



ACM ICPC 2019 国内予選問題 解説

平石 裕実

(京都産業大学 情報理工学部)



プログラムのソース・サイズ・実行時間や必要な知識等一覧

問題	ソース	行数	実行時間(秒)	解法や知識・テクニック等
A. 期末試験の成績	pa.c	26	0.182(0.201)	一次元配列、合計、最大値
B. スクリーンキーボード	pb.c	49	0.331(0.196)	二次元配列、文字列、マンハッタン距離
C. 天秤	pc.c	97	1.034(0.739)	組合せ生成、深さ優先探索、ソート、2分探索
D. 計数カウンタ	pd.c	47	1.189(0.418)	動的計画法
E. 立方体表面パズル	pe.c	211	0.201(0.278)	空間思考力、深さ優先探索、枝刈り
F. 色の切り替え	pf.c	100	86.734(27.758)	全域木、Union Find、深さ優先探索
G. タイルを動かそう！	工事中			
H. 凸多角形の加法				

- 行数は、コメントや空行やデバッグ用コードを除いた行数
- 実行時間(秒): コンパイルやサンプルデータとジャッジデータに対する判定時間を含む (括弧内は-O2でコンパイル)
Core i7-8559U @ 2.70GHz, 16GBメモリ, MacOS 10.15.7, Apple clang version 12.0.0 (clang-1200.0.32.0)



問題A 期末試験の成績



- 各生徒の合計点を求めるための一次元配列 `sum[1000]` を用いる（生徒の人数の最大値は1000）。
- 最初に、一次元配列 `sum[0]~sum[n-1]` を0に初期化する。 // `n = 生徒数`
- 各科目 `i` に対して
 - ▶ 各生徒 `j` の得点を読み込み `sum[j]` に加算する。
- `sum[j]` は生徒 `j` の合計点となるので、その最大値を求めれば良い。
- 時間計算量 = $O(m*n)$ // `m = 科目数`、`n = 生徒数`

問題B スクリーンキーボード



A B C D E F G H



- スクリーンキーボードは char型の2次元配列に格納。
- スクリーンキーボード上で、あるキー k_1 の次に別のキー k_2 を入力するためには、 k_1 と k_2 のマンハッタン距離+1回ボタンを押す必要がある（マンハッタン距離の回数だけ矢印キーを押し、最後に「OK」キーを押す）。（問題の図B-3参照）
- k_1, k_2 のスクリーンキーボード上の位置（2次元配列上の位置）を各々 r_1 行 c_1 列, r_2 行 c_2 列 とすると、マンハッタン距離は $|r_1 - r_2| + |c_1 - c_2|$ である。
- 最初のカーソル位置を0行0列とし、入力したい文字列の先頭の文字から順にボタンを押す回数を計算し、その合計を求めれば良い。
- 入力したいキーのスクリーンキーボード上の位置を求めるのに2次元配列を単純にスキャンする場合、時間計算量は $O(h * w * |s|)$ となる（ h, w はスクリーンキーボードの高さと幅、 $|s|$ は入力したい文字列の長さ）。 h, w の最大値は 50, $|s|$ の最大値は 1000 なので、 $h * w * |s| = 2.5 * 10^6$ であり問題ない。
- データセットの入力
 - 一行目の $h w$ は int 型で入力し、2行目以降は文字列として入力するならば、 h, w は「scanf(“%d%d\n”, &h, &w);」で入力し行末の「改行」を読み飛ばすと良い。
 - 文字列の入力に「gets」を用いると、環境によっては警告が表示されるので、「fgets」を用いてstdinから入力すれば良い。この場合、入力した文字列の最後に「改行」と「NULL」が付加される点に注意。

問題C 天秤



- 解説は工事中



問題D 計数カウンタ



- 解説は工事中

問題E 立体表面パズル



A B C D E F G H



- 解説工事中

問題F 色の切り替え



A B C D E F G H



- 解説工事中

問題G タイルを動かそう



A B C D E F G H



- プログラム・解説工事中

問題H 凸多角形の加法



A B C D E F G H



- プログラム・解説工事中