

SuperCon

解説

SuperCon:

スーパーコンピュータを使った 高校生向けプログラミングコン テスト

松田 裕幸 東京工業大学 matsuda@gsic.titech.ac.jp

SuperCon とは

SuperCon とは、Supercomputer Programming Contest for High School Students の略称です^{☆1,1)}。1995 年、東京工業大学においてスタートしたときは、現在のように特に明確なメッセージを必ずしも持っていたわけではありませんでした。世の中で若者の理工系離れが懸念され、大学も何かしなければならぬとさまざまな試みがなされ始めたとき、たまたま、大学にスーパーコンピュータ (Cray C90) があり、これを活用しようと考えたにすぎません。当初、コンテストという形ではなく、スーパーコンピュータ (以下、スパコンと略) に関する公開講座にしたかどうかという意見が強くありました。スパコンは主として科学計算に使われますが、そこには高度な科学的知識と高いプログラミング能力が要求されます。そんなコンピュータを、高校生^{☆2} が使いこなして、プログラミングを主体としたコンテストなんかできるはずがないというのが当時の大方の見方でした。したがって、たとえば、コンテストのためにスパコンに使用制限をかけざるを得ないことに対し、第 3 回大会あたりまでは、何人かのユーザからは手厳しい批判を受けました。どうしてコンテスト形式にしたかという問に対して正直あまり明確な答えは持っていません。しかし、たとえば、スパコンに関する公開講座を開くにしても、単純に

講義を聞くよりは、実際に、触ってみる方がはるかにいいにきまっていますが、パソコンと異なり、簡単に組み立ててみるというわけにはいきません。ならば、プログラミングを通じてスパコンを体験してもらおう、そして、それならば、いっそ、コンテストという形式の方がより高校生に対してはチャレンジングで面白いのではないかと、というのが当時、コンテストの開催を決定した関係者の思いだったと記憶しています。

コンテストは予選と本選からなり、どちらもプログラミングの課題が出されます。予選課題はスパコン上で解くことは想定しておらず、手元のパソコン等で計算すればよく、そのプログラムと結果から、正確さ・速度・コーディング・レポートによって上位 10 組が選抜されます^{☆3}。毎年、40 組弱の応募がありますが、これは、予選の解答をもって応募とするととなっているためであり、潜在的な応募者はもう少し多いと想像しています。本選は、東京工業大学で行います。インターネットの時代、遠隔での参加も可能であるのに対し、依然、大学に来てもらうことにこだわっているのは、本コンテストが単なる競争ではなく、プログラミングを通じての高校生たちの交流も大きな目的の 1 つになっているからです。予選課題は 6 月上旬頃に公開され、6 月末を応募締め切りとしています。

一方、本選は、8 月上旬前後の 1 週間を使い、課題と取り組んでもらいます。たいていの場合、初日にスパコンとスパコン上のプログラミング技法について、簡単な講習を行い、合わせて課題の説明を行います。課題については事前に概略を公開してきましたが、10 回大会からは、非公開としました。インターネット上の掲示板等での議論情報が参加者の問題解決行動へ与える影響が好ましいものと思われないと判断したためです。ただし、一定時間内に決められた個数の問題を解く試験型のコンテストと異なり、期間中、他の組と情報交換したり、イ

☆1 2005 年度第 11 回大会からは、Supercomputer の代わりに、Supercomputing に変更することになりました。

☆2 高等専門学校生も 3 年生までは高校生として扱っています。

☆3 1 組は 3 人以下で、当初は 1 人での参加も可としていましたが、現在では 2 人、もしくは 3 人でなければなりません。また、10 組という数も年によって変動がありましたが、現在は正確に 10 組となっています。これ以外にも、開催期間、参加方法、コンテスト方法、副賞等に関し、年によって異なっています。したがって、ここに述べることは、特に言及がない限り、最近の運営内容に基づいて述べるものとします。



図-1 問題に取り組む中国チームと様子を見守る学生チューター

インターネットへアクセスすることは許されています。しかし、解法に関するヒントをコンテスト参加者以外から受けることは禁止しています。これを厳密に管理することは難しいので、参加者への誓約書という形の中でゆるく縛りを与えています。

プログラムの設計とコーディングには3日ないし4日(5日のときもあり)が与えられます。大学で計算機資源を使える時間は朝8時から夜8時までの12時間です。かつては、それ以外の時間もインターネットによる接続を許していたときもありましたが、全員が同じ便宜を受けられる環境にないため、今は時間外の外部からの接続はできないようにしています。課題について、ここではまだ説明していないため、理解しにくいかと思いますが、この数日間という時間は、プログラムの設計・コーディングとしては、非常に少ないというのが毎回の参加者の感想です。しかしながら、遠隔地からの参加者を考えると、これ以上の期間延長は無理だと考えています。

