

Intel[®] Visual Fortran

リファレンスガイド

菅原清文／堀越将司／黒沢一平●共著



本書で取り上げられているシステム名／製品名は、一般に開発各社の登録商標／商品名です。本書では、TMおよび®マークは明記していません。本書に掲載されている団体／商品に対して、その商標権を侵害する意図は一切ありません。本書で紹介しているURLや各サイトの内容は変更される場合があります。

はじめに

最古の高級言語である FORTRAN ですが、いまだに多くのユーザーに支持され、インテル Visual Fortran のユーザーも年々増えつつあります。インテルでは Windows 以外に Linux と Mac OS X 向けの Fortran コンパイラを提供していますが、Windows 版は Visual Studio 統合環境での利用をサポートし、旧製品の名称を引き継ぐ意味でも「インテル Visual Fortran」と他の OS 向けの製品と呼び名を異にします。しかし言語自体の機能は他の OS で提供される Fortran コンパイラと同じで、9.0 以降のバージョンでは Fortran2003 の言語の一部仕様をサポートし、拡張が続いています。

インテル Fortran と C/C++ コンパイラの最適化に関するオプション群はほぼ共通ですので、インテル C/C++ コンパイラを利用したことのあるユーザーには取り組み易いでしょう。またその逆もしかりで、今後インテル C/C++ を使うときに本書で紹介される最適化オプションや概要は役に立つと思います。第9章では Fortran プログラムのデバックや最適化に有効なツール群も紹介していますので是非お試しください。

ここ数年、マルチコアプロセッサに対応したプログラミングが注目されていますが、C/C++ の世界では様々なスレッドを抽象化したライブラリがサポートされはじめています。その点では Fortran は遅れているといわざるをえません。しかし、Fortran は OpenMP をサポートした最初の言語でもあり、本書では OpenMP の標準仕様とともにインテルによる拡張仕様を紹介しています。

本書が皆様に役立つことを期待しています。

著者一同

■ 本書の対象読者

本書はFORTRAN コンパイラの利用経験と知識があり、コマンドラインモードでは使えるが、Visual Studio に統合した利用法がよくわからないというユーザーや、Linux や Mac OS X から Windows への移行を考えているユーザー向けに、インテル Visual Fortran コンパイラを Visual Studio 環境で利用する方法を解説するものです。FORTRAN の言語仕様については他の文献を参考にしてください。

■ 本書で説明するプログラムの実行環境

- Windows XP SP2
- Windows Vista
- Visual Studio 2005
- Visual Studio 2008
- インテル Visual Fortran バージョン 10.0
- インテル Visual Fortran バージョン 10.1

■ 注意

- 本書を利用、参照、実装した結果の影響について、著者、編集者、出版社はいっさい責任を負いかねます。あらかじめご了承ください。
- 本書に記載される情報は2008年6月執筆時のものです。インテル Visual Fortran コンパイラやマイクロソフト Visual Studio のバージョンやリビジョンは変更される可能性があることをご了承ください。
- 本書に含まれる説明や例題は、本書執筆時の著者の環境を元に記述されています。特定の環境でのプログラムのコンパイルおよび実行を保証するものではありません。
- 本書に含まれるすべての情報は、本書執筆時のものです。定義や権利所有者、ソフトウェアの仕様は、将来変更される可能性があります。
- ここに記載されている会社名、商品名や製品名などは、一般に各権利所有者の登録済み社名、商標または登録商標です。
- Intel、インテル、Intel ロゴ、「Intel さあ、その先へ。」ロゴ、Pentium、Itanium、Xeon、Core、は Intel Corporation の商標または登録商標です。

■ 本書の内容

第1章 インテル Visual Fortran 概要

インテル Visual Fortran コンパイラの歴史と背景、および各歴代のバージョンの特徴を説明します。また、パッケージに含まれるインテル Visual Fortran を構成するコンポーネントやサポートツールについて解説します。

第2章 インストールとコマンドラインでの利用

インテル Visual Fortran コンパイラのインストールパッケージとライセンス及びサポートアカウントの登録と利用方法を説明します。また、Windows 環境でのインストールとコマンドラインからの起動方法を解説します。

第3章 Visual Studio でのインテル Visual Fortran の利用

インテル Visual Fortran はマイクロソフト Visual Studio 環境へ統合して利用することができます。統合環境での Fortran プロジェクトの作成方法やインテル Visual Fortran に含まれるテンプレートを使用した各種 Windows プログラム環境について説明します。

