

インテル® Fortran コンパイラー 19.1 for Linux* リリースノート (インテル® Parallel Studio XE 2020)

2019年12月4日

このドキュメントでは、新機能、変更された機能、注意事項、および製品ドキュメントに記述されていない既知の問題について説明します。

パッケージに含まれるライセンスと本リリースノートの「著作権と商標について」をお読みください。本リリースのインテル® Fortran コンパイラー 19.1 についての詳細は、次のリンクを参照してください。

- [動作環境](#)
- [インストール方法](#)
- [使用方法](#)
- [ドキュメント](#)
- [インテルが提供するデバッグ・ソリューション](#)
- [日本語のサポート](#)
- [サンプル](#)
- [再配布可能なライブラリー](#)
- [テクニカルサポート](#)
- [互換性](#)
- [新機能と変更された機能](#)
- [新規および変更されたコンパイラー・オプション](#)
- [終了予定のサポート](#)
- [終了したサポート](#)
- [既知の問題](#)
- [Fortran 2008 および Fortran 2018 機能の概要](#)
- [著作権と商標について](#)

変更履歴

このセクションでは製品アップデートにおける重要な変更内容を説明します。

インテル® Fortran コンパイラー 19.0 以降 (インテル® Fortran コンパイラー 19.1.0 での変更)

- [Fortran 2018 の新機能](#)
- [OpenMP* 5.0 の新機能](#)
- [新しいコンパイラー・オプション](#)
 - [-assume \[no\]old_inquire_recl](#)
 - [-assume \[no\]old_ldout_zero](#)
 - [-check \[no\]udio_iostat](#)
 - [-warn \[no\]externals](#)
- [コンパイラーの動作の変更](#)
- 報告された問題を修正

動作環境

アーキテクチャー名についての説明は、「[インテル® アーキテクチャー・プラットフォームの用語](#)」(英語)を参照してください。

- インテル® ストリーミング SIMD 拡張命令 2 (インテル® SSE2) 対応のインテル® 64 アーキテクチャー・ベースのプロセッサを搭載したコンピューター (第 2 世代以降のインテル® Core™ i3/i5/i7 プロセッサ、インテル® Xeon® プロセッサ E3/E5 ファミリー、または互換性のあるインテル以外のプロセッサ)
 - 64 ビット・バージョンの OS で 32 ビット・アプリケーションを開発する場合は、Linux* ディストリビューションからオプションのライブラリー・コンポーネント (ia32-libs、lib32gcc1、lib32stdc++6、libc6-dev-i386、gcc-multilib、g++-multilib) をインストールする必要があります。
- 機能を最大限に活用できるよう、マルチコアまたはマルチプロセッサ・システムの使用を推奨します。
- RAM 2GB (4GB 推奨)
- 14GB のディスク空き容量 (すべての機能をインストールする場合)
- IA-32 対応アプリケーションまたはインテル® 64 対応アプリケーションを開発する場合は、次の Linux* ディストリビューションのいずれか (本リストは、インテル社により動作確認が行われたディストリビューションのリストです。その他のディストリビューションでも動作する可能性はありますが、推奨しません。ご質問は、[テクニカルサポート](#)までお問い合わせください。)
 - Red Hat* Enterprise Linux* 7、8
 - CentOS* 8
 - Ubuntu* 18.04 LTS、19.04
 - Fedora* 30
 - SUSE* Linux* Enterprise Server 15

- Debian* 10.0
- Linux* 開発ツール・コンポーネント (gcc, g++ および関連ツールを含む)。(本リストは、インテル社により動作確認が行われたコンポーネント・バージョンのリストです。その他のバージョンでも動作する可能性はありますが、推奨しません。ご質問は、[テクニカルサポート](#)までお問い合わせください。)
 - gcc 4.8.5 - 9.0.1
 - binutils 2.27 - 2.31.1
- `-traceback` オプションを使用するには、`libunwind.so` が必要です。一部の Linux* ディストリビューションでは、別途入手して、インストールする必要があります。

注

- インテル® コンパイラーは、さまざまな Linux* ディストリビューションと gcc バージョンで動作確認されています。使用する glibc のバージョンは、gcc のバージョンと同じでなければなりません。最良の結果を得るため、上記のディストリビューションで提供されている gcc バージョンのみを使用してください。
- 非常に大きなソースファイル (数千行以上) を `-O3`、`-ipo` および `-qopenmp` などの高度な最適化オプションを使用してコンパイルする場合は、大量の RAM が必要になります。
- 一部の最適化オプションには、アプリケーションを実行するプロセッサの種類に関する制限があります。詳細は、オプションの説明を参照してください。

