BROCADE

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Helvetic Coding Contest

Lausanne, March 13th 2010

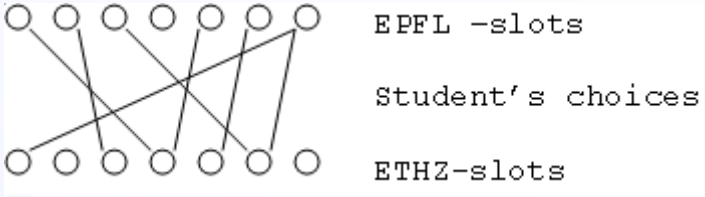
solutions & results

	# Submissions	Best score
A	9	2
B	16	4
C	0	0
D	26	10
E	6	10
F	3	6
G	25	10
H	26	9
I	10	10

Always there for you

- # Submissions : 9
- Best Score : 2
- First best team : **YAWN**

- Represent problem as a graph



- Find minimum vertex cover
- Graph is bipartite
- König: min. vertex cover equals max. matching in bipartite graphs
- Hopcroft-Karp ($O(N^{5/2})$) or Ford-Fulkerson ($O(N^3)$)
- Hungarian forests of alternating trees

Book of secret stories

- # Submissions : 16
- Best Score : 4
- Best team : **ASH**

- Perform frequency analysis of mono-, bi- and trigrams in any language
- Use Enderlein and Kauth as reference texts to decode Verne, Kafka, Shakespeare, Alighieri and co. ☺
- Then you need a bit of creativity and you solve test-case by test-case until…
- you reach test-case 10: no 'E' in the text, to perturb frequency analysis

Carving around the marmots

- # Submissions : 0 ☹
- # Solved inputs : 0

- Use recursion and backtracking to generate all possible villages and evaluate all paths on them (solves 40%)
- These optimizations lead to 100% :
  - Bit-masking
  - Speed-up village generation considerably by keeping track of maximum allowable offset of next group in each row and column
  - Look for collisions on the maps already during construction of paths (cut-off)
  - Maps on which part of path collided already can be ignored down the sub-tree of path generation
- Overall speed-up > 1s/24h

Delivery

- # Submissions : 26
- Best Score : 10
- First 'accepted' team : **YAWN**

- Every move of fondue can be decomposed into a sequence of several moves between adjacent dining locations.
- So greedy works!
- Just transport the excessive or virtually missing portions from dining place 1 to dining place 2, then from dining place 2 to dining place 3 and so on.
- Do simultaneously sum the moved portions (absolutely!).
- Solutions without 64-bit integers score 60%

Exploring Switzerland

- # Submissions : 6
- Best Score : 10
- Only 'accepted' team :

VIS IV Ballmer Peak

- Sort locations alphabetically
- For any pair of distinct locations i and j, store the maximum number of locations along a path that starts at i, moves up the list to the starting location and then down the list to j
- From such a path we can dynamically build a path that starts or ends further down in the list.
- Find longest such path
- Reverse the list and redo the job to cover both types of paths (firstDownThenUp, firstUpThenDown)

Finest selection

- # Submissions : 3
- Best Score : 6
- Only 'quasi-accepted' team :
  VIS IV Ballmer Peak

- Consider a directed graph whose nodes are the pairs <*player*,*Plates*> where *player* indicates who is next to move and *Plates* is any possible game constellation (disregarding from the pieces of chocolate that have already been eaten).
- There is a directed edge between two nodes if the players in the two nodes differ and the constellation in the second node can be obtained from the constellation in the first node by a legal move.
- As the number of pieces of chocolate decreases at each turn (at least one piece is eaten), the graph is acyclic.
- Run min-max (e.g. negamax) on that graph
- Use memorization to solve 100%

Good pictures

- # Submissions : 25
- Best Score : 10
- Only 'accepted' team : **YAWN**

- This problem can be brute forced!
- Order all the parameter values by the preferences as indicated in the problem statement and then visit all possible combinations sequentially until you find one that exposes the picture correctly.
- Keep track of whether you've encountered underexposed pictures, overexposed pictures or both

Heidi @ HC²

- # Submissions : 26
- Best Score : 9
- Only 'quasi-accepted' team :
  VIS I – We take no prisoners

- No bounds on scores known a priori
  => stopping rule strategy
- Accumulate knowledge by interviewing candidates and rejecting them. How many?
- *Rule 1* : never accept candidate with score lower than any previous candidate!
- *Def.* Hopefuls satisfy rule 1
- Strategy : @each interview :
  - Is candidate a hopeful one?
  - If so, compare
    probability[winning by accepting] to
    probability[winning by rejecting]
- *Def.* STRAT(s) : reject first s, then accept first hopeful