コンテストへの応募は日本全国からあり、本選に参加した場合、遠隔地(関東圏外)からの参加者には若干の補助を出していますが、大学の公的財源をこうしたことに使うことはできず、外部の企業、後援会等に援助をお願いしています。その関係で、援助できる金額は、年によって異なります。また、コンテスト上位入賞者へは、さまざまな副賞を出してきました(これらも支援企業・団体の援助によってまかなわれています)。当初は、パソコン等を個人へ、ついで学校への寄付と形を変えてきましたが、いまではメダル等によって代替し、高額な副賞を個人へ出すことはやめています。本コンテスト優勝者への副賞としてとてもユニークだったのが、その年のSupercomputing(米国開催)に参加し、また、開催地近隣の大学・研究機関を訪問するというものでした。しかし、これは諸々の理由により第4回大会を最後に取りやめになっています。

海外の高校生との交流もはかってきました。第2回と第9回はインターネットを通じ、それぞれ、ロシアと米国、および、タイ王国から参加があり、また、第10回大会では、中華人民共和国、大韓民国、シンガポール共和国、タイ王国からそれぞれ1組を日本に招待しました²⁾。残念ながら、現在の運営規模では、恒常的に海外の高校生に参加してもらうことは難しいというのが実情です。

コンテストの運営は大学内の学術国際情報センター(旧、総合情報処理センター)によって行われてきました。正確には、コンテスト実施委員会が上部にあり、その下部の実働組織として情報センターが位置します。次の11回大会からは大学外部のメンバも加わる予定になっていますが、10回大会までは、大学内の研究者から構成されてきました。コンテストの運営にあたっては重要な事柄がたくさんありますが、最大のものが課題作成です。情報系の先生を中心に、毎年、数カ月をかけ、課題を決定しています。

コンテスト課題を作る

コンテストの成否は課題の作成にすべてかかっているといっても過言ではありません。それだけに、毎回、課題の作成には非常に苦労しています。コンテストでは今まで表-1のような課題を考えてきました。

予選課題では、問題の理解力とプログラミング能力をみることに主眼を置き、新たなアルゴリズムの発見にはそれほど重きを置いていません。そして、実行時間に関し優劣をつけるために、データ数が増えるに伴い指数関数的に時間計算量が増加するタイプの問題を意識して作っています。しかし、第9回大会のときのように、動的計画法を使わなくても、わずかな工夫により定数時間で解けてしまう問題を作るという失敗もありました。それでも、おおむねうまく問題が作られているせいか、ほとんどの場合、予選審査において、実行時間だけで10組+ α 程度を選ぶことができます。レポートによる順位付けは、基本的には実行時間による判別の補助的な役割しか与えていないのですが、第1回大会優勝者のように、実行時間ではトップ10組に入っていなかったのに、レポートの内容で敗者復活した例もあります。ただし、これは唯一の例外となっています。

本選は、他のコンテスト、たとえば、情報オリンピック(IOI: International Olympiad in Informatics)、世界大学生プログラミングコンテスト(ICPC: International Collegiate Programming Contest)などと異なり、一定時間、たとえば、5時間の間に10問解く(解答プログラムを作る)といった形式になっていません。こうした

回	年	本選課題	予選課題
第1回	1995	石選び問題 [ナップザック暗号問題]	2 の自然対数の計算 [多倍長計算]
第2回	1996	カフェテリア問題 [待ち行列シミュレーション]	π の自然対数の計算 [多倍長計算]
第3回	1997	三角ピリヤードの周期軌道探索 [量子力学の問題]	つじつま合わせ問題 [文字列照合問題]
第4回	1998	板詰め問題 [配置スケジュール問題]	三次方程式の解 [多倍長計算]
第5回	1999	ライフゲーム [絶滅しない初期値発見]	オイラー定数の計算 [多倍長計算]
第6回	2000	道路網建設大作戦 [シュタイナー木]	道路網建設大作戦 [格子点に限定したシュタイナー木]
第7回	2001	N 体問題	連鎖行列式の最小コスト計算 [動的計画法]
第8回	2002	タンパク質の構造予測 [組合せ問題]	漸化式の解 [動的計画法]
第9回	2003	最大タンパク質ファミリーの探索 [クラスタ問題]	最長連鎖探索
第10回	2004	暗号画像の解読	スーパーコン数の計算 [素因数分解]