インテル® Fortran コンパイラーのインストール方法

インストール手順は、ドキュメントの一部として、インテル® ソフトウェア開発製品に同梱されています。最新のインテル® Parallel Studio XE バージョンのインストール・ガイドは、オンラインでも利用できます。

「[インテル® Parallel Studio XE 入門](#)」(英語) でインストール・ガイドを確認してください。インストール・ガイドへのリンクは、必要なインストール・パッケージをダウンロードしている間、[インテル® ソフトウェア開発製品レジストレーション・センター](#)からも利用できます。詳細は、「[インストールに関するよくある質問](#)」(英語) を参照してください。

インテル® Fortran コンパイラーの使用方法

インテル® Fortran コンパイラーの使用方法は、「インテル® Parallel Studio XE 2020: インテル® Fortran++ コンパイラー 19.1 for Linux* 入門」(<install-dir>/documentation_2020/ja/compiler_f/ps2020/get_started_lf.htm) を参照してください。

ドキュメント

製品ドキュメントへのリンクは、<install-dir>/documentation_2020/ja/compiler_f/ps2020/get_started_lf.htm にあります。

インストール・イメージからオフライン・コア・ドキュメントを削除

インテル® Parallel Studio XE のインストール・イメージからオフライン・コア・ドキュメントが削除されました。インテル® Parallel Studio XE のコンポーネントのコア・ドキュメントは、[インテル® ソフトウェア・ドキュメント・ライブラリー](#) (英語) からオンラインで参照できます。また、[インテル® ソフトウェア開発製品レジストレーション・センター](#) から、以前のバージョンのオフライン・ドキュメントをダウンロードすることもできます:

[Product List > Intel® Parallel Studio XE Documentation](#).

デベロッパー・ガイドおよびリファレンス、新機能とリリースノート、インストール・ガイド

すべてのツール・コンポーネントのデベロッパー・ガイドおよびリファレンス、新機能とリリースノート、インストール・ガイドは、[Intel® Parallel Studio XE Support > Documentation](#) (英語) から入手できます。

日本語のサポート

日本語のサポートは、すべてのアップデートではなく、一部のアップデートで提供されます。

日本語オペレーティング・システムで英語のサポートを使用する (または英語オペレーティング・システムで日本語のサポートを使用する) 方法については、[こちらの記事](#) (英語) を参照してください。

インテルが提供するデバッグ・ソリューション

インテルが提供するデバッグ・ソリューションは GNU* GDB ベースです。詳細は、「[インテル® Parallel Studio 2020 Composer Edition for Fortran - デバッグ・ソリューション・リリースノート](#)」(英語) を参照してください。

サンプル

製品のサンプルは、「[インテル® ソフトウェア製品のサンプルとチュートリアル](#)」(英語) からダウンロードできます。

再配布可能なライブラリー

詳細は、「[インテル® Parallel Studio XE の再配布ライブラリー](#)」(英語) を参照してください。

テクニカルサポート

[インテル® ソフトウェア開発製品レジストレーション・センター](#)でライセンスを登録してください。登録を行うことで、サポートサービス期間中 (通常は 1 年間)、製品アップデートと新しいバージョンの入手を含む無償テクニカルサポートが提供されます。

テクニカルサポート、製品のアップデート、ユーザーフォーラム、FAQ、ヒント、およびその他のサポート情報は、<http://www.intel.com/software/products/support/> (英語) を参照してください。

注: 販売代理店が製品のテクニカルサポートを提供している場合、インテルではなく販売代理店にお問い合わせください。

互換性

一般に、インテル® Fortran コンパイラー for Linux* の以前のバージョン (8.0 以降) でコンパイルされたオブジェクト・コードおよびモジュールは、バージョン 19.1 でもそのまま使用できます。ただし、次の例外がありません。

