

---

# プログラミング演習2 (1)

## イントロダクション

---

中村, 高橋, 小林, 橋本

# まずは

---

- 自己紹介を

# プログラミング演習2について

---

- プログラミング演習1と同じところ
  - Processingをやります
- プログラミング演習1との違い
  - 予習より復習を重視して下さい
  - 中間試験があります（一応今のところ）
  - 宿題があります
  - グループ制作 / 発表してもらいます

# 何故，試験を？

---

- プログラミングは基本的に，想定通りに動かすことができるが良い
- 研究室に所属したり，企業に就職してプログラムと関わる場合は，知識も必要
  - 人のプログラムを読み理解することができる
  - プログラミングにまつわる言葉を理解する
  - 頭のなかでプログラミングする
- FMSの学生さんには基本情報技術者試験（国家資格）を軽くクリアして欲しい
  - 基本情報技術者試験にはプログラミングがある

- 情報技術全般に関する基本的な知識・技能をもつ者（情報システム開発プロジェクトにおいて、プログラム設計書を作成し、プログラムの開発を行い、単体テストまでの一連のプロセスを担当しているか、将来、そのような業務を担当する者を含む）

# 情報処理技術者試験

---

- 基本情報技術者試験
- 応用情報技術者試験
- 高度試験
  - ITストラテジスト
  - システムアーキテクト
  - ネットワークスペシャリスト
  - 情報セキュリティスペシャリスト
  - ITサービスマネージャ

# グループ制作 / 発表

---

- 研究室をベースとしてグループ分け
  - 中間試験の点数を基準にグループ分け予定
- 他者とプログラムする事の難しさ, 楽しさを知り, 1人ではできない事を実現する!

# 講義の実施形態

---

- 序盤の「クラス（オブジェクト指向）」などの座学を中心とした内容についてはオンラインで実施
  - Remo Conferenceを利用
  - askTAを利用して質問応答（また案内します）
- 中間試験は対面で実施予定（COVID-19の状況次第で実施方法を検討予定）
  - 11月上旬を予定していますが、COVID-19の状況次第で変更するかもしれませんし、他の手立てを検討するかもしれません
- 11～12月の応用実習系はハイブリッドで実施予定
  - 対面 + Zoom + Discord
  - ネットワーク、ファイル入出力、XMLやWebAPI、フィジカルコンピューティングなど
- 1月に発表会
  - グループ制作が無理な場合は個人発表またはその他の方法に切り替えるかもしれません



# 講義の流れ（あくまで目安）

---

- 1コマ目：
  - 0～20：宿題の解説
  - 20～50：今日のトピックの解説＋課題の提示
  - 50～100：今日の課題に取り組む（要提出）
- 2コマ目：
  - 0～50：今日の課題に取り組む
  - 50～100：今日の課題の解説（答え合わせ）
  - 次回までの宿題の提示

# 評価

---

- 宿題： 30点
- 授業中課題： 20点（要提出）
- 中間テスト： 30点
- 発表会： 20点
  
- 注意
  - 5回休み = 単位不認定
  - 課題は演習中にチェックします

# 要注意！

---

- 他者のプログラムをコピーしない＆コピーさせないこと
- コピーなどの不正行為が発覚した場合、その学期の全ての科目登録が取り消され、懲戒処分の対象ともなります
- コピーさせた人も同様の処分になることに注意してください！

# 講義予定（案）

---

- 第01回 ガイダンス + 前期の復習
- 第02回 クラス (1)
- 第03回 クラス (2)
- 第04回 継承
- 第05回 ArrayList
- 第06回 中間テスト
- 第07回 振り返りとファイル入出力
- 第08回 ネットワーク
- 第09回 Web API
- 第10回 フィジカルコンピューティング
- 第11回 フィジカルコンピューティング
- 第12回 フィジカルコンピューティング
- 第13回 発表会？
- 第14回 発表会（予備日）

# 本日の流れ

---

- 現役組
  - ガイダンス & 宿題提示
  - check ( 出席確認 ) 申請
  - Discordの利用テスト
  - remoをログアウトし、Discordでプレゼン大会
    - 最初研究室グループ、その後は指定のグループで実施
    - 随時、テキストチャットでアナウンスします
  - 時間が来たらテキストチャットに案内出しますので解散
- 再履修組
  - ガイダンス & 宿題提示
  - check ( 出席確認 ) 申請
  - Discordの利用テスト
  - Remo Conferenceで課題提示
  - 課題の提出テスト
  - 課題確認できたら解散

# 再履修の学生さん向け

---

- typing.run というサービスを作りました。こちらをログインして使うと、記録が残る仕組みになっています。
- タイピング速度を向上させつつ、基本的なところを抑えることができるので取り組んでみると良いと思います