表-1 過去のコンテスト課題一覧

種類のコンテストでは、主要なアルゴリズムに精通して、過去の問題に対しよく練習を重ね、かつ、短時間でプログラムを正確に組む力を持っていることが重要となってきます。喩えは適切ではないかもしれませんが、大学受験で求められる、いわゆる勉強成果型の能力が必要とされるといえます。これに対し、本コンテストは、「プロジェクト型」の課題を前提とし、正確さと同時に、より高速のプログラムを作ることを求められます。時には正解を得ることが難しい場合には、決められた実行時間内に近似解を出すといった、戦略的な解法も必要になってきます。いずれの場合も、決して汎用的なプログラムを作る必要はなく、問題仕様をクリアする範囲において、最大限の高速化に挑戦します。IOI やコンテスト OB として ICPC を経験したものが、異口同音に本コンテストの課題の難しさに言及するのは、この課題スタイルによるところが大きいと思われる。

参加者は短時間に課題仕様を理解し、アルゴリズムを決定し、プログラムを書き、ついで、高速化のためにアルゴリズムあるいはプログラムを改良していきます。数日間という時間は、このための時間としてはかなり短く、その一方で、参加 10 組の間にある程度の差を生じさせる（分かりやすくいえば、成績が上、中、下にうまく分かれる）には、課題仕様が非常に重要となってきます。数日間という日程の中で、半日程度でアルゴリズムを決定し、1 日半程度で基本的なプログラムは完成し、ついで、並列化等により高速化する部分に残りの時間を費やすことを考えると、問題仕様はアルゴリズムの決定と高速化に大きく影響するからです。過去最大の失敗例は第 2 回大会のカフェテリア問題です。課題自体は興味深い問題だったのですが、仕様が複雑すぎて、大会期間が終

わりかけていた段階でほとんどの組がプログラム自体完成しておらず、まして並列化にはまったく手をつけることができませんでした。優勝した組の並列化プログラムより、主催者側の逐次プログラムの方がよりよい答えをより短い時間で解答したのも仕様の悪さによるところが一番大きかったとみています。実際、コンテスト参加者へのアンケートでもこの点が強く指摘されていました。

各組に割り当てられる審査時間は 3 分です。本来ならば 1 時間程度の実行時間で審査したいのですが、10 組以上を審査するとなると、審査環境の準備も含め相当な時間が必要となっているからです。そして、この 3 分という時間が、問題を作る上でもう 1 つの難関となっています。単純な話ですが、全組が数秒で終わっても困るし、逆にどの組も 3 分で終わらなかつたら困ります。上位はおおよそ数秒から 1 分程度、中位の組は 2 分前後、残りは 3 分前後からそれ以上という風に分かれるのが理想です。しかし、事前に参加者がどういうプログラムを書くかは分かりません。そこで、前もって、課題で必要とするデータサイズと実行時間の関係を念入りに何度もシミュレーションします。第 1 回は、この推定が大幅に狂い、主催者側が数分とみていた計算を優勝組は 4.7 秒という驚異的な時間で解いてしまうという大ハプニングがありました（このときは、解法そのものが予想外ということもあったのですが）。いずれにしても、本選開始 1 カ月前はほぼ毎日、このデータサイズの予測に時間を費やしています。こうした努力のせいか、ここ数回の大会では、成績が上位、中位、その他とうまくバランスするようになってきました。第 10 回記念大会の課題「暗号画像の解読」では、できるだけ多くの解読が求められたのですが、その解読数において、前半圧倒的

な強さを誇った中国組のプログラムを、日本チームが制限時間 5 秒前で追い抜いたのは劇的でした²⁾、☆⁴⁾。

プログラミングの楽しみ

コンテストに参加する高校生たちは多少の温度差はあるでしょうが、全員が全員プログラミング大好き少年、少女たちです。そして、周囲に同じような興味を持つ人が少ないというのも共通しています。そのため、プログラミング能力を競い、プログラミングという共通の趣味を持つ同世代の若者たちと交流できる場が提供されていることは非常にうれしいようです。

今は凝った商用ツールが多くなり、かつてのように自分で使うプログラムは自分で作るといった楽しみは少なくなってきました。そのため、コンテストに参加した高校生たちに質問すると、プログラミングには興味あるけど、それをどういふ分野や用途に応用していいのかわからないという人が結構います。よく聞くのは学園祭の出しものとしてのコンピュータ占いや、グラフィクスと連動したコンピュータゲームなどです。プログラミング言語は勉強したけど、適当な応用先を知らない、という高校生たちは案外多いのではないのでしょうか。