- バージョン 12.0 よりも前のコンパイラーでビルドされた CLASS キーワードを使用して多相変数を宣言しているソースは再コンパイルする必要があります。
- マルチファイルのプロシージャラー間の最適化 (-ipo) オプションを使用してビルドされたオブジェクトは、最新のバージョンで再コンパイルする必要があります。
- バージョン 12.0 よりも前のコンパイラーでビルドされた REAL (16)、REAL*16、COMPLEX (16)、COMPLEX*32 データ型を使用しているオブジェクトは再コンパイルする必要があります。
- バージョン 10.0 よりも前のコンパイラーでインテル® 64 アーキテクチャー用にビルドされたモジュール変数を含むオブジェクトは再コンパイルする必要があります。Fortran 以外のソースからこれらの変数を参照する場合、不正な先頭の下線を削除するように外部名を変更する必要があります。
- バージョン 11.0 よりも前のコンパイラーでコンパイルされた、派生型宣言の外部で ATTRIBUTES ALIGN ディレクティブを指定したモジュールは再コンパイルする必要があります。この問題を検出すると、コンパイラーはメッセージを表示します。
- 派生型宣言の内部で ATTRIBUTES ALIGN ディレクティブを指定したモジュールは 13.0.1 以前のコンパイラーでは使用できません。
- Fortran 2008 のサブモジュール機能を実装するため、バイナリー .mod ファイルの内部フォーマットが大幅に変更されました。このため、バージョン 16.0 以降の Fortran コンパイラーで作成されたモジュールファイルは、バージョン 15.0 以前の Fortran コンパイラーで使用することはできません。
- インテル® Xeon Phi™ x100 製品ファミリー向けにコンパイル/ビルドされたオブジェクト/ライブラリーは、インテル® Xeon Phi™ x200 製品ファミリー向けにコンパイル/ビルドされたオブジェクト/ライブラリーと互換性がありません。

- インテル® Fortran コンパイラー 18.0 以前のバージョンの `regcall` 呼び出し規約でビルドされたオブジェクトは、19.0 以降で再コンパイルする必要があります。インテルの `libirng` ライブラリーは `regcall` を使用しています。インテル® Fortran コンパイラー 19.0 以降に同梱の `libirng` ライブラリーは、古いバージョンと互換性がありません。`regcall` は、次の属性で暗黙的に使用されています。
`!DIR$ ATTRIBUTES [att,] VECTOR [:clause] [, att]... :: routine-name`

REAL(16) および COMPLEX(16) データ型のスタック・アライメントの変更

バージョン 12.0 よりも古いコンパイラーでは、`REAL(16)` または `COMPLEX(16)` (`REAL*16` または `COMPLEX*32`) 項目が値で渡される場合、スタックアドレスは 4 バイトでアラインされます。パフォーマンスを向上するため、バージョン 12 以降のコンパイラーは、これらの項目を 16 バイトでアラインし、引数が 16 バイト境界でアラインされていると仮定します。これは、`gcc` と互換性があります。

この変更は、主にコンパイラーにより生成される `REAL(16)` 値の計算を行うライブラリー・ルーチン (組込み関数を含む) の呼び出しに影響します。以前のバージョンでコンパイルしたコードをバージョン 12 のライブラリーとリンクする場合、またはアプリケーションをインテルのランタイム・ライブラリーの共有バージョンにリンクする場合、正しくない結果が返される可能性があります。

バージョン 12.0 よりも古いコンパイラーでコンパイルされている場合、この問題を回避するには、`REAL(16)` および `COMPLEX(16)` データ型を使用しているすべての Fortran ソースを再コンパイルしてください。

新規および変更されたコンパイラー機能

Fortran 2018 の新機能

- `IMPLICIT` 文の拡張によりすべての外部プロシーチャーを `EXTERNAL` で宣言する指定を許可
- `GENERIC` 文の拡張により汎用インターフェイスの宣言での使用を許可
- 変数の局所性を `DO CONCURRENT` 文で指定可能に
- `E`、`D`、`EN`、`ES`、および `G` 編集記述子の拡張によりゼロ幅のフィールドを許可、`F` 編集記述子に類似
- データ編集記述子の指数の幅 `e` をゼロに、ゼロ幅のフィールドに類似
- `RN` 編集記述子を Fortran 2018 および ISO/IEC/IEEE 60559:2011 で指定された最近接丸めに
- `EX` 編集記述子で浮動小数点値の 16 進形式の出力を許可。16 進形式の浮動小数点値を入力で許可。
- `SIZE=` を非前進 I/O で指定
- 保留中の非同期操作で `INQUIRE` 文の `SIZE=` および `POS=` の値を標準化
- `INQUIRE` 文の `RECL=` 指定子に割り当てられる値を標準化
- `CMPLX` 組込み関数の新しい形式、最初の引数が `COMPLEX` 型の場合、`KIND=` キーワードは不要
- `SIGN` 関数の引数で異なる種類を指定可能に
- 組込みプロシーチャーの `INTEGER` および `LOGICAL` 引数でデフォルトの種類が不要に