- What is P(WIN by STRAT(s))?
  - *Hyp:* highest score at position k.
  - if k<=s, P=0
  - So $P(\text{WIN by STRAT(s)})$
    $= \sum_{k=s+1}^{N} P(\text{WIN by STRAT(s)} \cap \text{maximum is at k})$
    $= \sum_{k=s+1}^{N} P(\text{WIN by STRAT(s)} \mid \text{maximum is at k}) \cdot P(\text{maximum is at k})$
  - Random order implies $P(\text{maximum is at k}) = \frac{1}{N}$
  - For the other P, we have to ensure that candidate k is the first hopeful after s. This happens only if the maximum of the first k-1 candidates lies within the first s, which occurs with probability s/(k-1)
  - Finally $P(WIN\ by\ STRAT(s)) = \frac{s}{N} \sum_{k=s+1}^{N} \frac{1}{k-1}$

- What s* maximizes P(WIN by STRAT(s)) for given N?
  - find s* by DP in linear time
  - or in sub-linear time by ODDS-algorithm

Ice Pyramids

- # Submissions : 10
- Best Score : 10
- First 'accepted' team : **Ciresarii**

- 1 geometric problem, so many approaches:
  - analytical solution (100%)
  - descent methods (in 2D) (100%)
  - constrained non-linear optimization (Constrained Newton, Interior point, Augmented Lagrangian, Sequential quadratic programming, …) (100%)
  - ternary search (70%)
  - some intuition ☺

- *Analytical solution:*
  - Cosine law gives vertex set for base
  - Volume = 1/3 * Area[base] * height, gives h
  - *Def.* P top, Q it's projection on base
  - *Def.* u1..u3 distance from Q to sides a1..a3

- *objective F:*

$$\text{Area}(u1,u2,u3) = \text{Area}[base] + \frac{a1\cdot\sqrt{u1^2+h^2}}{2} + \frac{a2\cdot\sqrt{u2^2+h^2}}{2} + \frac{a3\cdot\sqrt{u3^2+h^2}}{2}$$

- *constraint G:*

$$\text{Area}[base] = \frac{a1\cdot u1}{2} + \frac{a2\cdot u2}{2} + \frac{a3\cdot u3}{2}$$

- Use Lagrange multiplier to minimize *F* under *G*

$$\begin{cases} \frac{\partial F}{\partial u1} = L\cdot\frac{\partial G}{\partial u1} \Rightarrow \frac{u1}{\sqrt{u1^2+h^2}} = L \\ \frac{\partial F}{\partial u2} = L\cdot\frac{\partial G}{\partial u2} \Rightarrow \frac{u2}{\sqrt{u2^2+h^2}} = L \\ \frac{\partial F}{\partial u3} = L\cdot\frac{\partial G}{\partial u3} \Rightarrow \frac{u3}{\sqrt{u3^2+h^2}} = L \end{cases}$$

This means u1=u2=u3
=> Q is the incenter of the base

- So all that's left is
  - computing the intersection of 2 of the angle bisectors of the base (geometrically or by binary-search)
  - or observe that the radius of the incircle equals r = 2*Area[base]/(a1+a2+a3)
  - then use Pythagoras for the total area of pyramid

```
// very elegant solution & implementation by Ciresarii
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main(void)
{
        double a, b, c, v, S, p, r, h, A;
        double eps=0.000001;
        scanf("%lf %lf %lf %lf", &a, &b, &c, &v);
        while(abs(v) > eps)
        {
                p = (a+b+c)/2;
                S = sqrt(p*(p-a)*(p-b)*(p-c));
                r = S / p;
                h = 3*v / S;
                A = S + sqrt(h*h + r*r) * p;
                printf("%lf\n", A);
                scanf("%lf %lf %lf %lf", &a, &b, &c, &v);
        }
        return 0;
}
```

