

# ソフトウェア概論 A/B

-- ガイダンス/Hello, World --

数学科 栗野 俊一 / 渡辺 俊一

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く  
禁じます

# 伝言

---

## 私語は慎むように !!

□ 出席パスワード : 20200424

□ 色々なお知らせについて

○ 栗野の Web Page に注意する事

<http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino>

# お知らせ

---

## □ 本日の予定

- 「ソフトウェア概論」ガイダンス
- 講議の方針と成績
- 講議中の諸注意
  - ▶ 私語の扱い / 質問の扱い / 遅刻について

## □ 本日の目標

- 講議の進行方針を把握する
- 演習
  - ▶ ハングアウトで、メッセージを送る
  - ▶ 仮想環境上の `ubuntu` が動かせるようにする
  - ▶ 課題の提出

# 担当者について

---

ソフトウェア概論 A/B (2020/04/24)

## 担当者について

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 担当者

---

## □ 教員(二名)

- 渡辺 俊一 (わたなべ しゅんいち) 先生
- 栗野 俊一 (くりの しゅんいち) : ダブル「しゅんいち」

# 講義の方針と成績

---

## □ 講義の方針

- 基本は去年の「コンピュータ概論」と同じって事

- ▶ 復習と、昨年コンピュータ概論を受けていない人のために..

- 内容：講義 + PC 演習

- ▶ 毎回 Note-PC と電源が必須 (ネットワークも利用する)

- 資料：原則 On-Line

- ▶ 栗野の Web Page : <http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino>

- ▶ 当日の朝公開：後日(講義中?) 修正される事もある

- ▶ プリントは配らないので必要な物は、各自、自分でプリント

- 毎回提出課題が出る

- ▶ 課題の提出期限は、課題が提示された講義の次の講義のある日内

## □ 評価：課題, 試験, 講義中の態度

- 課題は全て提出が原則 (全提出で 100%)

- ▶ 課題毎の期限はあるが、最終期限(試験の当日の朝)までは何時でも受け取る

- ▶ 出席は取るが、成績には反映しない

- 試験は、最後の講義の時間に行う

- ▶ 持ち込みはなんでも「可」

# 課題

---

ソフトウェア概論 A/B (2020/04/24)

## 課題

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 先週の課題

---

- 先週の課題
  - なし (初回なので)

# 今週 (2020/04/24) の課題

---

## □ 今週 (2020/04/24) の課題

### ○ 次の C Program ファイルを作成し提出しなさい

▶ 今回は提出先は二つある ( CST Portal / e-mail )

### ○ CST Portal

▶ ファイル名 : 20200424-01-QQQQ.c (QQQQ は学生番号)

▶ 内容 : 「Hello, 自分の名前」を出力する C 言語のプログラム

▶ ファイル形式 : テキストファイル(C 言語プログラムファイル)

### ○ NU-AppsG のメール機能を利用して課題を提出する

▶ 宛先: kurino.shunichi@nihon-u.ac.jp

▶ 表題:「ソフトウェア概論:20200424-01-QQQQ」

▶ 内容: 自分の学籍番号と名前

▶ 添付: 20200424-01-QQQQ.c (QQQQ は学生番号)

# 講義中のルール等

---

ソフトウェア概論 A/B (2020/04/24)

## 講義中のルール等

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 講義中のルールなど

---

## □ 相談 (演習中)

- 大いにやろう / 互いに教え合おう / コピー可
  - ▶ ただ、コピーだけしていると試験で点が取れない
- 得意そうな人と友達になれ
  - ▶ 教えると、より学習が進む
- インターネットを活用しよう
  - ▶ ポータルなどで、情報交換しよう
  - ▶ メールで質問もしよう

## □ [禁止]

- ID/PW を教えて、代わりに操作してもらう事
  - ▶ 自分で操作しなければ、覚えません !!
  - ▶ 教えてもらいながらやるのは問題なし
- 講義中に話をする事 (演習中は可)
  - ▶ 私語厳禁 -> PC でチャット/ツイート/ラインしてください

# 私語について

---

## □ 講義中の私語は厳禁

- 教室内を騒がしくして他人の邪魔するのは止めてください
  - ▶ 貴方には詰まらなくても、他人は興味をもっているかもしれません
- 他の学生の学習を妨害しようとする行為は厳罰します
  - ▶ 講義中に騒ぐ学生は教室の外に追い出します
  - ▶ 「望ましくない態度」としてカウントします