第4章 Visual Studio でのオプションの設定とビルド環境

Visual Studio 環境でインテル Visual Fortran を利用する際のオプション設定と、関連する Visual Studio の操作方法について説明します。また、Makefile を使用する場合のビルド方法についてもこの章で解説します。

第5章 プログラムの実行とデバッグ

インテル Visual Fortran コンパイラで作成したプログラムを実行する際のランタイム環境変数とその役割について説明します。また、OpenMP ライブラリを使用する際の標準 OpenMP の環境変数とインテル拡張 OpenMP 環境変数についてもここで説明します。さらに Visual Studio 環境での Fortran プログラムのデバッグ方法やトレースバック機能について解説します。

第6章 Windows API を利用する

QuickWin モードや Fortran Windows アプリケーションから Windows API を利用する方法と各種 API について解説します。また、Fortran を利用したマルチスレッドアプリケーションの作成方法についてもこの章で説明します。

第7章 C/C++ と Fortran の混在

Fortran と C を混在して利用する場合の、呼び出し規則、データ交換および互換性の保持について説明します。Fortran と C の双方からの呼び出しと開発時の注意点をまとめます。

第8章 Fortran プログラムの最適化と並列化

インテル Visual Fortran が提供する特徴的な最適化機能、ベクトル化、ループ変換、自動並列化、OpenMP、プロシージャ間の最適化そしてプロファイルに基づく最適化の特徴と活用法を解説します。また、コード開発時のパフォーマンスの目標設定とキャッシュの最適化についても触れます。

第9章 他のインテルツールとの共存

Fortran プログラムの最適化、デバッグ、スレッドチューニングに役立つインテルソフトウェア開発ツールについての説明を行います。対象となるツールは、VTUNE アナライザ、スレッドチェッカ、スレッドプロファイラ、そしてマスカーネルライブラリです。

付録A インテル Visual Fortran コンパイラの全オプションクイックリファレンス

付録B 用語の説明

付録C 関数一覧

目次

はじめに	iii
------------	-----

第1章 インテル Visual Fortran 概要 ... 1

1-1 インテル Fortran とインテル Visual Fortran の履歴	2
1-2 インテル Visual Fortran を構成するコンポーネントと サポート環境	6
1-3 インテル Visual Fortran に含まれるツール	9
1-4 開発したアプリケーションと再配布可能ライブラリ	14

第2章 インストールとコマンドラインでの利用 ... 17

2-1 インストールパッケージとライセンスファイルの入手	18
2-1-1 評価版を入手する	18
2-1-2 製品版ユーザーが最新のバージョンを入手する場合	21
2-1-3 サポートアカウントの登録と利用	24
2-2 インストールと環境	25
2-2-1 インテル Visual Fortran コンパイラをインストールする前に	25
2-2-2 インストールを開始する	26
2-3 コマンドラインから起動する	30
2-3-1 コマンドプロンプトを開く	30
2-3-2 ヘルプを表示する	32
2-3-3 コンパイラを起動する	32

第3章 Visual Studio でのインテル Visual Fortran の利用 ... 35

3-1 インテル Visual Fortran の統合を確認する	36
3-2 サンプルコードのコンパイル	37

3-3	プロジェクトの作成	42
3-3-1	コンソールモード	44
3-3-2	QuickWin モード	45
3-3-3	Windowing アプリケーションモード	47
3-3-4	ライブラリモード	50
3-3-5	COM サーバモード	53
3-3-6	Visual Fortran モジュールウィザード	57
3-4	古いプロジェクトを変換する	58
3-4-1	Compaq Visual Fortran プロジェクトワークスペースの変換方法	60
3-4-2	旧 Visual Fortran を移植する場合の注意	61

第4章 Visual Studio でのオプションの設定とビルド環境 ... 63

4-1	Visual Fortran のオプション設定	65
4-1-1	General (全般) オプション	68
4-1-2	Optimization (最適化) オプション	70
4-1-3	Debugging (デバッグ) オプション	74
4-1-4	Preprocessor (プリプロセッサ) オプション	75
4-1-5	Code Generation (コード生成) オプション	77
4-1-6	Language (言語) オプション	77
4-1-7	Compatibility (互換性) オプション	79
4-1-8	Diagnostics	82
4-1-9	Data (データ) オプション	87
4-1-10	Floating Point (浮動小数点アンダーフロー) オプション	91
4-1-11	External Procedures (外部プロシージャ) オプション	93
4-1-12	Output Files (出力ファイル) オプション	94
4-1-13	Run-Time (ランタイム) オプション	95
4-1-14	Libraries (ライブラリ) オプション	97
4-1-15	Command Line (コマンドライン) オプション	98
4-2	関連する Visual Studio の設定	99
4-2-1	ソリューションの構成選択	99
4-2-2	プラットフォームタイプの選択	101
4-2-3	ツールオプションの設定	104

