

【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第3区分

【発行日】令和1年11月7日(2019.11.7)

【公開番号】特開2019-71120(P2019-71120A)

【公開日】令和1年5月9日(2019.5.9)

【年通号数】公開・登録公報2019-017

【出願番号】特願2019-4419(P2019-4419)

【国際特許分類】

G 06 F 9/48 (2006.01)

【F I】

G 06 F 9/48 370

【手続補正書】

【提出日】令和1年9月25日(2019.9.25)

【手続補正1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項1】

コンピューティングシステムによって実行されるタスクを制御するための方法であって

グラフに基づくプログラムの仕様に関するデータ構造をデータストレージシステムに記憶するステップであって、前記グラフに基づくプログラムの仕様が、タスクを表す複数のノードを含み、前記グラフに基づくプログラムの仕様が、前記ノードによって表される複数のタスクの間の少なくとも部分的な順序を規定するノードの間の有向辺を含む、前記記憶するステップと、

前記グラフに基づくプログラムの仕様によって規定されたプログラムを少なくとも1つのプロセッサを使用して実行するステップであって、

第1のタスクに対応する第1のサブルーチンを第1のプロセスによって実行することであって、前記第1のサブルーチンが、前記第1のタスクを実行するための第1のタスクセクション、及び前記グラフに基づくプログラムの仕様の中で前記第1のタスクのために1又は2以上の対応する後続のタスクの実行を制御するための論理を含む前記第1のサブルーチンに含まれる第1の制御セクションを含み、前記第1の制御セクションが、第2のタスクに対応する第2のサブルーチンを呼び出すための第1の関数呼び出しを含み、前記第2のタスクが、前記第2のタスクの実行を続けるために前記第1のプロセスに第2のプロセスをスポーニングさせる遅延閾値タイマーが超えられるまで前記第1のプロセスを使用して実行を開始するように構成され、前記第1のサブルーチンが、前記遅延閾値タイマーが超えられるときに前記第2のタスク及び第3のタスクが同時に実行されるように、前記第1のプロセスが前記第2のプロセスをスポーニングすると前記第1のプロセス上で実行される、前記第3のタスクに対応する第3のサブルーチンを呼び出すための第2の関数呼び出しをさらに含む、前記実行すること、

前記第1のタスクセクションが前記第1のタスクを実行するのを待っている保留状態及び前記第1のタスクセクションが前記第1のタスクを実行することを防止された抑制状態を含む1組の可能な状態から選択された前記第1のタスクの状態を示す状態情報を記憶することを含む、前記実行するステップと、

前記第2のタスクに対応する前記第2のサブルーチンを実行するステップであって、前記第2のサブルーチンが、前記第2のタスクを実行するための第2のタスクセクション、

及び記憶された状態情報によって示される前記第1のタスクの前記状態に少なくとも部分的に基づいて前記第2のタスクセクションの実行を制御する第2の制御セクションを含む、前記実行するステップとを含む、前記方法。

【請求項2】

上流のノードから下流のノードへの制御有向辺が、部分的な順序で前記上流のノードによって表されるタスクが前記下流のノードによって表される別のタスクに先立つことを示す、請求項1に記載の方法。

【請求項3】

1組の可能な状態が、第1のタスクセクションが第1のタスクを実行し終えた完了状態又は前記第1のタスクセクションが前記第1のタスクを実行する過程にあるアクティブ状態のうちの1又は2以上をさらに含む、請求項1に記載の方法。

【請求項4】

第2のサブルーチンの第2の制御セクションの論理が、第2のタスクセクションが呼び出されるか否かをフラグの値に基づいて判定する、請求項1に記載の方法。

【請求項5】

第2のサブルーチンの第2の制御セクションの論理が、第2のタスクを表すノードに制御有向辺によって接続されるノードによって表されるすべてのタスクが抑制状態にある場合、第2のタスクセクションが呼び出されないと判定する、請求項1に記載の方法。

【請求項6】

第2のサブルーチンの第2の制御セクションの論理が、第2のタスクを表すノードに制御有向辺によって接続されるノードによって表される少なくとも1つのタスクが完了状態にあり、前記第2のタスクを表すノードに制御有向辺によって接続されるノードによって表されるタスクのいずれも保留状態にない場合、第2のタスクセクションが呼び出されると判定する、請求項1に記載の方法。

【請求項7】

第2のサブルーチンの第2の制御セクションが、抑制状態を1又は2以上の他のタスクに伝播させるために、記憶される状態情報を判定する論理を含む、請求項1に記載の方法。

【請求項8】

1群の複数のタスクの各々の状態情報が、前記1群の複数のタスクの前記各々のそれぞれ1つによって記憶され、第2のサブルーチンに含まれる第2の制御セクションによってアクセスされる、請求項1に記載の方法。

【請求項9】

第2のタスクが、第2のタスクセクションが呼び出されるかどうかを前記第2のタスクに関連するアクティベーションカウンタに基づいて判定し、前記アクティベーションカウンタが、複数のタスクの状態の変化に応じて調整されるカウンタ値を含み、前記第2のタスクを実行することが、調整されたアクティベーションカウンタが所定の値に一致することに応じて前記第2のタスクを実行することを含む、請求項1に記載の方法。

【請求項10】

第2のタスクの実行を続けるように構成されたスポーニングされた第2のプロセスが、前記第2のタスクに関連するスタックフレームを取得し、前記スタックフレームに保有されたリターンポインタを取り除くように構成される、請求項1に記載の方法。