## □ 内容が詰まらなかつたら..

- どうぞ、ボイコットしてください (教室を出ましょう)
  - ▶ 後で、抗議のメールを出すと良いでしょう
  - ▶ 掲示板で批判するのも適切な手段です
- 出席は取りません
  - ▶ 単位を取るには課題を出し、試験で結果を出せば十分です
  - ▶ 課題の内容は Web を参照すれば良いので出席しなくても大丈夫です

## □ 受講者間での情報交換

- ネットワーク経由で行いましょう
  - ▶ メール / チャット / 掲示板

# 口答での質問について

---

## □ 質問をする事は基本的に良い事です

○「質問」とは「聞きそびれた事を尋ねる事」ではありません

▶ 公開している資料に書かれている内容は自分で読んで確認してください

○「質問」とは、次の様な形をした物です

▶ 「話を聞いていて～までは判ったのだが、～の部分が理解できません」

▶ 「～に関する内容は、～に有る事は判ったが、その中の～の意味が解りません」

▶ 「～の資料にあるように～の操作をしたのに、～の様にならないでますが」

○「全然わかんないから教えて」は、質問とは認めません

▶ 「聴いていませんでした」「見ていませんでした」も基本的に対応しません

▶ この手の「発言」は、「講議の邪魔」と解釈します

▶ 「ちゃんと質問したい人」の時間を奪っていますので

## □ 疑問の提示は口頭でなく、ネットワークを利用しましょう

○ 知りたい事(質問を含む)はできるだけ掲示板で

▶ 情報を共有するように努力しましょう (講議への不満も重要な情報!!)

▶ 私的な内容を含む場合は、メールを利用しましょう

# 「望ましくない態度」

---

## □「望ましくない態度」とは？

- 教室で、他の学生の学習を阻害する行為

  - ▶ 講義中に 私語をする / 騒ぐ / 音を鳴らす / etc..

- 自分が聴かなかった事を質問する

  - ▶ 「自分 \*だけ\* がよければ良い」という態度

  - ▶ 「質問をする権利」には「自分でできる事は済ませておくという義務」が伴う

## □以下は、「望ましくない態度」とカウントしません

- 欠席 / 居眠り / 生理現象への対応

  - ▶ ボイコットなどによる途中退出を含む (当然の権利)

- 抗議のメール / 掲示板などでの講義内容に関する批判

- 講義と無関係な Web 参照 / チャット

## □「望ましい態度」とは？

- 講義の発言内容に関する 3 秒以内の質問

  - ▶ 聴いて変だと思ったら、その時点で発言しよう (手を挙げるのがベター)

- 資料に関する、誤りや、不明な点の指摘

# 遅刻者の悲劇

---

## □ 遅刻の何が悪いか

- 遅刻した本人が被害を被るのは、本人の問題
- 問題は、遅刻者が被害を減らすために、他人に迷惑をかけること

## □ 遅刻者の罪

- 講義中に入ってきて、わざわざ真中に来る
  - ▶ 後から静かに入って、他の人の邪魔にならないように遅刻者専用の廊下側の一行に座る
- 友達と挨拶をしようとする
  - ▶ 既に講義は始まっている、他の人の邪魔をするな
- 友達に状況を確認しようとする
  - ▶ 不安は解るが、今の内容を聞け/不足は講義終了後にしろ
- 重要なアナウンスをききおとす
  - ▶ 講義の最初に重要なアナウンスがある事が多い/後で確認しろ

## □ 遅刻はしょうがない

- 遅刻者として自覚をもってふるまわない事が罪

# 講義の目標

---

ソフトウェア概論 A/B (2020/04/24)

## 講義の目標

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 講議の目標

---

## □ 講議の目標

- C 言語でプログラムが書けるようにする

## □ 講議の内容

- プログラミングとは何か
- プログラミング言語 C とは
- プログラムを作成する手順
- アルゴリズム
- データ構造

## □ 講議の方針

- 実践的にやりたい (希望)
  - ▶ そのための工夫を考えている

# プログラミングの学習方法

---

## □ プログラミングの学習方法

- 結局は、実学、コンピュータの学び方と同じ

- ▶ 習うより慣れる

- ▶ 学(まな)ぶは真似(まね)る

## □ プログラムの面白さ

- 自分の思った内容が、実現できる

- すぐに役立つ物が作れる

## □ 情報を収集しよう

- 自分と同じ事をしたい人は沢山いる

- ▶ まずはググってみよう

# 環境設定

---

ソフトウェア概論 A/B (2020/04/24)

