# Lifeguard in the Pool

問題原案: 牟田
解説・データ作成: 牟田
解いた人:平野, 泉

# 解答状況

- Submit チーム数:1
- 正解チーム数:0

- やはり，最難問問題の一つと予想されていただけに厳しかったようです.
しかし書くべきコードはそんなに大変ではないので本番で幾何問題が解けそうな場合は是非チャレンジしてみてください.

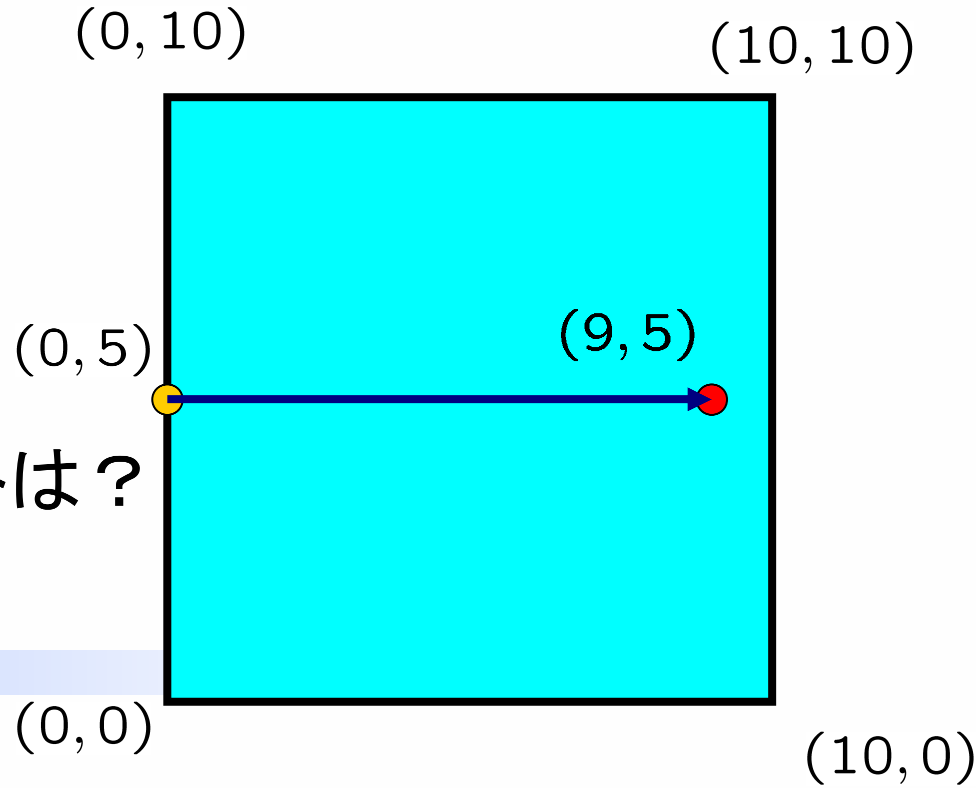
# まず，入力を検討せよ！
# 話はそれからだ！

- Sample Input に問題の重要なヒントが含まれていることがあります.
- 一度図に描いてみるくらいの気持ちでよく検討すること！
- もちろん，Sample に現れない陰険な例もしっかり考えて作っておくこと

# Sample Input/Output(1)

- プールの座標は (0, 0), (10, 0), (10, 10), (0, 10)
- 地上のコスト 10
- 水中のコスト 12
- 監視員は (0, 5)
- 少女は (9, 5)

