**Contest Link:**[http://acm.hust.edu.cn/vjudge/contest/view.action?cid=41260](http://l.facebook.com/l.php?u=http%3A%2F%2Facm.hust.edu.cn%2Fvjudge%2Fcontest%2Fview.action%3Fcid%3D41260&h=qAQGbuGo-&s=1)

**Problem A: UVa 11723**

রাস্তার সংখ্যা R এবং ব্যবহারযোগ্য নাম্বারের সংখ্যা N দেওয়া আছে। N সংখ্যক রাস্তা অক্ষরের সাহায্য ছাড়াই লেখা যায়। সুতরাং (R-N)/N এর সিলিং নিলেই ন্যূনতম কতটা অক্ষর প্রয়োজন হচ্ছে সেটা পাওয়া যাবে।

source code: [http://paste.ubuntu.com/7016285/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016285%2F&h=dAQEWq0Z5&s=1)

complexity: O(1)

**Problem B: UVa 11934**

সাধারণ কমপ্লিট সার্চ প্রবলেম। 0 থেকে L পর্যন্ত লুপ চালিয়ে d দিয়ে কতটা f(n) ডিভিজিবল সেটাই আন্সার।

source code: [http://paste.ubuntu.com/7016309/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016309%2F&h=EAQGQtAEB&s=1)

complexity: O(L)

(আমি কোড করার সময় ইনপুট লিমিট খেয়াল করিনি, তাই ইন্টেজার ওভারফ্লো পরিহারের জন্য মডুলার অ্যারিথমেটিক অ্যাপ্লাই করেছি যেটার দরকার ছিল না। মডুলার অ্যারিথমেটিক খুবই গুরুত্বপূর্ন একটা জিনিস, না জানা থাকলে এখান থেকে দেখে নিতে পারেনঃ <http://www.shafaetsplanet.com/planetcoding/?p=936> )

**Problem C: UVa 10235**

স্টেটমেন্টে যা বলা আছে তা সরাসরি ইমপ্লিমেন্ট করতে হবে। শুরুতে প্রাইম নাম্বারগুলো সিভ অভ ইরাটস্থেনিজ ([http://www.shafaetsplanet.com/planetcoding/?p=624](http://l.facebook.com/l.php?u=http%3A%2F%2Fwww.shafaetsplanet.com%2Fplanetcoding%2F%3Fp%3D624&h=RAQHwdoP6&s=1)) দিয়ে জেনারেট করে রাখলে বাকি কাজ থাকে শুধু সাধারণ একটা চেক।

source code: [http://paste.ubuntu.com/7016348/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016348%2F&h=MAQELpibN&s=1)

complexity: O(n^2) for pre-calculation, O(1) for output.

**Problem D: CodeForces 394B**

এই প্রবলেমে বলা আছে, n-ডিজিটের এমন একটা সংখ্যা প্রিন্ট করতে হবে যার k-গুণ হল ওই সংখ্যার লাস্ট ডিজিটটা সংখ্যাটার ফার্স্ট ডিজিটের বামে নিয়ে আসলে যে সংখ্যা পাওয়া যায়, সেটা। যেমন- ১৪২৮৫৭ সংখ্যাটার শেষ ডিজিট ৭ কে প্রথম ডিজিটের বামে আনলে দাঁড়ায় ৭১৪২৮৫, যা ১৪২৮৫৭ এর ৫ গুণ। যেহেতু সংখ্যাটার ডিজিট ৬টা, সুতরাং (n,k) = (৬,৫) ইনপুটের জন্য আউটপুট ১৪২৮৫৭, যা স্যাম্পলে দেওয়া আছে। মনে রাখা দরকার, আন্সার এবং আন্সার \* k উভয়ই n ডিজিটের হবে।

এখানে শেষ ডিজিট থেকে খুব সহজেই পুরো সংখ্যা জেনারেট করা যায়। শেষ ডিজিটকে k দিয়ে গুণ করলে এককের ঘরে যা পাওয়া যাবে তা হল আন্সার\*k এর শেষ ডিজিট, অর্থাৎ আন্সারের শেষের আগের ডিজিট। এভাবে শেষ ডিজিটের যেকোন একটা ভ্যালুর জন্য n-ডিজিটের পুরো আন্সার জেনারেট করা যায়। শেষ পর্যন্ত জেনারেট করা আন্সারের সবচেয়ে বামের ডিজিটকে k দিয়ে গুণ দিলে যদি এক ডিজিটের নন-জিরো একটা সংখ্যা পাওয়া যায় যা হল আন্সারের ধরে নেওয়া শেষ ডিজিট, তাহলে জেনারেট করা আন্সারটা সঠিক। কাজেই লাস্ট ডিজিটের জায়গায় ১ থেকে ৯ পর্যন্ত বসিয়ে জেনারেটেড আন্সার ভেরিফাই করা যায়।