- 名前付き定数 `STAT_FAILED_IMAGE` および `STAT_UNLOCKED_FAILED_IMAGE` を組み込み `ISO_FORTRAN_ENV` モジュールで定義
- 非ブロック `DO` 文および算術 `IF` 文は Fortran 2018 で削除。Intel® Fortran コンパイラーは FORTRAN 標準で削除された機能をフルサポート。
- `COMMON`、`EQUIVALENCE` および `BLOCKDATA` 文が古い機能に
- ラベル付き `DO` ループが古い機能に
- `DO CONCURRENT` 構文の変数の局所性を `DO CONCURRENT` 文で宣言可能に
- 組み込みプロシージャの固有名が古い機能に
- `FAIL IMAGE` 文により実際のイメージ失敗を待つ必要なく失敗したイメージの回復コードのデバッグを許可
- オプション引数 `STAT=` を `ATOMIC_REF` および `ATOMIC_DEFINE` 組み込みプロシージャに追加
- オプションの `STAT=` および `ERRMSG=` 指定子を `MOVE_ALLOC` 組み込みプロシージャ、イメージセクター、`CRITICAL` 文と構文に追加
- `ATOMIC_ADD`、`ATOMIC_AND`、`ATOMIC_CAS`、`ATOMIC_FETCH_ADD`、`ATOMIC_FETCH_AND`、`ATOMIC_FETCH_OR`、`ATOMIC_FETCH_XOR`、`ATOMIC_OR`、および `ATOMIC_XOR` アトミック・サブルーチンを実装
- `CO_BROADCAST`、`CO_MAX`、`CO_MIN`、`CO_REDUCE`、および `CO_SUM` 集合サブルーチンを実装
- ランク引き継ぎ仮引数の操作を許可する `SELECT RANK` 構文を実装
- Fortran 2018 で必要な非標準組み込みプロシージャとモジュールの使用をコンパイラーで診断
- `ISO_C_BINDING`、`IEEE_ARITHMETIC`、および `IEEE_EXCEPTIONS` 組み込みモジュールの変形組み込み関数を宣言式で許可
- `IEEE_GET_ROUNDING_MODE` および `IEEE_SET_ROUNDING_MODE` 組み込みモジュール・プロシージャでオプション引数 `RADIX` を指定可能に
- オプションの `ROUND` 引数を `IEEE_ARITHMETIC` 組み込みモジュールの `IEEE_RINT` 関数に追加
- `IEEE_ARITHMETIC` 組み込みモジュールに `IEEE_FMA`、`IEEE_SIGN_BIT`、`IEEE_NEXT_UP` および `IEEE_NEXT_DOWN` 関数を追加
- `IEEE_MAX`、`IEEE_MIN`、`IEEE_MAX_MAG`、および `IEEE_MIN_MAG` 組み込みモジュール・プロシージャを実装
- `IEEE_INT` および `IEEE_REAL` 組み込みモジュール・プロシージャを実装
- `IEEE_EXCEPTIONS` 組み込みモジュールに `IEEE_GET_MODES` および `IEEE_SET_MODES` 組み込みモジュール・プロシージャを使用した `IEEE_MODES` の保存および回復に使用できる新しい派生型、`IEEE_MODES_TYPE` を追加
- 新しい丸めモード、`IEEE_AWAY` を追加
- `SUBNORMAL` が `DENORMAL` と同義に
- `IEEE_QUIET_EQ`、`IEEE_QUIET_NE`、`IEEE_QUIET_LT`、`IEEE_QUIET_LE`、`IEEE_QUIET_GT`、`IEEE_QUIET_GE`、`IEEE_SIGNALING_EQ`、`IEEE_SIGNALING_NE`、`IEEE_SIGNALING_GT`、`IEEE_SIGNALING_GE`、`IEEE_SIGNALING_LT`、および `IEEE_SIGNALING_LE` 組み込みモジュール・プロシージャを実装

OpenMP* 5.0 の新機能

- SIMD ディレクティブの IF 節
- SIMD ディレクティブの NONTEMPORAL 節
- SIMD および TARGET SIMD 構文の包括/排他スキャン操作

