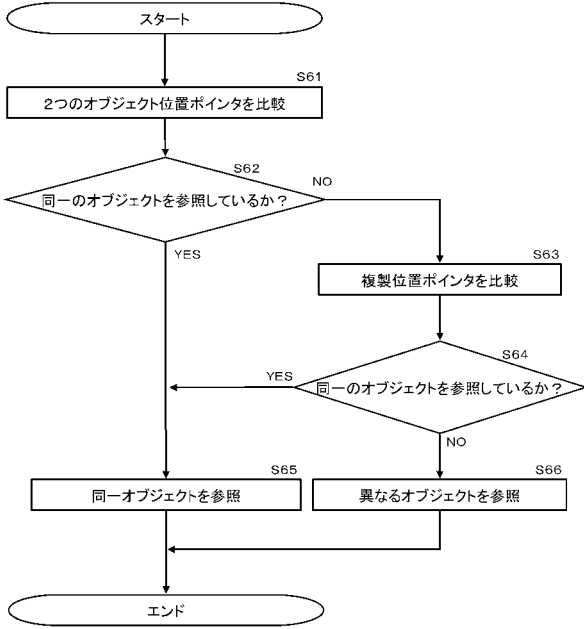
【図11】



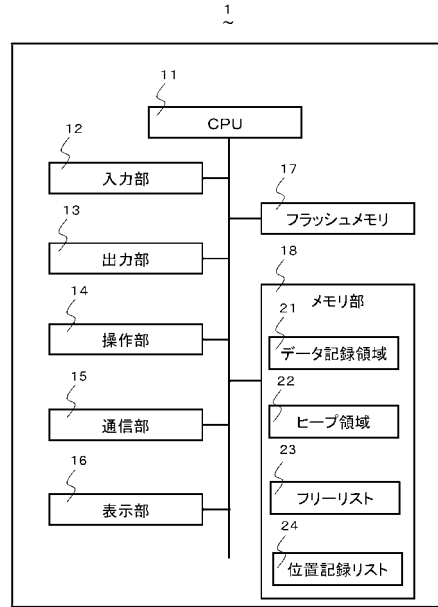
【図13】



【図14】



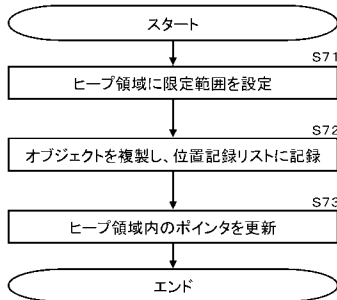
【図15】



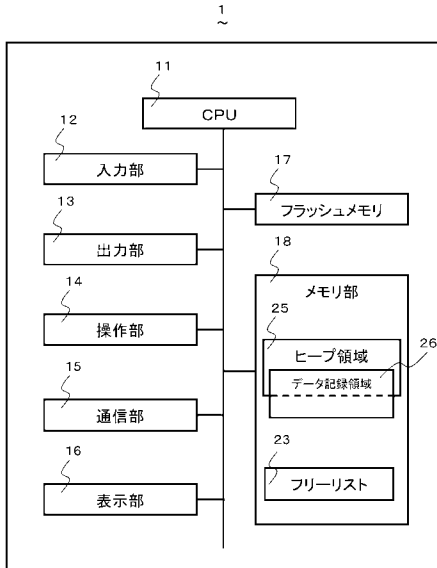
【図16】

複製元オブジェクト群の起点	複製先オブジェクト群の終点	複製先オブジェクト群の起点	複製先オブジェクト群の終点
16	31	128	143
64	95	144	175
96	111	192	207

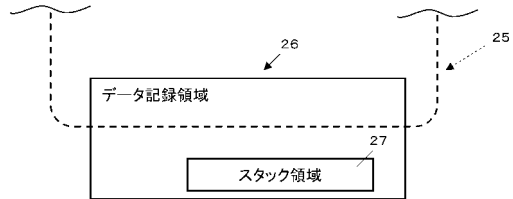
【図17】



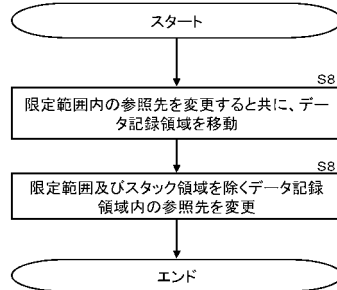
【図18】



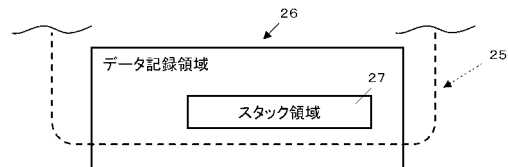
【図19】



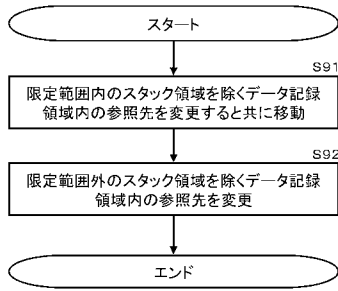
【図20】



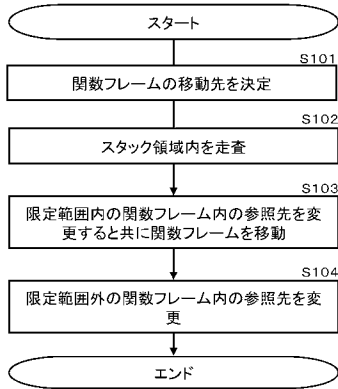
【図21】



【図 2 2】



【図 2 3】



フロントページの続き

審査官 多賀 実

- (56)参考文献 特表2002-506550(JP,A)
米国特許第06249793(US,B1)
特許第3530887(JP,B2)
寺島 元章 外2名,圧縮型並列ガーベッジコレクション,情報処理学会研究報告,日本,社団法人情報処理学会,1997年 8月22日,第97巻 第78号(97-PRO-14),p.115-120
Ori Ben-Yitzhak 外4名,An algorithm for parallel incremental compaction,ISMM '02 Proceedings of the 3rd international symposium on Memory management,米国,ACM,2002年,p.100-105

- (58)調査した分野(Int.Cl.,DB名)
G06F12/00
G06F12/02