どんな複雑なプログラムも、代入・条件分岐・繰り返しという単純な構造の組合せにしかすぎません。しかし、これはでき上がったものをその構成要素に分解するときと言えることであり、元の仕様から完成プログラムを作るのはこんなに簡単ではありません。そこには、この単純な構造をどのように組み合わせるかというロジック、あるいは手順（アルゴリズム）が必要となります。その手順の中には、すでに学んだことのあるパターンが含まれていることもあるでしょうが、未知のパターンを再発見することもあるはずで、予選の課題には動的計画法によって指数オーダーの計算をデータサイズのオーダーに変えることができる問題がいくつかありましたが、動的計画法を知らず、再発見した高校生がいました。こうした、アルゴリズムやパターンをコーディングしていく過程にプログラミングの面白さがあるのではないのでしょうか。一度このコンテストに参加した多くの高校生たちがまた次回に応募してくるのも、こうした楽しみを体験できる課題と機会を本コンテストが与えてきたからだと思います。

☆⁴⁾ 本選課題は、さまざまな数理的事象を背景に作られており、その応用範囲も存在するのですが、一般的には、これらの問題がどんな役に立つのだろうかという疑問も当然、聞こえてきます。そこで、第 11 回大会からは、問題を公募することにしました。コンテストページにアクセスし、ぜひ、そのような問題を考えてみてください。

高校生が書くプログラム

予選応募には、1 と 0 の整数フラグの組合せに対し、1 と 0 をビットとみなし全体を整数化し、整数単位の論理演算などによって高速化しているプログラムをよく見かけます。ただし、整数演算が苦手なスパコン上で動かすとかえって遅くなってしまいますが、木やグラフに関する基本的なアルゴリズムを自分で勉強して知っている子もいます。第 1 回大会の本選課題は暗号に関する問題でしたが、優勝組と 2 位の組は暗号鍵を見つけることあきらめ、想定されるすべての組合せを作り出し、その中から対象となる組合せを見つけるという力づくの方法で、それぞれ 4.7 秒、7 秒という驚異的な時間で解答を出しました。ちなみに、3 位は 10 分時間切れ、そのまま実行を続け 1 時間以上たっても答えは出ませんでした。

優勝組のプログラムは、与えられた複数データのパートナー（2 の 40 乗の組合せ！）をハッシュテーブルへのインデックスとして持つというアイデアを思いつき、準優勝のプログラムはすべての組合せを二分木に構成し、解の探索のデータサイズの log のオーダーとなるようにしました。このとき、使用したスパコンは Cray C90 で、高速化手法はベクトル並列を利用することでした。テーブル・木を構成するにはベクトル並列プログラムは非常に向いており、上記時間のほとんどすべてはこのテーブル・木の構成に費やされています。

コンテストに参加してくる高校生たちは趣味としてプログラムは書いていますが、スパコン向けプログラムを書くのは、全員初めてです。時間単位でみると、コンテスト初日に半日程度の講習を受けたあと、1 日程度で、ベクトル化あるいは並列化用コードに書き直しているわけです。かつ、このときの優勝チームは約 30 本ほどのプログラムを作っては計算時間による評価を行っていました。驚異的です。

コンテストでは、第 4 回大会まではベクトルプロセッサを中心とした Cray C90 を、第 5 回大会以降は高並列プロセッサ型 Origin2000 を使っています。高並列プロセッサ型に移行した際、ベクトルプロセッサで享受できていた自動並列化の部分が使えなくなり、並列化ライブラリとして OpenMP ではなく MPI を選択したため、それまでのデータ並列からタスク並列にプログラミング手法が変更になり、高校生にとっては並列化のハードルがかなり高くなったはずで、それでも、第 5 回大会では、優勝組はコーディング終了 5 分前に完成した MPI 並列で 32 プロセッサに対し約 30 倍の高速化を実現し、逆転優勝しています。

普段パソコン等でコーディングをしていると、並列化による高速化のイメージは湧かないようで、毎回、参加者のある割合は依然、逐次プログラムの中で高速化を行おうとしていますが、優勝はおろか入賞もできていません。ある程度アルゴリズムが固まったプログラムに関し、逐次上でかなり高速化しても、32 プロセッサあるいは64 プロセッサを使うことで得られる数十倍の性能向上を上回ることはかなり難しいからです。そのことに気づくのは、たいていコンテストが終わってからのようです。