4-3	コンパイル時の環境変数と関連情報	106
4-3-1	コンパイル時の環境変数の設定	106
4-3-2	デフォルトツール	108
4-3-3	Visual Fortran と Visual Studio のファイル関連付けについて	110
4-4	nmake コマンドを利用する	111

第5章 プログラムの実行とデバッグ ... 113

5-1	ランタイム環境変数	114
5-2	OpenMP の環境変数	119
5-2-1	一般的な OpenMP 環境変数	119
5-2-2	インテル固有の OpenMP 環境変数	120
5-2-3	スレッドアフィニティインターフェース	121
5-3	Visual Studio でのデバッグ	132
5-3-1	デバッグモードでプログラムを実行する	132
5-4	トレースバック機能	140
5-5	ランタイムメッセージの表示と書式	141

第6章 Windows API の利用 ... 145

6-1	QuickWin モードを利用したアプリケーションの作成	146
6-1-1	メニューのプログラムからの制御	146
6-1-2	ステータスバーおよび状態メッセージの変更	152
6-1-3	メッセージボックスの表示	152
6-1-4	About ボックスの定義	152
6-1-5	カスタムアイコンの使用	153
6-1-6	マウスの使用	154

6-2	マルチスレッドアプリケーションの作成	158
6-2-1	マルチスレッドプログラムのモジュール.....	160
6-2-2	スレッドの起動と停止.....	160
6-2-3	スレッドルーチンの書式.....	163
6-2-4	リソースの共有.....	164
6-2-5	スレッドローカルストレージ.....	167
6-2-6	スレッドの同期.....	168
6-2-7	マルチスレッドプログラムでのエラー処理.....	169
6-2-8	複数のプロセスの使用.....	169
6-2-9	マルチスレッドルーチンの表.....	170
6-3	Fortran Windows アプリケーションの作成	172
6-3-1	WIN32 ルーチンの呼び出し.....	173
6-3-2	Fortran Windows アプリケーションのコーディング要件.....	183
6-3-3	SDI / MDI の Fortran Windows アプリケーションでのメニューと ダイアログの使用.....	187
6-4	モジュールウィザードによって生成されたルーチンの呼び出し	191

第7章 C/C++ と Fortran の混在 ... 197

7-1	Fortran/C 呼び出し規則	200
7-1-1	命名規則.....	200
7-1-2	引数の受け渡し.....	200
7-1-3	プログラムのビルド (コンパイルとリンク).....	201
7-2	C との互換性保持	204
7-2-1	組み込み型の互換性保持.....	204
7-2-2	C ポインタとの互換性保持.....	205
7-2-3	派生型の互換性保持.....	206
7-2-4	変数の互換性保持.....	206
7-2-5	プロシージャの互換性保持.....	206
7-2-6	グローバルデータの互換性保持.....	208
7-2-7	Fortran からの C の呼び出し例.....	208
7-2-8	C からの Fortran の呼び出し例.....	209
7-3	言語が混在したプログラミングにおけるデータ交換と参照	211
7-4	言語が混在したプログラミングにおけるモジュールの使用	212

7-5	言語が混在したプログラミングにおける共通外部データの使用	215
7-5-1	言語が混在したプログラミングにおけるグローバル変数の使用	215
7-5-2	Fortran の共通ブロックおよびCの構造体の使用	216
7-5-3	共通ブロックおよびCの構造体への直接アクセス	217
7-5-4	共通ブロックのアドレスの受け渡し	218
7-6	言語が混在したプログラミングにおけるデータ型の処理の概要	219
7-6-1	数値、複素数、および論理データ型の処理	220
7-7	配列ポインタと割付け配列の処理	222
7-7-1	整数ポインタの処理	223
7-7-2	配列と Fortran 配列記述子の処理	226
7-7-3	文字列の処理	230
7-7-4	ユーザー定義型の処理	234