15

Zeptox

Hepia

Donoreaz Thomas
Lawrence David
Racordon Dimitri

1 : 0

14

soi-jsl

Swiss Olympiad in Informatics

Grütter Sämi
Todorović Lazar
Ziegler Josef

7 : -6

13

Les champomy

HEIG-VD

Brönnimann Florian
Lala Alain
Steiner Pierre

7 : -3

12

SOI 1

Swiss Olympiad in Informatics

Aulbach Adrian
Balicka Sofia

10 : -4

11

SOI 2

Swiss Olympiad in Informatics

Jehli Martin
Kayed Alexander

11 : -3

10

YAPT

Ecole Polytechnique Fédérale de Lausanne

Bonvin Philippe
Edelmann Romain
Raykov Pavel

12 : -6

9

The Wiggles

Université de Genève

Barbieri Bruno
Sartoretti Guillaume
Schlechten Jonathan

12 : -2

8

Modulmänner

FHNW Windisch

Suter Daniel
Uhlmann Patrick
Wettstein Claudio

14 : -14

7

VIS II
Visbjoern Strikes Back

ETH Zurich

Deutsch Isaac
Reiter Christian
Roos Adrian

18 : -6

6

Ciresarii

Ecole Polytechnique Fédérale de Lausanne

Stoianov George
Salajan Dan

20 : -14

5

ASH

Ecole Polytechnique Fédérale de Lausanne

Afshari Hossein
Amini Arash

20 : -7

4

VIS III
Codehängscht

ETH Zurich

Humbel Lukas
Manser Lukas
Stucki Yannick

20 : -1

Bronze medal

VIS IV - Ballmer Peak

ETH Zurich

Bruggmann Marc
Helbling Christian
Krapf Lars

Overview
Solutions
Ranking
Top 3
Evening Program

Silver medal

YAWN

Ecole Polytechnique Fédérale de Lausanne

Cieslewski Titus
Steiger Robin
Upadhyay Utkarsh

Gold medal

VIS I - We Take No Prisoners

ETH Zurich

Feuz Sandro
Gelashvili Rati
Serbinenko Vladimir

Switzerland's best coders

Final Ranking

#	Country	Team	Problems									Solved	Points
			A	B	C	D	E	F	G	H	I		
1	ETH	ETHZ VIS I - We Take No Prisoners		3 (-1:2)		10 (0)			5 (-3:6)	9 (-18:8)	10 (-5)	37	-27
2	EPFL	EPFL YAWN	2 (-2:2)	2 (0:1)		10 (0)	0 (0:4)		10 (0)	2 (0:1)	7 (0:4)	33	-2
3	ETH	ETHZ VIS IV - Ballmer Peak		2 (0:1)		10 (-12)	10 (0)	6 (0:2)	3 (0:1)	2 (0:1)		33	-12
4	ETH	ETHZ VIS III - Codehängscht		3 (-1:2)		10 (0)	3 (0:1)		2 (0:2)	2 (0:1)		20	-1
5	EPFL	EPFL ASH	0 (0:1)	4 (0:1)		6 (0:1)					10 (-7)	20	-7
6	EPFL	EPFL Ciresarii				10 (-14)					10 (0)	20	-14
7	ETH	ETHZ VIS II - Visbjoern Strikes Back	1 (-2:3)	2 (0:1)		10 (-4)			3 (0:5)	2 (0:4)		18	-6
8	FHNW	FHNW Modulmänner	0 (0:1)	2 (0:1)		10 (-14)			0 (0:5)	2 (0:4)		14	-14
9	UNIGE	UNIGE The Wiggles		2 (0:1)		6 (0:1)			4 (-2:3)			12	-2
10	EPFL	EPFL YAPT		2 (-2:2)		10 (-4)		0 (0:1)				12	-6
11	SOI	SOI 2	0 (0:2)	2 (0:1)		6 (0:1)				3 (-3:7)	0 (0:1)	11	-3
12	SOI	SOI 1				10 (-4)						10	-4
13	Hes	HEIG-VD Les champomy		2 (0:1)		5 (-3:2)						7	-3
14	SOI	SOI-jsl		1 (0:1)		6 (-6:2)						7	-6
15	Hes	Hepia Zeptox		1 (0:1)					0 (0:2)			1	0

Feedback

- Reception
- Daily routine
- Presentations
- Contest
- Who would come back in 2011 ?

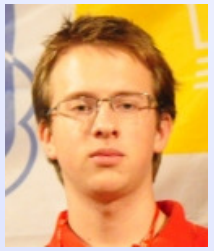
The organizers

Christian Kauth
Problems & Coordination

Jonas Wagner
Infrastructure & Graphical Design

Pierluca Borso
Rules & Communication

Robert Enderlein
Judging & Accounting



Titus Cieslewski
Website & Registration

Jean-Paul Wenger
Sponsoring

BROCADE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

We enjoyed having you at HC²

PolyProg
Organizer

BROCADE
Event Sponsor

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
Contest Host