

	到達目標	講義内容
<div> <div>#00a 講義説明 2022 年度 / プログラミング及び実習 III</div> <div>角川裕次 龍谷大学 先端理工学部</div> </div>	<p>【計算機システム】</p> <ul style="list-style-type: none"> ・ 計算機システムの基本的な仕組みを理解する ・ 情報システムの実現に必要な基礎知識を学ぶ <p>【プログラミングの基礎概念】</p> <ul style="list-style-type: none"> ・ コンピュータプログラミングの基礎概念を理解する <p>【C 言語プログラミングの基礎技術】</p> <ul style="list-style-type: none"> ・ C 言語プログラミングの基礎技術を身につける ・ 簡単な C 言語プログラムを読んで理解できる ・ 自分で簡単な C 言語プログラムを書くことができる 	<p>C 言語によるプログラミングの基礎から実践的手法まで</p> <div> <div> C 言語基礎 <ul style="list-style-type: none"> ■ 計算機アーキテクチャ ■ 変数 ■ 関数 ■ ポインタ ■ 文字・文字列 </div> <div> 実用プログラム基礎技法 <ul style="list-style-type: none"> ■ 動的メモリ割り当て ■ コマンドライン引数 ■ 分割コンパイル ■ パフォーマンス測定 </div> </div> <p>応用プログラム基礎技法</p> <ul style="list-style-type: none"> ■ 構造体 ■ ファイル ■ ファイルシステム
1 / 16	2 / 16	3 / 16
各種連絡用の情報システム：manaba★重要★	講義・実習の進め方★重要★	教科書★重要★
<p>manaba を使用します</p> <ul style="list-style-type: none"> ■ 重要事項の通知 ■ 講義資料の配布 ■ 実習・課題の提出 <p>respon を使用します</p> <ul style="list-style-type: none"> ■ 出席確認 ■ respon 出席と提出物の提出の両方が揃って出席とします <p>Teams も併用</p> <ul style="list-style-type: none"> ■ 講義録画 ■ コロナ再拡大のときはオンライン講義ライブ中継 	<p>講義時間中：「説明」、「実習」、「課題」</p> <ul style="list-style-type: none"> ■ 説明：教科書・配布資料に沿って教員が解説 ■ 実習：教科書掲載のプログラムを入力・実行・理解 ■ 課題：教科書に掲載の演習問題に対して各自でプログラム作成 <p>課題</p> <ul style="list-style-type: none"> ■ 時間内で出来なければ講義終了後の宿題とします ■ できるかぎり講義時間内で完成しましょう <p>提出物 (manaba)：実習 & 課題 (提出の有無は各回で異なる)</p> <ul style="list-style-type: none"> ■ 受付開始：その日の夜 10 時 ■ 提出〆切：次の週の朝 6 時 ■ 文章・プログラムを熟考する時間を取りましょう 	<p>注意：昨年度 (初版) とは異なります</p> <p>題名：新・明解 C 言語 入門編 第 2 版 著者：柴田望洋 出版社：SB クリエイティブ 発売日：2021/9/10 ISBN-13：978-4815609795</p> <p>早急に購入のこと</p> <p>持っていること前提で講義・実習・課題が進みます</p> 
4 / 16	5 / 16	6 / 16