# 宿題1-1: hw\_boundA11

---

- 5個の○と, 4個のxと, 3個の△が画面内を動き回るプログラムを作成せよ
  - ただし, その速度はx、y方向それぞれ-5~5の実数値とせよ
  - また, ○は壁で跳ね返り, xと△は跳ね返らずに反対側から出てくるようにせよ

# 宿題1-2: hw\_abababab

---

- aとbそれぞれに1から9までの数字を当てはめたときにできる8桁の数字について、約数の数が最小になるaとbの組み合わせと約数の数をプログラムで求め標準出力せよ
- ただし、aとbが同じ数字でも良い
  - 例： a=3, b=5のときは、35353535の約数の数を求めることになる
- 出典：数学検定 準1級の試験より
  - 大学受験数学で解けますので、数学好きな人はチャレンジしてみたら良いですよ！



# 宿題1-3: hw\_changeChara

---

- マウスカーソルの下に何らかのキャラクタを描くプログラム `hw_changeChara` を、`displayActive`、`displayInactive`を完成させることで作成せよ
  - `hw_changeChara.zip` を利用せよ
  - ただし、それぞれの関数内では`mouseX`、`mouseY`を利用しないようにせよ
- ただし、`displayActive`は活性状態、`displayInactive`は非活性状態の見た目となるようにせよ（わかればなんとなくでOK）
- マウスを動かすとキャラクタが動き、マウスのボタンが押されているときに非活性状態となり、そうでないときには活性状態となる

# 宿題の提出方法

---

- 提出期限は次回の13:00まで
  - それ以降のものは受理されませんので早め早めに提出をよろしくです！
- Google Driveの各自の宿題フォルダに提出

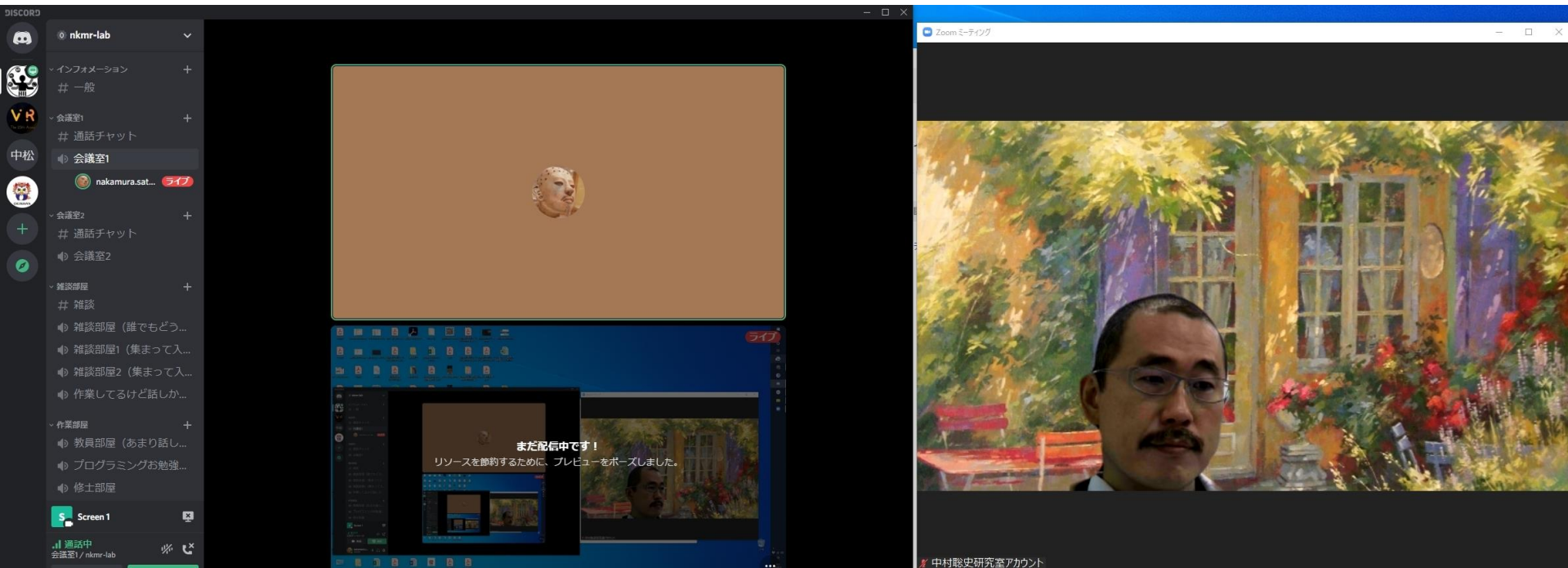
# Discordを活用してください

---

- 答えを見せることはダメですが、教え合うことは大歓迎です。是非ともDiscordを活用してください。
- プログラムを見せなければ、一致することはまず有りませんので大丈夫。見せたことはすぐバレますので要注意！
- Discord、講義を受けてる時間に毎回ログインして使うと良いと思います！
- Discordは遊びに使うのも大歓迎です
  - ゲーム大会とかしたらいいんじゃないかな？