コンテストでは期間中、各組とも相当数の補助ツールを作ります。通常のイメージからすれば、自組で独占すればいいのですが、気楽に他組に公開しています。この辺にも本コンテストの特徴があるようです。競争意識と同時に、自ら作ったものを評価して欲しいという欲求もあるのだと思います。このあたりはプログラミングに興味ある若者に共通するものなのでしょう。

コンテストからみえてくる情報化社会

情報化社会という言葉が使われてずいぶんと久しい。しかし、なぜか、日本の高校ではプログラミング教育が正規の教科としては行われてきませんでした。平成15年度から、教科「情報」という科目がスタートし、情報Bでは問題解決という形で間接的にプログラミング教育が可能となりましたが、肝心の教科書の方ではアルゴリズムやプログラミングに関する記述はほとんど含まれていません。筆者自身も教科書作成の現場にいましたが、高校の先生方のコメントとしては、とてもプログラミングは教えられない、という感想でした。

スポーツと社会、芸術と社会との関係は今ではあたりまえすぎてその相互効果についてあえて議論されることはありませんが、よく考えてみれば、スポーツや芸術が「直接」社会の生産性にかかわっているわけではありません。しかし、スポーツや芸術に関しては裾野を広げることも、トップの人材を養成することも同時に重要と考えられています。しかし、教育に関してはなぜか裾野を広げることは大事でも、トップを育てることは避けてきたようにみえます。そうした中で、プログラミング教育に関しては、裾野もトップもまったく手付かずな状況に日本はあります³⁾。

高校生を対象とした世界規模のコンテストに数学オリンピックや化学オリンピックがあります。それに、情報系の情報オリンピックが加わってきます。日本もかつては情報オリンピックに参加し、優勝したときもありましたが、ここ数年は日本での国内予選は休眠状態になっています。一方アジア各国、たとえば、中国、インド、タイなどでは、数千人から数万人単位の規模でプログラミ

ングコンテストを行い、情報オリンピックへの代表選手を決めています。第10回大会に招待した4カ国の高校生全員が情報オリンピック選考選手でした。

情報化社会の前提として、多くのプログラマーが必要かという質問があったとしたら答えはNOです。社会に大量の数学者が必要であることはなく、数学を教えることで全員を数学者にするわけではないからです。しかし、その一方で、社会に影響を与える優秀な数学者も生み出す可能性も教育は含んでいます。すべてに言えることですが、まずはその分野に興味を持つ若者が増え、そしてその中から世界で通用するトップの人材が現れ、その人たちがまた新たに興味を持つ若者をその世界に引き込んでいきます。しかし、情報に関してはそうしたダイナミックな動きが、ゲームやCGの世界を除いてはほとんど見られません。

我々が始めたコンテストには「**電脳甲子園**」⁴⁾という愛称があります。開催時期が野球の甲子園に近いという意味合いもあるのですが、もう1つは、プログラミングを野球に例えれば、野球ができるグラウンドを提供し、そこで多くの選手の活躍を見てみたいという願いも込められています。確かに彼ら、彼女らは、日本の高校生の中ではことプログラミングに関しては例外的な存在かもしれません。しかし、理由はいろいろとあるのでしょうか。プログラミングに興味を持つ若者たちの裾野が広がっていない現状において、コンテストに参加した彼らのさまざまな分野での活躍が、大きな刺激になっていくはずですが、ちょうどこの原稿を書いている時点で、世界大学生プログラミングコンテストのアジア地区代表(2組)のいずれにも、本コンテストのOBが含まれていることが分かりました。また、過去2回の世界大会にもコンテストOBが参加しています。一方、第1回出場者には大学で研究生生活を始めたものもいます。来年度以降も、形は変わるかもしれませんが^{☆5}、本コンテストは継続していくことになっています。ささやかな試みではありますが、ぜひ、多くの方々の応援がいただければ大変ありがたいと思います。なお、文献1)にコンテストページのアドレスが掲載されています。ご覧いただければ幸いです。

参考文献

- 1) SuperCon ページ: <http://www.gsic.titech.ac.jp/supercon/>
- 2) 第10回スーパーコンピュータプログラミングコンテスト, 東工大クロニクル, No.392 (Oct. 2004).
- 3) 穴戸周夫: プログラミングをしないコンピュータ先進国, 情報処理 Vol.42, No9 (Sep. 2001).
- 4) 松田裕幸, 渡辺 治: 電脳甲子園 (仮題), 日本評論社.
(平成17年2月15日受付)

☆5 2005年度第11回大会から、情報処理学会アルゴリズム研究会、電子情報通信学会コンピュータシミュレーション研究会が協賛に加わります。