コンパイラーの動作の変更

- 以前のコンパイラー・バージョンでは、派生型宣言の中で型引数を宣言する前に PRIVATE 文または SEQUENCE 文を使用することができました。これは誤りです。インテル® Fortran コンパイラー 19.1 では、この動作は許可されなくなりました。
- 呼び出しが成功した場合、INQUIRE 文はフルパスおよび標準化されたファイルパスを使用します。変更の例:
"bar/foo" としてファイルを開く。
変更前:
INQUIRE 文で "./bar/foo"、"bar//foo" または "bar/../bar/foo" が開かれているかどうか問い合わせる。動作しない。
変更後:
INQUIRE 文で "./bar/foo"、"bar//foo" または "bar/../bar/foo" が開かれているかどうか問い合わせる。動作する。
- コンパイラー 19.0 では、ISO_C_BINDING 組込みモジュールの C_F_POINTER 例外を含むすべてのプロシーチャーは Fortran 2018 標準に従って PURE にされていました。解釈 18/007 で、C_F_PROCPOINTER PURE にすることは誤りであったことが示されました。コンパイラー 19.1 では、この解釈に従って C_F_PROCPOINTER IMPURE にするように変更されました。

新規および変更されたコンパイラー・オプション

詳細は、コンパイラーのドキュメントを参照してください。次の新しいコンパイラー・オプションが追加されました。

- [-assume \[no\]old inquire recl](#)
- [-assume \[no\]old ldout zero](#)
- [-check \[no\]udio iostat](#)
- [-warn \[no\]externals](#)

廃止予定のコンパイラー・オプションのリストは、ドキュメントのコンパイラー・オプションのセクションを参照してください。

`-assume [no]old_inquire_recl`

未接続のユニットまたはストリームアクセス向けに接続されたユニットへの INQUIRE 文の RECL= 指定子の値を指定します。

`-assume [no]old_ldout_zero`

list-directed 出力で生成される浮動小数点ゼロの形式 (old_ldout_zero は指数形式、no_old_ldout_zero は分数形式) を指定します。

`-check [no]udio_iostat`

ユーザー定義の派生型入力/出力プロシージャが実行されたときに標準規格チェックを行うかどうかを指定します。

`-warn [no]externals`

明示的なインターフェイスの (または EXTERNAL と宣言されていない) 仮プロシージャやプロシージャ呼び出しで警告を出力するかどうかを指定します。

終了予定のサポート

終了したサポート

ループ・プロファイラー

ループ・プロファイラー機能は Intel® Fortran コンパイラー 19.1 で削除されました。ループ・プロファイラーに関連する次のコンパイラー・オプションは削除されました: `-profile-loops=keyword`、`-profile-loops-report=value`、`-profile-functions`、`-guide-profile`。

32 ビット・ホストへのインストールのサポートを終了

32 ビット・ホストへのインストールのサポートは、Intel® Parallel Studio XE 2018 で終了しました。32 ビット・ターゲット用コードの生成は 64 ビット・ホストでサポートされます (`-m32` コンパイラー・オプションを使用)。

Intel® Xeon Phi™ x100 製品ファミリー (開発コード名 Knights Corner) のサポートを終了

Intel® Xeon Phi™ x100 製品ファミリー (開発コード名 Knights Corner) は、2017 年 1 月に終息したため、Intel® Parallel Studio XE 2017 でのみサポートされます。Intel® Xeon Phi™ x100 製品ファミ

リーに対するインテル® Parallel Studio XE 2017 のサポートは、製品終息から 3 年後の 2020 年 1 月に終了します。有効なサポートサービスをお持ちの方にサポートが提供されます。

既知の問題

OpenMP* 4.5 の OMP THREADPRIVATE と共通ブロック名の特定の使用法が診断されない

OpenMP* 4.5 の規則では、共通ブロック名を指定する `THREADPRIVATE` ディレクティブが 1 つのプログラムユニットにある場合、同じ名前を指定する `COMMON` 文を含むすべてのプログラムユニットで、最後の該当する `COMMON` 文の後に `THREADPRIVATE` ディレクティブがなければなりません。インテル® Fortran コンパイラーでは、この使用法が適切に診断されません。

例えば、次のプログラムは OpenMP* 4.5 仕様に準拠していませんが、`ifort` は `OMP THREADPRIVATE` 文に続く 2 つの `COMMON` 文に対してエラーメッセージを出力しません。

```
PROGRAM ex1
  COMMON /common_blk1/x
  !$OMP THREADPRIVATE (/common_blk1/)

  COMMON /common_blk1/y
  COMMON /common_blk1/z

END PROGRAM
```