# Discordを利用したみんなを受講

- グループメンバーの誰かがZoomに接続
- その学生さんがZoomの画面をDiscordで共有
- グループの他のメンバーはZoomを開かず、Discordで講義を受講する



# 本日の課題

---

- check
  - まずはcheck（出席確認）申請
  - 再履修の学生さんにはあとで詳しく説明します
- discord
  - Discordに接続して、ニックネームを1年生は「組-番号-名前」、2年生は「2-組-番号-名前」としてご下さい
    - 数字とハイフンは半角でお願いします
  - 次に、研究室部屋または再履修部屋に入り、誰かと音声通話および画面共有をテストしたら提出してください
    - Remoのカメラ+マイクオフをしたほうが良いと思います
  - 以上が完了したら、discordにチェックを入れて申請してください
  - 全員完了し次第、次へ行きます

# ここから

---

- 現役組

- remoをログアウトし、Discordでプレゼン大会
  - まずは、研究室グループに入ってください
  - 入り次第、発表会で作ったプログラムについて、みんなに見せつつ、こだわりポイントなどを話して、他の人はどうやってそれを作ったの的な話をしてください
  - 随時、テキストチャットでアナウンスします
- 時間が来たらテキストチャットに案内出しますので解散

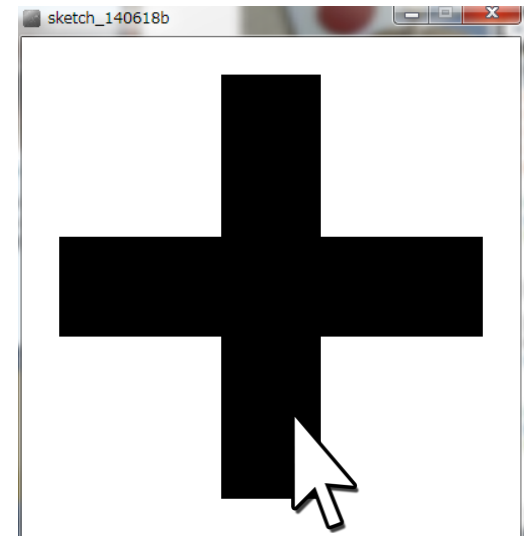
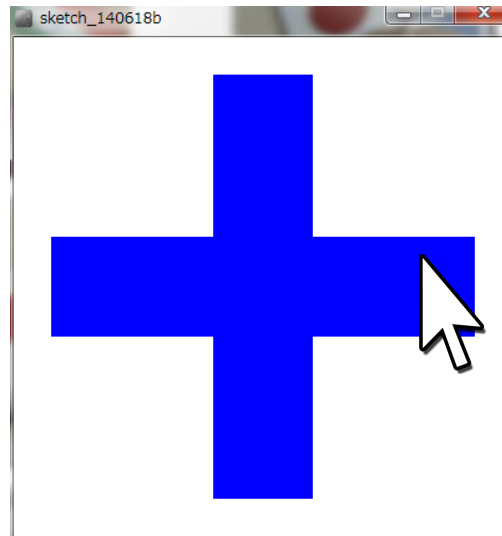
- 再履修組

- Remo Conferenceで課題提示
- 課題の提出テスト
- 課題確認できたら解散

# プログラミング演習II（再履用） 課題

## • スケッチ名：basic\_Cross

- 十字型の図形に対してマウスの当り判定をするプログラムを作ってください。
- 適当な十字型の図形を描き、図の【内部】をクリックしたら図形の色が赤→青→黒→赤→青→黒と順に変わるようにせよ。
- 図形外部をクリックしても反応しないようにすること。



# プログラミング演習II (再履用) 課題

- **basic\_ShowMinimumRect**

- ある点の座標を  $(x[i], y[i])$  と表現する点集合を考える
- この点集合を引数として与えたときに, その点群を囲う最小の四角形 (中を塗りつぶさないもの) を描画する関数 `drawMinimumRect` を作成したい
- 下図が出力例となるので確認せよ
- `basic_ShowMinimumRect.zip` を利用せよ