## 環境設定

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 仮想環境と C 言語環境

---

## □ C 言語プログラミング環境の構築

- Windows 上に環境を作るのは色々と不便

  - ▶ Unix (Ubuntu/Linux) 上には始めから揃っている

- Unix 環境を作れば、C 言語の環境もついてくる

## □ ソフトウェア概論でのプログラミング環境

- ubuntu アプリケーションで作業を行う

## □ 前年度コンピュータ概論からの移行

- 前年度、コンピュータ概論を受講した人は環境の設定は変更だけです(既に済んでいる)

  - ▶ きちんと ubuntu が動くかどうか確認してください

## □ Ubuntu のインストールができていない人

- 後日、追加動画を公開する予定

# 環境の確認

---

## □ ubuntu を起動

- 背景が黒いウィンドウが表示される
  - ▶ [Enter] キーを押すと、同じメッセージが表示される

## □ Ubuntu の終了

- バツ[x]ボタンで閉じる

# 作業フォルダの作成

---

## □ windows 側

- c:\usr\c を作成する
- その下に、本日の作業フォルダ (20200424) を作成する
- 作業フォルダの中に、サンプルプログラム (sample-001.c) をダウンロード

## □ Ubuntu 側

- Ubuntu を開く
- ln -s /mnt/c/usr/c .

## □ 確認

### ○ Ubuntu 側

- ▷ cd c
- ▷ ls -> 20200424 が表示される
- ▷ cd 20200424
- ▷ ls -> ダウンロードしたファイルが表示される

# C Compiler のインストール

---

- Ubuntu で C Compiler をインストールする
  - sudo apt update
  - sudo apt upgrade
  - sudo apt install gcc
    - ▷ パスワードは、「soft」をいれる

# プログラムの作成と実行

---

ソフトウェア概論 A/B (2020/04/24)

## プログラムの作成と実行

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# プログラムの作成と実行

---

## □ ファイルの共有

- windows の c:\usr\c が、ubuntu の /home/soft/c に対応

- ▶ 一方を変更すると他方も変更される

## □ プログラムの作成

- プログラムは暫く、windows の Sakura Editor を使う

- ▶ Windows 上の Sakura でプログラム作成

- ▶ Ubuntu 上 C Compiler で、プログラム実行

# 実習の流れ

---

## □ 準備 (教室に来たら、一度だけやれば良い)

### ○ ubuntu を起動

▶ 「ls c [Enter]」とする : 色々表示されれば OK、駄目なら、次のコマンド

▶ ln -s /mnt/c/usr/c .

### ○ 「cd c [Enter]」とする

## □ プログラム実行迄の流れ(ファイル毎に毎回実行)

○ ソースプログラムファイル ( foobar.c ) を Sakura Editor で作成し c:/usr/c に置く

○ ubuntu の端末内のコマンドで、次のコマンドを実行

▶ cc -c foobar.c

▶ cc -o foobar.exe foobar.o

▶ ./foobar.exe

○ ※コンピュータ概論で学んだ TeX と同じ仕組

# 日本語と改行

---

## □ 日本語の利用

- プログラム内で日本語を利用する場合は `utf-8` で表現する必要がある

- `ubuntu` の場合

- ▶ 標準で `utf-8` が利用されるので気を使う必要はない

- `windows` の場合

- ▶ 標準では `SJIS` が利用されるので、注意が必要

- ▶ `Sakura Editor` で、次の様に設定すれば、`utf-8` になる

- ▶ [設定] -> [文字コードの指定] -> 文字コードを「`UTF-8`」にする

## □ 改行コード

- プログラム内の改行コードは `LF` コード (`[Ctrl-J]`) 一文字である

- `ubuntu` の場合

- ▶ 標準でそうなので、問題ない

- `windows` の場合

- ▶ 標準では `CR LF` (`[Ctrl-M]`, `[Ctrl-J]`) の二文字となる

- ▶ `Sakura Editor` で、次の様に設定すれば、`LF` のみになる

- ▶ [設定] -> [入力改行コード指定] -> (`LF`)

# 演習

---

ソフトウェア概論 A/B (2020/04/24)

## 演習

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 演習

---

## □ 演習

- Sakura Editor でプログラムを作成
- Ubuntu 環境で、プログラムをコンパイル実行する
- プログラムファイルを CST Portal II に提出する