Fortran Co-Array アプリケーションが FAIL IMAGE の実行後にハングする

現在の `FAIL IMAGE` 実装には `FAIL IMAGE` 文を実行した後に発生する既知の問題があります。詳細は、[この記事](#) (英語) を参照してください。

Fortran 2008 および Fortran 2018 機能の概要

インテル® Fortran コンパイラーは、Fortran 2008 標準のすべての機能と Fortran 2018 標準のほとんどの機能をサポートします。現在のバージョンでサポートされている新しい Fortran 2018 の機能は、[Fortran 2018 の新機能](#) セクションを参照してください。

必要に応じて、[Fortran 2008 標準](#) (PDF、英語) および [Fortran 2018 標準](#) (PDF、英語) を参照してください。

著作権と商標について

最適化に関する注意事項

インテル® コンパイラーでは、インテル® マイクロプロセッサに限定されない最適化に関して、他社製マイクロプロセッサ用に同等の最適化を行えないことがあります。これには、インテル® ストリーミング SIMD 拡張命令 2、インテル® ストリーミング SIMD 拡張命令 3、インテル® ストリーミング SIMD 拡張命令 3 補足命令などの最適化が該当します。インテルは、他社製マイクロプロセッサに関して、いかなる最適化の利用、機能、または効果も保証いたしません。本製品のマイクロプロセッサ依存の最適化は、インテル® マイクロプロセッサでの使用を前提としています。インテル® マイクロアーキテクチャーに限定されない最適化のなかにも、インテル® マイクロプロセッサ用のものがあります。この注意事項で言及した命令セットの詳細については、該当する製品のユーザー・リファレンス・ガイドを参照してください。

注意事項の改訂 #20110804

本資料に掲載されている情報は、インテル製品の概要説明を目的としたものです。本資料は、明示されているか否かにかかわらず、また禁反言によるとよらずにかかわらず、いかなる知的財産権のライセンスも許諾するものではありません。製品に付属の売買契約書『Intel's Terms and Conditions of Sale』に規定されている場合を除き、インテルはいかなる責任を負うものではなく、またインテル製品の販売や使用に関する明示または黙示の保証 (特定目的への適合性、商品適格性、あらゆる特許権、著作権、その他知的財産権の非侵害性への保証を含む) に関してもいかなる責任も負いません。インテルによる書面での合意がない限り、インテル製品は、インテル製品の欠陥や故障によって人身事故が発生するような用途向けに使用することを前提としたものではありません。

インテル製品は、予告なく仕様や説明が変更されることがあります。機能または命令の一覧で「留保」または「未定義」と記されているものがありますが、その「機能が存在しない」あるいは「性質が留保付である」という状態を設計の前提にしないでください。これらの項目は、インテルが将来のために留保しているものです。インテルが将来これらの項目を定義したことにより、衝突が生じたり互換性が失われたりしても、インテルは一切責任を負いません。この情報は予告なく変更されることがあります。この情報だけに基づいて設計を最終的なものとししないでください。

本資料で説明されている製品には、エラッタと呼ばれる設計上の不具合が含まれている可能性があり、公表されている仕様とは異なる動作をする場合があります。現在確認済みのエラッタについては、インテルまでお問い合わせください。

最新の仕様をご希望の場合や製品をご注文の場合は、お近くのインテルの営業所または販売代理店にお問い合わせください。

本資料で紹介されている資料番号付きのドキュメントや、インテルのその他の資料を入手するには、1-800-548-4725 (アメリカ合衆国) までご連絡いただくか、<http://www.intel.com/design/literature.htm> (英語) を参照してください。

インテル・プロセッサ・ナンバーはパフォーマンスの指標ではありません。プロセッサ・ナンバーは同一プロセッサ・ファミリー内の製品の機能を区別します。異なるプロセッサ・ファミリー間の機能の区別には用いません。詳細については、http://www.intel.co.jp/jp/products/processor_number/ を参照してください。

インテル® Fortran コンパイラーは、インテルのソフトウェア使用許諾契約書 (EULA) の下で提供されます。

詳細は、製品に含まれるライセンスを確認してください。

Intel、インテル、Intel ロゴ、Intel Core、Xeon、Intel Xeon Phi は、アメリカ合衆国および / またはその他の国における Intel Corporation またはその子会社の商標です。

* その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

© 2019 Intel Corporation. 無断での引用、転載を禁じます。

コンパイラーの最適化に関する詳細は、[最適化に関する注意事項](#)を参照してください。