আরো বলা আছে এভাবে পাওয়া সবচেয়ে ছোট সংখ্যাটা প্রিন্ট করতে হবে। এখন আন্সারের লাস্ট ডিজিট হবে আন্সার\*k এর ফার্স্ট ডিজিট, সুতরাং আন্সারের লাস্ট ডিজিট যত ছোট হবে, আন্সার\*k অর্থাৎ আন্সারও তত ছোট হবে। সুতরাং ১ থেকে ৯ পর্যন্ত চালানো লুপে পাওয়া প্রথম ভেরিফাইড আন্সারই হবে আউটপুট।

source code: [http://paste.ubuntu.com/7016363/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016363%2F&h=AAQFQxsSu&s=1)

complexity: O(9\*n)

**Problem E: UVa 10176**

এই প্রবলেমে একটা বাইনারি সংখ্যা ১৩১০৭১ দিয়ে বিভাজ্য কিনা দেখতে হবে। বাইনারি থেকে আমরা ডেসিমালে কনভার্ট করি এভাবেঃ

(11001) = 1\*2^4 + 1\*2^3 + 0\*2^2 + 0\*2^1 + 1\*2^0

এখানে ইনপুট সর্বোচ্চ ১০০০০ ডিজিট থাকতে পারে। সুতরাং আমরা শুরুতেই 2^0 থেকে 2^10000 পর্যন্ত মডুলো 131071 এর রেজাল্ট স্টোর করে রাখতে পারি। এক্ষেত্রে আমরা মডুলার অ্যারিথমেটিক ও ডাইনামিক প্রোগ্রামিং এর কৌশল ব্যবহার করতে পারি। এরপরে ইনপুটের জন্য (a+b+c)%d = a%d + b%d + c%d মডুলার অ্যারিথমেটিকের এই সূত্র ব্যবহার করে প্রিক্যাল্কুলেটেড ভ্যালুগুলো থেকে ডিভিজিবিলিটি বের করা যায়।

source code: [http://paste.ubuntu.com/7016459/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016459%2F&h=HAQEEnDss&s=1)

complexity: O(n) for pre-calculation, O(length\_of\_bit\_string) for output

**Problem F: CodeForces 376C**

এই প্রবলেমে সবচেয়ে গুরুত্বপূর্ণ পয়েন্ট হল ইনপুট সংখ্যায় সবসময় ১,৬,৮ এবং ৯ থাকবে। এখন একটু ঘাঁটাঘাটি করলেই দেখা যায় ১৬৮৯ এর সবগুলো পারমিউটেশনে মডুলো ৭ এর সবগুলো ভ্যালুই পাওয়া যায় (০,১,২,৩,৪,৫,৬)। একটা ইনপুট নিয়ে দেখা যাক।

ইনপুটঃ ১২৩৪৫৬৭৮৯০

রি-অ্যারেঞ্জ করেঃ ২৩৪৫৭০১৬৮৯ = ২৩৪৫৭০০০০০ + ১৬৮৯

কাজেই, (২৩৪৫৭০১৬৮৯)%৭ = ((২৩৪৫৭০০০০০)%৭ + (১৬৮৯)%৭)%৭

সংখ্যাটাকে ৭ দিয়ে ডিভিজিবল করতে আমরা ২৩৪...০০০ কে ৭ দিয়ে মড করলে যদি x পাই, তাহলে ১৬৮৯-কে এমনভাবে রি-অ্যারেঞ্জ করতে হবে যেন ৭ দিয়ে মড করে (৭-x) পাওয়া যায়। তাহলে, (৭+৭-x)%৭ = ০ অর্থাৎ সংখ্যাটা ৭ দিয়ে ডিভিজিবল হয়ে যাচ্ছে। যদি রিঅ্যারেঞ্জ করার পর লিডিং জিরো থাকে সেগুলোকে ১৬৮৯ এর পারমিউটেশনের শেষে বসাতে হবে।

source code: [http://paste.ubuntu.com/7016622/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016622%2F&h=ZAQG23Kv_&s=1)

complexity: O(length\_of\_number) for rearrangement, O(4!) for permuting 1689

**Problem G: UVa 10940**

এই প্রবলেম সিমুলেট করলে একটা সুন্দর প্যাটার্ন পাওয়া যায়। সেই প্যাটার্ন প্রিক্যালকুলেট করে রাখলে সহজেই O(1) এ আউটপুট দেওয়া যায়।

source code: [http://paste.ubuntu.com/7016647/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016647%2F&h=jAQHrQ88H&s=1)

complexity: O(n) for pre-calculation, O(1) for output.

