Chapter-4 C++標準テンプレートライブラリの利用(上級)～

4.2 計算量オーダーについて

さて、ここから可変長配列、双方向リスト、マップなどの特性をアルゴリズムの詳細な実装を交えて解説をしていきたいのですが、その前にアルゴリズムの計算量オーダーというものについて勉強しておきましょう。計算量オーダーとは、そのアルゴリズムを実行したときの処理時間を図るための物差しです。可変長配列、双方向リスト、マップなどの古典的なアルゴリズムは、ネットを調べると計算量オーダーを見つけることができます。また、これら以外の多種多様なアルゴリズムも計算量オーダーに関する記述を見つけることができます。適切なアルゴリズムを選択するためには、この記述の意味が分かる必要があります。また、この記述の意味が分かると、自身が作成している処理の計算量も分かるようになってきます。これが分かるようになってくると、「あ・・・俺・・・今、ヤバいコード書いてる・・・。このコードでプログラムを実行すると、いつまでたっても処理終わらなくなるかも。。。」ということが考えられるようになってきます。計算量オーダーは次のような表記がされます。

1. O(1)
2. O(N)
3. O(N^2)
4. O(logN)
5. O(NlogN)

例えば、可変長配列、std::vectorの[]演算子の計算量オーダーはO(1)です。std::map::find()関数の計算量のオーダーはO(logN)です。std::find()関数を利用して、std::vectorやstd::listに格納されている要素を検索する計算量はO(N)です。 さて、ここで、質問です。10000万を超える膨大なデータから要素を検索する場合、std::mapとstd::vectorのどちらにデータを記憶したほうが良いでしょうか？ この場合、std::mapを選択することが最も良いです。理由は、std::find()関数を利用して、std::vectorの要素を検索する時間がO(N)なのに対して、std::map::find()関数の検索時間はO(logN)だからです。 計算量オーダーの読み方が分かっている人はこれがすぐに分かるようになります。では、計算量オーダーについて説明していきます。

4.2.1 O(1)

アルゴリズムで扱うデータ数が多かろうが、少なかろうが、処理時間は一定であることを表しています。std::vectorの[]演算子の計算量オーダーがこれに該当します。 多くの場合で、O(1)のアルゴリズムは最速である場合が多いです。

4.2.2 O(N)

O(N)という表記はアルゴリズムが扱うデータ数(N)の量に応じて、比例して計算時間が増加することを表しています。この手のアルゴリズムは少数のデータを扱うときには十分な速度が期待できますが、データ数が多くなってくると注意が必要になってきます。 std::find()関数の検索時間はO(N)です。

4.2.3 O(N^2)

O(N^2)という表記はアルゴリズムが扱うデータ数(N)の量応じて、N^2、つまり指数関数的に計算量が増加していくことを表しています。このオーダーのアルゴリズムがもっとも注意が必要です。バブルソートの計

算量はO(N^2)です。大量のデータを扱うときに、このアルゴリズムを選択すると最悪の結果となるでしょう。

4.2.4 O(logN)

計算量がO(logN)のアルゴリズムは、扱うデータ数(N)に応じて、2を底とする対数の時間の計算量となります。この計算量の表記を理解するのに、数学の対数の知識はなくても構いません。この計算量のアルゴリズムは大量のデータを扱うときに、最もすぐれたパフォーマンスを出す可能性があるアルゴリズムです。もし、計算量がO(N)やO(N^2)のアルゴリズムの変更、最適化を行う場合は、まず計算量がO(logN)になることを目指すべきです。 例えば、O(N)のアルゴリズムで100万個のデータを扱うときに、100時間かかっているときに、そのアルゴリズムをO(logN)になるように改良すると、処理時間は7.2秒になります。 なぜこのように高速になるのかは、std::map::find()の実装の詳細を見るときに説明します。

4.2.5 O(NlogN)

計算量がO(NlogN)のアルゴリズムは、扱うデータ数(N)に応じて、2を底とする対数時間に比例して増加していく計算量となります。こちらも大量のデータを扱うときに、優れたパフォーマンスを出す可能性があるアルゴリズムです。有名どころではクイックソートなどがこの計算量となります。 例えば100個のデータをバブルソートにかけると10000時間かかるとき、クイックソートを利用すると600時間に短縮することができます。

評価テスト

次の評価テストを行いなさい。
評価テストへジャンプ