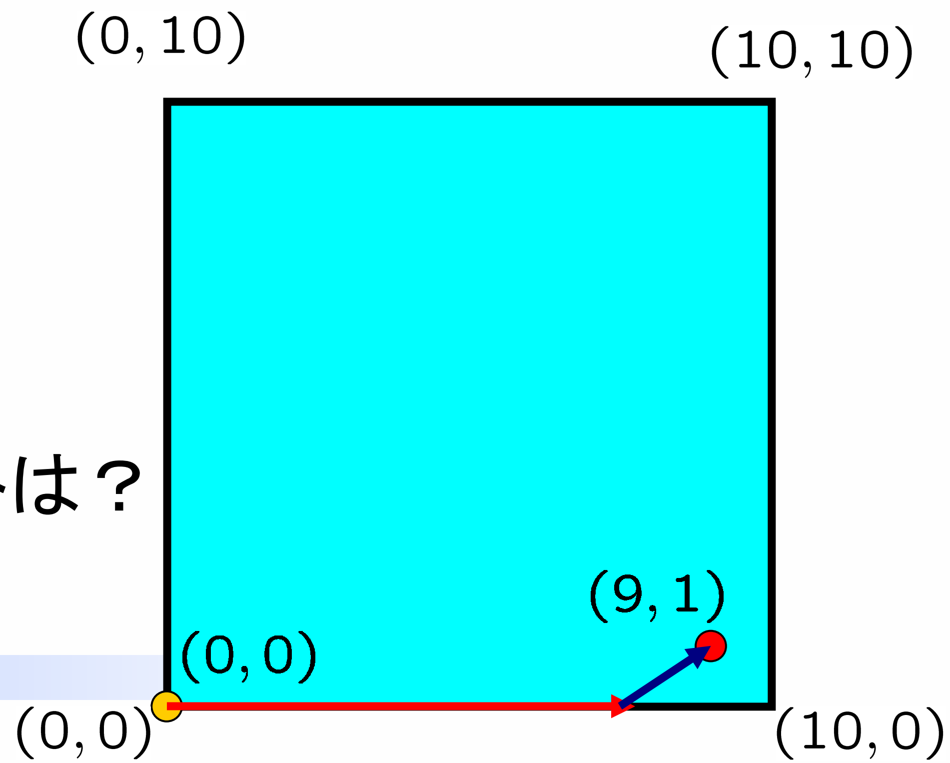
- 最適値 108.0 の経路は？
  水中のみの直線



# Sample Input/Output(2)

- プールの座標は (0, 0), (10, 0), (10, 10), (0, 10)
- 地上のコスト 10
- 水中のコスト 12
- 監視員は (0, 0)
- 少女は (9, 1)

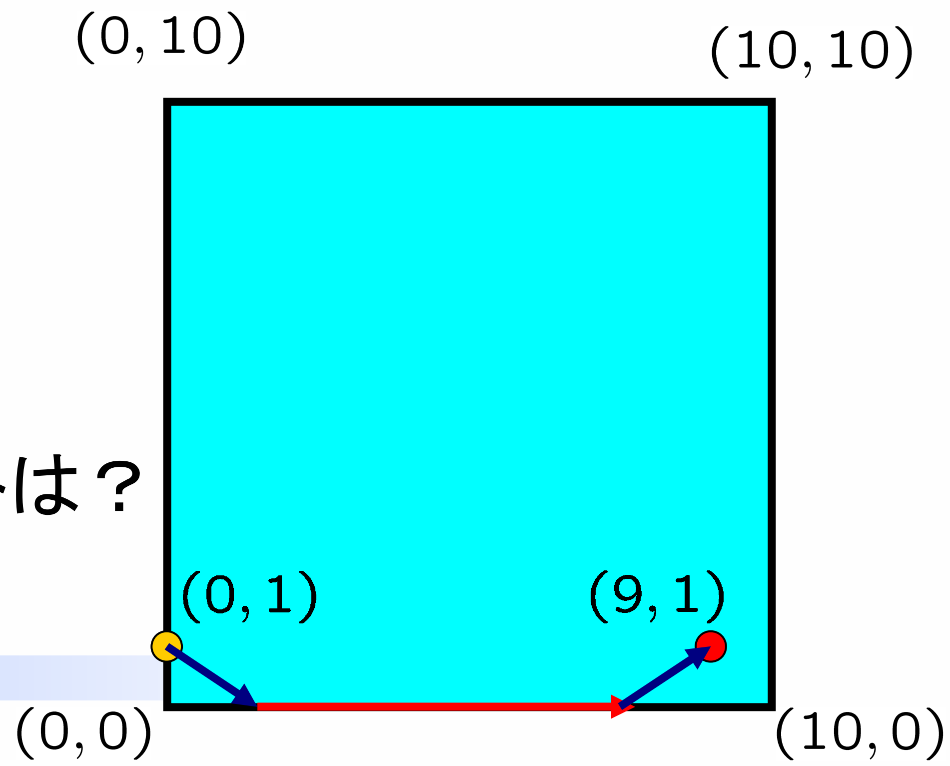
- 最適値 96.63 の経路は？
  陸上→水中の折れ線



# Sample Input/Output(3)

- プールの座標は (0, 0), (10, 0), (10, 10), (0, 10)
- 地上のコスト 10
- 水中のコスト 12
- 監視員は (0, 1)
- 少女は (9, 1)

