

# 8

---

第8章

---

## Fortran プログラムの 最適化

インテルコンパイラは、最適化のためのさまざまなオプションや機能を提供していますが、ほとんどの場合、次にリストされた最適化手法を用いてその利点を得ることができます。

### 最適化オプション

/O1、/O2、/O3 や /fast などのような基本的な最適化オプションを使用すると、アプリケーションに最適な方法を見つけることができます。これらのオプションを使用して得られる性能の違いを、各コンパイルごとに測定します。アプリケーションが特定のアーキテクチャを対象としている場合は、ターゲットプロセッサ向けのコンパイラオプションと最適化オプションを組み合わせて試してください。/Qx オプションと/Qax オプションを組み合わせて、特定のインテルプロセッサ向けに最適化されたコードと IA-32 アーキテクチャをベースとしたほとんどのプロセッサで実行できる汎用コードの両方を作成します。他の最適化手法を適用する前に、プロセッサ固有のオプションと基本的な最適化オプションを組み合わせてみてください。

### IPO と PGO

プロシージャ間の最適化 (IPO) とプロファイルに基づく最適化 (PGO) を試します。最適化適用後のパフォーマンスを測定し、アプリケーションのパフォーマンスが向上したかどうかを調べます。

トップダウン方式の反復手法を利用します。インテル VTune パフォーマンスアナライザやコンパイラレポートなどのパフォーマンス監視ツールを使用し、パフォーマンスの妨げとなるコードを特定し、解決します。

### 並列化

インテルコンパイラでサポートされている並列プログラミングの3つの主要機能 (OpenMP、自動並列化、自動ベクトル化) について説明します。

## 8-1 基本的な最適化の概要

インテルコンパイラは、IA-32 アーキテクチャ (32 ビットアプリケーション)、インテル 64 アーキテクチャ (32 ビットアプリケーションと 64 ビットアプリケーション)、または IA-64 アーキテクチャ (64 ビットアプリケーション) をベースとしたプロセッサ向けのバイナリを生成することができます。

デフォルトでは、コンパイラはコンパイル時間とアプリケーションのランタイムパフォーマンスのバランスがとれた最適化オプションの組み合わせが選択されます。アプリケーションの

構成に応じて手動で最適化を選択することもできます。

次の表は、コンパイラがサポートしている基本的な最適化オプションをまとめたものです。これらのオプションによって効率的な結果を素早く得ることができます。

オプション	説明
/O3	コードの速度におけるさらに強力な最適化を有効にします。計算を実行する、または大きなデータセットを処理するループを含むコードに推奨します。
/O2	コードの速度に影響を与えます。これはデフォルトオプションです。何も指定しない場合、コンパイラではこの最適化レベルが使用されます。
/O1	コードサイズと局所性に影響を与えます。特定の最適化を無効にします。
/fast	ランタイムのパフォーマンス用の一般的かつ推奨される最適化オプション一式を有効にします。アーキテクチャの依存性を招くことがあります。/fast オプションを構成する推奨オプションはコンパイラのバージョンで異なります。
/Od	最適化を無効にします。アプリケーションのデバッグ中は、このオプションを使用して素早くコンパイルを行います。

インテルコンパイラのさまざまな最適化を適用して、アプリケーションのパフォーマンスを素早く向上させることができます。基本的な最適化オプションの他、コンパイラではソースコードの宣言子、最適化されたライブラリルーチン、パフォーマンス拡張ユーティリティなどと併用して有効になるその他の最適化も行います。

### 8-1-1 一般的な最適化

このトピックでは、最も一般的なコードの最適化オプションをリストし、IA-32 アーキテクチャ、インテル 64 アーキテクチャ、IA-64 アーキテクチャで共通の特性および各アーキテクチャの一般的な動作について説明します。

オプション	説明
/O1	<p>好ましい（より小さな）コードサイズとコードの局所性に最適化します。ほとんどの場合は、このオプションよりも/O2を推奨します。</p> <p>この最適化は、通常コードサイズを増加させる可能性があるいくつかの最適化を無効にします。この最適化レベルは、分岐が多く、実行時間がループ内のコードに支配されない、非常に大きなコードサイズのアプリケーションのパフォーマンスを向上させます。一般に、この最適化レベルは次のことを行います。</p> <p>グローバルな最適化を有効にします。</p> <p>組み込み関数の認識と組み込み関数のインライン展開を無効にします。</p> <p>IA-64 アーキテクチャ：</p> <p>このオプションは、ソフトウェアのパイプライン化、ループのアンロール、およびグローバルコードスケジューリングを無効にします。</p>
/O2	<p>速度について最適化します。デフォルトの最適化レベルです。最適化レベルを指定しなかった場合、コンパイラはこの最適化レベルを自動的に使用します。一般的に推奨される最適化レベルです。しかし、他のコンパイラオプションを指定して、このレベルで通常行われる最適化と同じ最適化を行うこともできます。</p> <p>一般に、コードサイズは/O1を使用して生成されたコードサイズよりも大きくなります。</p> <p>このオプションは、パフォーマンスの向上のために関数のインライン展開と次の機能を有効にします。定数伝播、コピー伝播、不要コードの排除、グローバルレジスタ割り当て、グローバル命令スケジューリング、スペキュレーションコントロール、ループのアンロール、コード選択の最適化、部分冗長の排除、ストレンジスリダクション／誘導変数の簡略化、変数名の変更、例外処理の最適化、末端再帰、ピープホールの最適化、構造体代入の最適化、および不要ストアの排除。</p> <p>IA-64 アーキテクチャ：</p> <p>速度の最適化を有効にします。グローバルコードスケジューリング、ソフトウェアのパイプライン化、プレディケーション、スペキュレーション、およびデータプリフェッチが含まれます。</p>
/O3	<p>/O2の最適化に加えて、プリフェッチ、スカラ置換、キャッシュブロッキング、ループ変換、およびメモリアクセス変換などのより強力な最適化を有効にします。</p> <p>最大の実行速度が得られる可能性のある最適化を行いますが、ループ変換およびメモリアクセス変換が行われない限り、パフォーマンスが向上しない場合があります。このオプションによる最適化は、コードによっては/O2の最適化よりも遅くなります。</p> <p>このオプションは、浮動小数点演算を多く使用するループや大きなデータセットを処理するループを含むアプリケーションに推奨します。</p> <p>他のコード最適化オプションと同じように、このオプションの動作はアーキテクチャおよびオペレーティングシステムによって異なります。</p> <p>IA-32 アーキテクチャ：</p> <p>/Qax または /Qx と組み合わせて使用すると/O2よりも詳細にデータの依存性を解析します。その結果、コンパイル時間が長くなることがあります。</p> <p>IA-64 アーキテクチャ：</p> <p>技術計算を行うアプリケーション（ループを多用するコード）の最適化、つまりループとデータプリフェッチの最適化を有効にします。</p>

(次ページへ続く)