<div>★重要★</div> <div>プログラミング環境について</div>	成績評価	質問する際の注意
<div>C 言語の開発環境を使用します</div> <div> 実習・課題の進め方のイメージ <ul style="list-style-type: none"> ■ テキストエディタでプログラムコードを入力/作成 ■ コンパイル・実行・デバッグ </div> <div> 主に想定するプログラミング環境 <ul style="list-style-type: none"> ■ BYOD Windows 上の WSL2 ■ BYOD Mac 上の xcode ■ 情報実習室 PC 上の Ubuntu </div> <div> 避けてほしいプログラミング環境 <ul style="list-style-type: none"> ■ paiza.IO (オンライン) ■ Visual Studio (Windows 上) ■ いずれも一部の实習内容 (ファイル関係) が実施できないため </div> <div>7 / 16</div>	<div> <ul style="list-style-type: none"> ■ 期末試験：70% ■ 実習：10% ■ 課題：20% </div> <div> 期末試験で良い得点を得てください 毎週の実習には出席してプログラムを実際に動かして下さい 毎回の課題を完成させて提出して下さい </div> <div>重要: 実習の提出がなければ欠席扱いにします</div> <div>8 / 16</div>	<div> 悪い例 1: (画面を指さして)「これ」 <ul style="list-style-type: none"> ■ 問題点は一体なに? ■ そもそも何をしようとしているの? → 十分な説明を試みましょう (問題点を客観的に認識・説明できるのは基本的・重要な能力) </div> <div> 悪い例 2: 「プログラムが動きません」 <ul style="list-style-type: none"> ■ コンパイルエラーなの? ■ コンパイルはできたけど期待の動作と違うの? ■ エラーメッセージが出たならそれを教えてよ → 何度も聞き返さないとならないです </div> <div> 良い例: 「実習 A のプログラムでコンパイルエラーが取れません。エラーメッセージは“(略)”なので原因は...(略)... だと思ったので、...(略)... したり...(略)... したりしてみたのですが上手く行きません。」 → こういう質問の仕方だと解決が早いことが多いです </div> <div>9 / 16</div>
エラーいっぱい出るのが普通	エラーへの対処法	
<div> 悪い例 1: (画面を指さして)「エラーでした」 → エラーメッセージを読んで下さい <ul style="list-style-type: none"> ■ コンパイルエラーなら行番号も表示されている ■ ソースコード一番最初のエラーメッセージから片付けてゆく (一番最後のエラーメッセージ見ない人おおすぎ) </div> <div> 悪い例 2: 「エラーメッセージの意味が分かりません」 → 辞書でも引いてみましょう </div> <div> 悪い例 3: 「でも意味分かりません」 → ググって下さい。お仲間がいっぱいみつかるよ。 </div> <div> 悪い例 4: 「ググっても答え載ってません」 → 検索結果がトップに来るとは限らないよ。上位 30 位程は見ようね。 </div> <div> 悪い例 5: 「さっぱり分かりません」 → 思い込みが激しいとミスに気が付かないよね... (「正しいはずなのに」を捨てましょう) </div> <div>10 / 16</div>	<div> コンパイル時 (コンパイル時のエラーメッセージ) <ul style="list-style-type: none"> ■ 一番最初のエラーメッセージにある行番号を見る ■ その行のどこがおかしい ■ 又はその行で使用の変数の宣言等がおかしい (ソースコード前方を見よ) </div> <div> 実行時 (思い通りに動かない) <ul style="list-style-type: none"> ■ 所々に printf 文を入れてみる → 実行の進み具合や変数値を表示して動作を追ってみる ■ 配列の添字の範囲を超えていないかチェック ■ ポインタがアクセス可能な範囲を超えていないかチェック </div> <div>わけわからんエラーへの対処が出来るようになるのも能力のうち</div> <div> エラー直せない = 「正しい」と「間違い」の区別ができてない = 理解できてない・分かったつもり </div> <div>11 / 16</div>	<div>講義スケジュール</div> <div>12 / 16</div>

シラバス記載の授業計画と実施回の対応 (1/3)	シラバス記載の授業計画と実施回の対応 (2/3)	シラバス記載の授業計画と実施回の対応 (3/3)
<p>第1週目: CPU・メモリ・機械語・変数・スタック</p> <ul style="list-style-type: none"> ■ 第1回 メモリとアドレス ■ 第2回 機械語プログラムのメモリ配置 <p>第2週目: 関数</p> <ul style="list-style-type: none"> ■ 第3回 変数の有効範囲と有効期間 ■ 第4回 静的/自動変数のメモリ配置 <p>第3週目: 基本型</p> <ul style="list-style-type: none"> ■ 第5回 データ型と sizeof 演算子 <p>第4週目: マクロ, 列挙体, 再帰</p> <ul style="list-style-type: none"> ■ 第6回 関数呼び出しとスタック領域 <p>第5週目: 文字列の基本</p> <ul style="list-style-type: none"> ■ 第7回 文字コードと制御文字 ■ 第8回 文字列データの表現 <p>第6週目: 文字列の操作</p> <ul style="list-style-type: none"> ■ 第9回 文字列のプログラミング 	<p>第7週目: ポインタの基礎</p> <ul style="list-style-type: none"> ■ 第10回 アドレス演算子とポインタ ■ 第11回 ポインタを利用したプログラミング <p>第8週目: ポインタと配列</p> <ul style="list-style-type: none"> ■ 第12回 配列型とポインタ型 ■ 第13回 ポインタの演算 ■ 第14回 配列を引数とする関数 <p>第9週目: 文字列とポインタ</p> <ul style="list-style-type: none"> ■ 第15回 文字列を扱う標準ライブラリ関数 <p>第10週目: 構造体</p> <ul style="list-style-type: none"> ■ 第16回 構造体 ■ 第17回 構造体へのポインタ型 <p>第11週目: 動的メモリ割当て</p> <ul style="list-style-type: none"> ■ 第18回 構造体を利用したプログラミング ■ 第19回 動的メモリ割り当て <p>第12週目: ファイルとストリーム</p> <ul style="list-style-type: none"> ■ 第20回: ファイル入出力の標準ライブラリ関数 	<p>第13週目: ファイル入出力のプログラミング</p> <ul style="list-style-type: none"> ■ 第21回 ファイル入出力のプログラミング <p>第14週目: ファイルシステムのプログラミング</p> <ul style="list-style-type: none"> ■ 第22回 ファイルとディレクトリ ■ 第23回 ファイルシステム ■ 第24回 ファイル操作のシステムコール <p>第15週目: C言語プログラムの実践的開発技法</p> <ul style="list-style-type: none"> ■ 第25回 main の引数 ■ 第26回 ライブラリとリンク ■ 第27回 分割コンパイル ■ 第28回 プログラムの終了コード ■ 第29回 実行時間の測定 ■ 第30回 まとめ
13 / 16	14 / 16	15 / 16
<div> <div>おわり</div> <div>16 / 16</div> </div>		