- 最適値 103.3 の経路は？
  水→陸→水の折れ線



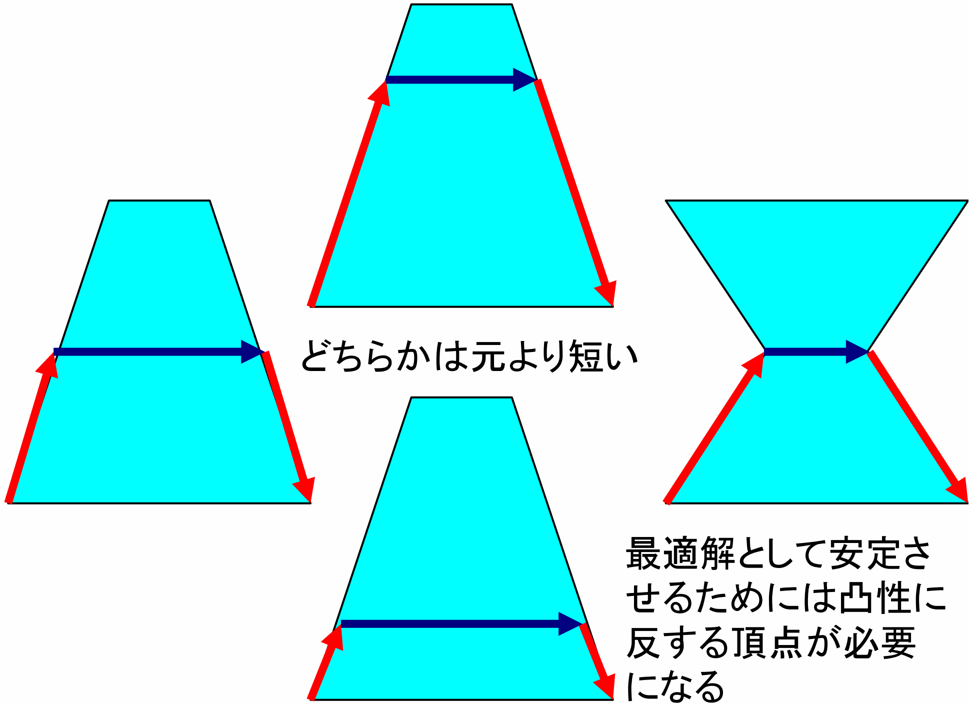
# Sample Input/Output まとめ

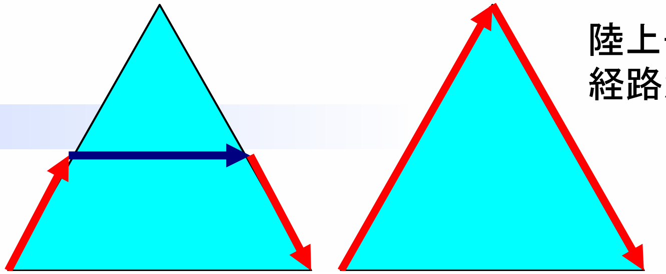
- 最適経路は少なくとも3種類ある
  - 直線的に泳ぐ
  - 陸上を移動してある位置から飛び込む
  - 泳いで，上陸して陸上を走り，又飛び込む