第8章 Fortranプログラムの最適化 … 237

8-1	基本的な最適化の概要	238
8-1-1	一般的な最適化	239
8-1-2	特定のIA-32アーキテクチャベースのプロセッサを対象にする	241
8-1-3	自動的にIA-32アーキテクチャベースのプロセッサを判断する	244
8-1-4	自動でItaniumプロセッサを対象にする	246
8-1-5	最適化の制限	247
8-1-6	コンパイラのサンプルを使用した最適化の例	247
8-2	プロシージャ間の最適化 (IPO) の使用	250
8-2-1	プロシージャ間の最適化 (IPO) の概要	250
8-2-2	プロシージャ間の最適化 (IPO) のクイックリファレンス	253
8-2-3	IPOの使用	254
8-2-4	IPOに関連するパフォーマンス問題	256
8-2-5	IPOサンプル	257
8-2-6	インライン関数の展開	258
8-3	プロファイルに基づく最適化 (PGO) の使用	259
8-3-1	プロファイルに基づく最適化の概要	260
8-3-2	プロファイルに基づく最適化 (PGO) の利用法	261
8-3-3	アプリケーションのプロファイル	264
8-3-4	PGOサンプル	265

8-4	並列化の使用	267
8-4-1	OpenMP* サポートの概要.....	268
8-4-2	OpenMP の使用.....	269
8-4-3	OpenMP の環境変数.....	272
8-4-4	自動並列化の概要.....	272
8-4-5	自動並列化オプションのクイックリファレンス.....	274
8-4-6	自動並列化：有効、オプション、宣言子、および環境変数.....	274
8-4-7	自動ベクトル化の概要.....	276
8-4-8	自動ベクトル化オプションのクイックリファレンス.....	276
8-4-9	ベクトル化のプログラミングにおけるガイドライン.....	277
8-4-10	ベクトル化の例.....	278
8-4-11	ベクトル化のサンプル.....	281
8-4-12	宣言子によるベクトル化のヒント.....	282
8-5	浮動小数点の最適化	287
8-6	最適化の簡単な手順	288

第9章 他のインテルツールとの共存 ... 291

9-1	パフォーマンスサイクル	292
9-2	マルチスレッドソフトウェア開発サイクル	294
9-3	インテルVTune パフォーマンスアナライザ	296
9-3-1	サンプリング.....	296
9-3-2	コールグラフ.....	299
9-3-3	パフォーマンスモニタ.....	300
9-3-4	VTune を利用する際の注意点.....	300
9-4	インテルスレッドチェッカ	301
9-5	インテルスレッドプロファイラ	303
9-6	パフォーマンスライブラリ	306

付録A コンパイラオプション一覧 … 310

A-1	一般的な最適化に関するオプション	310
A-2	高度な最適化	312
A-3	コード生成	313
A-4	言語仕様	315
A-5	互換性	318
A-6	コンパイラの診断メッセージ	319
A-7	インライン制御	320
A-8	プロシージャ間の最適化 (IPO)	321
A-9	プロファイルに基づく最適化 (PGO)	322
A-10	最適化レポート	323
A-11	OpenMPと並列化	324
A-12	浮動小数点精度	325
A-13	プリプロセッサ	327
A-14	出力とデバッグ	328
A-15	データ属性	329
A-16	その他	331
A-17	コンポーネント制御	332
A-18	リンカとリンク制御	332

付録B 用語集 … 334

B-1	レコード型	334
B-2	固定長レコード	335
B-3	可変長レコード	336
B-4	セグメントレコード	337
B-5	ストリームファイルデータ	338
B-6	Stream_CRレコード、Stream_LFレコード、 Stream_CRLFレコード	338
B-7	レコード型を選択する際のガイドライン	339
B-8	書式なしファイルでの区切り文字の指定	339
B-9	データ書式の指定方法	340

付録 C 関数一覧 ... 342

C-1	QuickWin API 関数	342
	ABOUTBOXQQ	342
	APPENDMENUQQ	342
	CLICKMENUQQ	344
	DELETEMENUQQ	345
	INCHARQQ	346
	INITIALSETTINGS	346
	INSERTMENUQQ	347
	IOR	349
	LOC	350
	MESSAGEBOXQQ	350
	MODIFYMENUFLAGSQQ	352
	MODIFYMENUSTRINGQQ	352
	MODIFYMENUROUTINEQQ	353
	REGISTERMOUSEEVENT	354
	SETMESSAGEQQ	356
	SETWINDOWMENUQQ	357
	SETWSIZEQQ	358
	SETWINDOWCONFIG	359
	UNREGISTERMOUSEEVENT	362
	WAITONMOUSEEVENT	363
C-2	グラフィックス関数	364
	SETMOUSECURSOR	364
	参考文献	369
	索引	370