【請求項11】

タスクを制御するためのコンピューティングシステムであって、グラフに基づくプログラムの仕様に関するデータ構造を記憶するための手段であって、前記グラフに基づくプログラムの仕様が、タスクを表す複数のノードを含み、前記グラフに基づくプログラムの仕様が、前記ノードによって表される複数のタスクの間の少なくとも部分的な順序を規定するノードの間の有向辺を含む、前記記憶するための手段と、前記グラフに基づくプログラムの仕様によって規定されたプログラムを実行するための手段とを含み、実行することが、

第1のタスクに対応する第1のサブルーチンを第1のプロセスによって実行することであって、前記第1のサブルーチンが、前記第1のタスクを実行するための第1のタスクセクション、及び前記グラフに基づくプログラムの仕様の中で前記第1のタスクのために1又は2以上の対応する後続のタスクの実行を制御するための論理を含む前記第1のサブルーチンに含まれる第1の制御セクションを含み、前記第1の制御セクションが、第2のタスクに対応する第2のサブルーチンを呼び出すための第1の関数呼び出しを含み、前記第2のタスクが、前記第2のタスクの実行を続けるために前記第1のプロセスに第2のプロセスをスポーニングさせる遅延閾値タイマーが超えられるまで前記第1のプロセスを使用して実行を開始するように構成され、前記第1のサブルーチンが、前記遅延閾値タイマーが超えられるときに前記第2のタスク及び第3のタスクが同時に実行されるように、前記第1のプロセスが前記第2のプロセスをスポーニングすると前記第1のプロセス上で実行される、前記第3のタスクに対応する第3のサブルーチンを呼び出すための第2の関数呼び出しをさらに含む、前記実行すること、

前記第1のタスクセクションが前記第1のタスクを実行するのを待っている保留状態及び前記第1のタスクセクションが前記第1のタスクを実行することを防止された抑制状態を含む1組の可能な状態から選択された前記第1のタスクの状態を示す状態情報を記憶すること、並びに

前記第2のタスクに対応する前記第2のサブルーチンを実行することであって、前記第2のサブルーチンが、前記第2のタスクを実行するための第2のタスクセクション、及び記憶された状態情報によって示される前記第1のタスクの前記状態に少なくとも部分的に基づいて前記第2のタスクセクションの実行を制御する第2の制御セクションを含む、前記実行することを含む、前記コンピューティングシステム。

#### 【請求項12】

上流のノードから下流のノードへの制御有向辺が、部分的な順序で前記上流のノードによって表されるタスクが前記下流のノードによって表される別のタスクに先立つことを示す、請求項11に記載のコンピューティングシステム。

#### 【請求項13】

1組の可能な状態が、第1のタスクセクションが第1のタスクを実行し終えた完了状態又は前記第1のタスクセクションが前記第1のタスクを実行する過程にあるアクティブ状態のうちの1又は2以上をさらに含む、請求項11に記載のコンピューティングシステム。

#### 【請求項14】

第2のサブルーチンの第2の制御セクションの論理が、第2のタスクセクションが呼び出されるか否かをフラグの値に基づいて判定する、請求項11に記載のコンピューティングシステム。

#### 【請求項15】

第2のサブルーチンの第2の制御セクションの論理が、第2のタスクを表すノードに制御有向辺によって接続されるノードによって表されるすべてのタスクが抑制状態にある場合、第2のタスクセクションが呼び出されないと判定する、請求項11に記載のコンピューティングシステム。

#### 【請求項16】

第2のサブルーチンの第2の制御セクションの論理が、第2のタスクを表すノードに制御有向辺によって接続されるノードによって表される少なくとも1つのタスクが完了状態にあり、前記第2のタスクを表すノードに制御有向辺によって接続されるノードによって表されるタスクのいずれも保留状態にない場合、第2のタスクセクションが呼び出されると判定する、請求項11に記載のコンピューティングシステム。

#### 【請求項17】

第2のサブルーチンの第2の制御セクションが、抑制状態を1又は2以上のその他のタスクに伝播させるために、記憶される状態情報を判定する論理を含む、請求項11に記載のコンピューティングシステム。

**【請求項 1 8】**

1群の複数のタスクの各々の状態情報が、前記1群の複数のタスクの前記各々のそれぞれ1つによって記憶され、第2のサブルーチンに含まれる第2の制御セクションによってアクセスされる、請求項11に記載のコンピューティングシステム。

**【請求項 1 9】**

第2のタスクが、第2のタスクセクションが呼び出されるかどうかを前記第2のタスクに関連するアクティベーションカウンタに基づいて判定し、前記アクティベーションカウンタが、複数のタスクの状態の変化に応じて調整されるカウンタ値を含み、前記第2のタスクを実行することが、調整されたアクティベーションカウンタが所定の値に一致することに応じて前記第2のタスクを実行することを含む、請求項11に記載のコンピューティングシステム。

**【請求項 2 0】**

第2のタスクの実行を続けるように構成されたスポーニングされた第2のプロセスが、前記第2のタスクに関連するスタックフレームを取得し、前記スタックフレームに保有されたリターンポインタを取り除くように構成される、請求項11に記載のコンピューティングシステム。

**【請求項 2 1】**

タスクを制御するための、コンピュータ可読ストレージ媒体に記憶されたコンピュータプログラムであって、コンピューティングシステムに請求項1～10のいずれかに記載の方法を実行させるための命令を含む、前記コンピュータプログラム。

**【請求項 2 2】**

タスクを制御するためのコンピューティングシステムであって、グラフに基づくプログラムの仕様に関するデータ構造を記憶するデータストレージシステムと、

請求項1～10のいずれかに記載の方法を実行するように構成された少なくとも1つのプロセッサとを含む、前記コンピューティングシステム。