- 残り考えることは
  - 別の最適経路パターンを考慮する必要があるか？
    「陸上→水中→陸上→水中」等
  - どこから飛び込む？

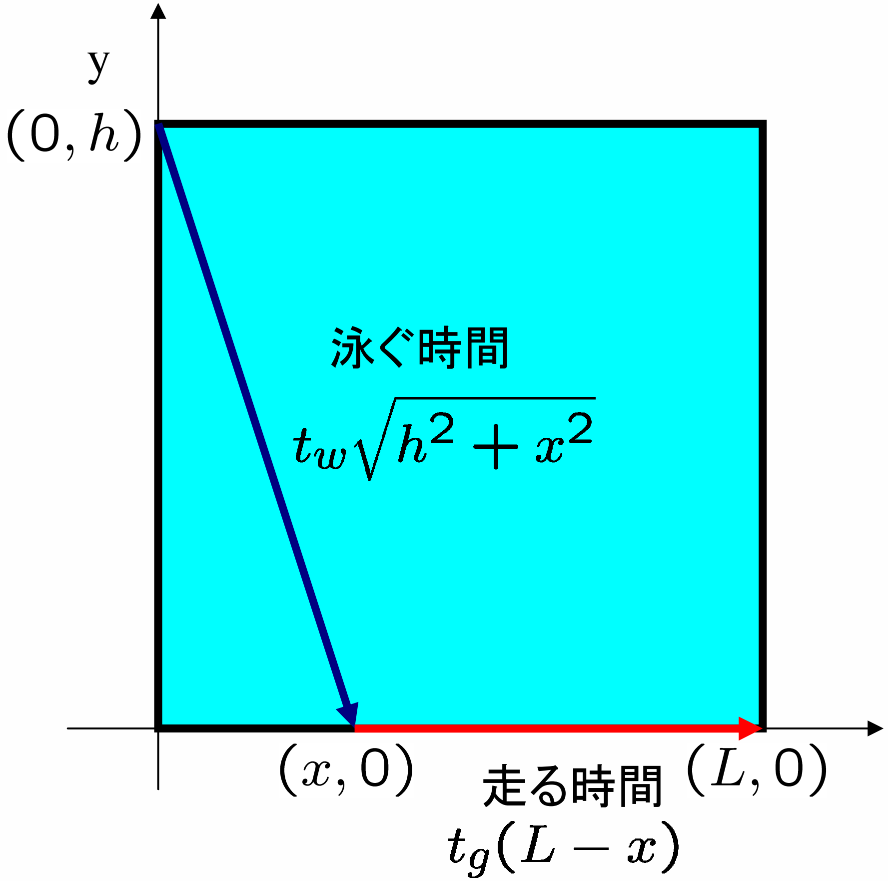
# 別の最適経路パターンを考慮する必要があるか？概略

- 「陸上→水中→陸上」という部分経路を持つ最適経路を仮定しても，プールが凸多角形であることを利用すると
  - 水中移動部分を上下どちらかに動かすことで，より所要時間の短い経路が作ることが出来る
  - 同じ所要時間で「陸上→水中→陸上」が「陸上」だけですむような最適経路が作れたりする
- 結局，「水中→陸上→水中」という経路だけを考えれば，最適解を求めることが出来る



# 飛び込み地点の推定




(0, h) からプールに飛び込み（青→），

(x, 0) で浮上してプールの淵を走り（赤→），

(L, 0) に至る最短時間を考える

浮上地点のx 座標を変数として総計時間を関数にすると

$$f(x) = t_w\sqrt{h^2+x^2} + t_g(L-x)$$

変数 x で微分すると

$$f'(x) = t_w\frac{x}{\sqrt{h^2+x^2}} - t_g$$

| $x$ | | $\frac{ht_g}{\sqrt{t_w^2-t_g^2}}$ | | |
|---|---|---|---|---|
| $f'(x)$ | $-$ | 0 | $+$ | |
| $f(x)$ | ↘ | 極小 | ↗ | |

上陸する辺に垂線を下ろし，その足より $\frac{t_g}{\sqrt{t_w^2-t_g^2}}$ だけずらした地点から上陸するのが最適

# 解法

- 3度泳ぐことは無いと示されたので，「水→陸→水」の経路のみを考えれば十分
  - 「陸→水」の経路は最初の水中移動時間0として処理
  - 上陸する辺と再飛び込みする辺を選ぶ
  - 上陸ポイントと再飛び込みポイントを前スライドに基づき計算
  - 所要時間を計算，最小なものを探す
- 最後に直線的に向かう場合の所要時間を計算

# 幾何について

- 今回は要求精度が $10^{-8}$ と厳しかったので近似的解法は厳しいです．精度を見つつどのアルゴリズムなら解けるかを考えましょう
- 今回の解説にあったように，幾何の解法の正当性を確かめるには証明が必要な場合があります
  - どうせキーボードは一つなのだから手が空いているメンバーがアルゴリズムが正しいかどうかを考えておくと解答速度が大きく向上します．うまく役割分担をしましょう
  - 完全な証明は大変ですし，採点の対象ではないので，8割くらい確信したら，ダメ元でチャレンジしてもいい場面もあるかもしれません．

# ライブラリについて

- 下記のような計算ができないと実装は大変かと思います．要チェックです
  - 回転
  - 直線同士の交点計算
- 幾何の問題では，ライブラリのあるなしが大きく実装能力に影響します．
- 高性能，高信頼な幾何ライブラリの作成はお勧めです

それでは，本番頑張ってください