**Problem H: CodeForces 322B**

এই প্রবলেমের জন্য প্রত্যেক রং-এর ফুল সংখ্যার এক কম নিয়ে ৩ দিয়ে ভাগ দিলে আমরা একই রঙ এর ৩টা করে নিয়ে কতটা বুকে বানানো যায় তা পাচ্ছি। শেষে যে কয়টা রঙ এর ফুল অবশিষ্ট থাকে সেগুলো দিয়ে বেস্ট সলুশন বের করতে হবে। এটা এই কনটেস্টের সবচেয়ে ট্রিকি প্রবলেম ছিল।

source code: [http://paste.ubuntu.com/7016678/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016678%2F&h=NAQFrto7Z&s=1)

complexity: O(1)

**Problem I: UVa 11847**

প্রতিদিন আগের দিনগুলোতে দেওয়া পিসগুলোর দৈর্ঘ্যের যোগফল থেকে এক একক লম্বা পিস দিলেই অপটিমাল হচ্ছে অর্থাৎ সবচেয়ে কম সংখ্যকবার কাটা লাগবে। কাজেই ১,২,৪,৮..... এই সিরিজের যোগফল n এর চেয়ে বড় হওয়ার ঠিক আগ পর্যন্ত যতটা এলিমেন্ট পাওয়া যায় সেটাই হবে অপ্টিমাল কাটের সংখ্যা।

source code: [http://paste.ubuntu.com/7016683/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016683%2F&h=1AQEcCopc&s=1)

complexity: O(r) where r-th term is the furthest the series can go before the cumulative sum exceeds n

**Problem J: CodeForces 371B**

এখানে সহজেই বোঝা যায় যে সবচেয়ে তাড়াতাড়ি যে ওয়েইটে টুকরা দুটো সমান হবে তা হল তাদের গসাগু। সুতরাং দুটো টুকরোকে তাদের গসাগু দিয়ে ভাগ দিলে যে দুটো সংখ্যা পাওয়া যায়, তাদের ডিভিজগুলোর মধ্যে যতগুলা ২,৩,৫ আছে ততগুলাই ভাগ করতে হবে। যদি ২,৩,৫ ছাড়া আর কিছু থাকে সেক্ষেত্রে ভাগ করা সম্ভব না।

source code: [http://paste.ubuntu.com/7016750/](http://l.facebook.com/l.php?u=http%3A%2F%2Fpaste.ubuntu.com%2F7016750%2F&h=0AQHa2wpg&s=1)

complexity: O(prime\_factors\_of\_a\_and\_b)

**Problem K: UVa 10407**

এই প্রবলেম কিছু সংখ্যা দেওয়া থাকবে। এমন একটা সংখ্যা বের করতে হবে যাতে ওই সংখ্যা দিয়ে প্রত্যেক সংখ্যাকে ভাগ করলে একই ভাগশেষ পাওয়া যায়।

ধরা যাক দুটো সংখ্যা a এবং b দেওয়া আছে, তাদেরকে x দিয়ে ভাগ করলে d ভাগশেষ থাকে। তাহলে,

a % x = d

b % x = d

(a-b) % x = a%x - b%x = d - d = 0

অর্থাৎ দুটো সংখ্যার ডিফারেন্স x দিয়ে বিভাজ্য হবে।

কাজেই n সংখ্যক সংখ্যার ক্ষেত্রে পরপর দুটো সংখ্যার যে n-1 সংখ্যক ডিফারেন্স পাব, তাদের সবাই আমাদের আন্সার দিয়ে ডিভিজিবল হবে। যেহেতু এরকম সবচেয়ে বড় সংখ্যা চাওয়া হয়েছে, সুতরাং তাদের গসাগুই হবে আউটপুট।

source code: <http://paste.ubuntu.com/7016780/>

complexity: O(n\*a/b)

**কারো কোন জিজ্ঞাসা থাকলে বা কোন প্রবলেমের বিকল্প সমাধান জানা থাকলে কমেন্টে জানানোর অনুরোধ রইলো। আর এডিটোরিয়ালে কোন ভুল থাকলে অবশ্যই জানাবেন।**

**হ্যাপি কোডিং! :)**