

● オペレーティングシステム

第1回

オペレーティングシステムの基本概念

http://www.info.kindai.ac.jp/OS
E館3階E-331 内線5459
takasi-i@info.kindai.ac.jp

1

本科目の内容

- 本科目で学ぶこと
 - オペレーティングシステム(OS)の基本概念
 - OSの役割・目的
 - プロセス・スレッドの生成・管理
 - スケジューリング
 - プロセス間通信・デッドロック
 - メモリ管理
 - 割込み
 - 入出力の制御

2

成績について

評価基準	
オンライン課題 (毎週)	30%
オンライン試験 (第15週)	70%

- 無届欠席禁止
 - やむを得ず欠席した場合は翌週までに欠席届を提出すること
 - 無届欠席が複数回ある場合は試験の点数に関わりなく不受となる

オンライン授業では GoogleClassroom から出席カードが提出されれば出席とします

3

昨年度の受講状況

学年	コース	受講者数	合格	不可	不受	合格率
2	システム	89	85	1	3	97%
2	メディア	4	3	1	0	75%
3	システム	2	2	0	0	100%
3	メディア	5	5	0	0	100%
4	メディア	1	1	0	0	100%

※全出席し、全レポートを〆切までに提出して不可になった受講生はいない

4

https://classroom.google.com/h

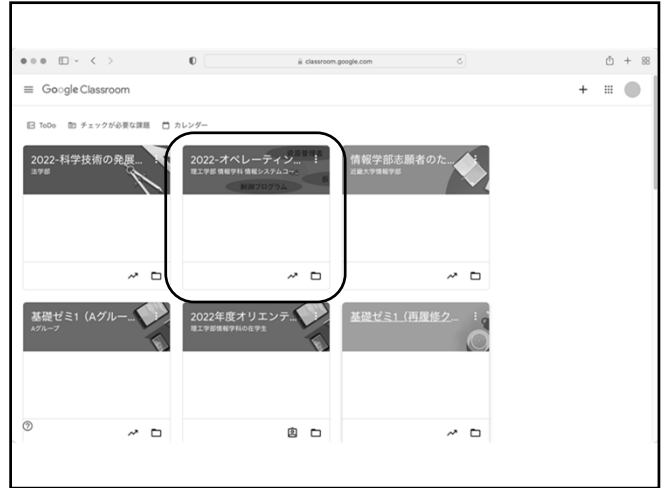
5

https://classroom.google.com/h

6



7



8



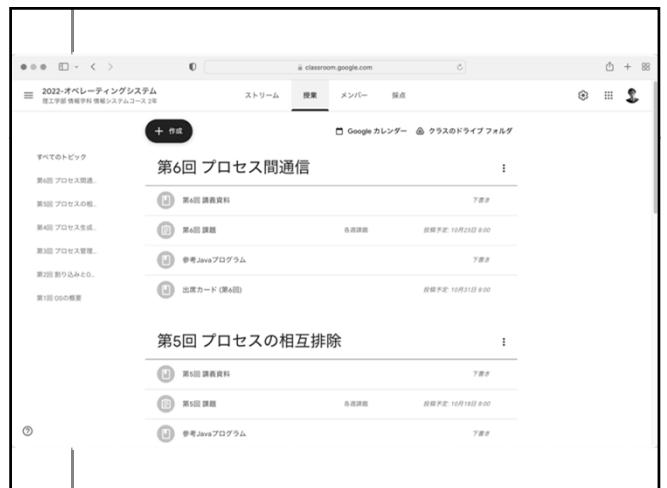
9



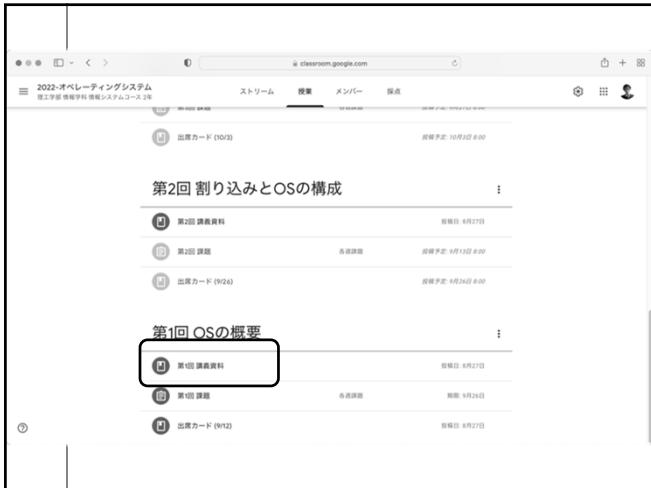
10



11



12



13



14

あらかし

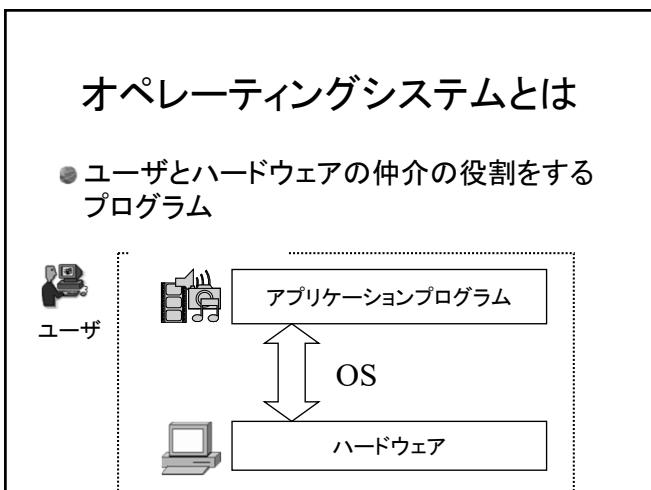
- オペレーティングシステム(OS)とは
- OSの役割
- OSの歴史
- OSの構成

15

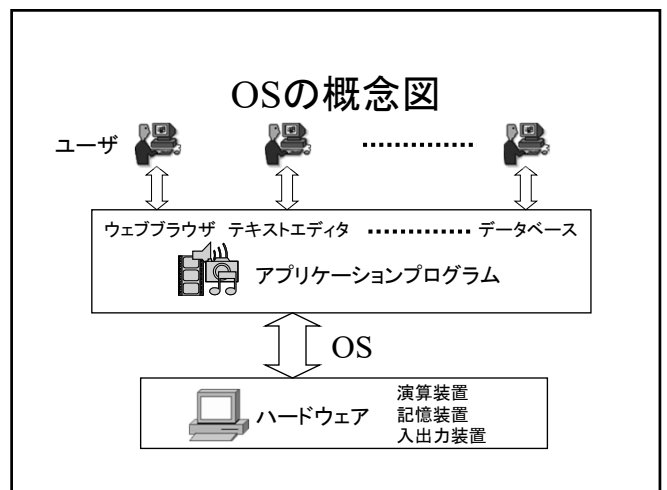
計算機システムの要素

- ハードウェア
 - 基本的な計算資源を供給する
 - CPU, 主記憶, IO装置等
- アプリケーションプログラム(AP)
 - システム資源を利用してユーザの問題を解く
 - コンパイラ, データベースシステム, ビデオゲーム, ビジネスプログラム等
- ユーザ
 - 計算機を利用して問題を解く
 - 人間, マシン, 他の計算機等

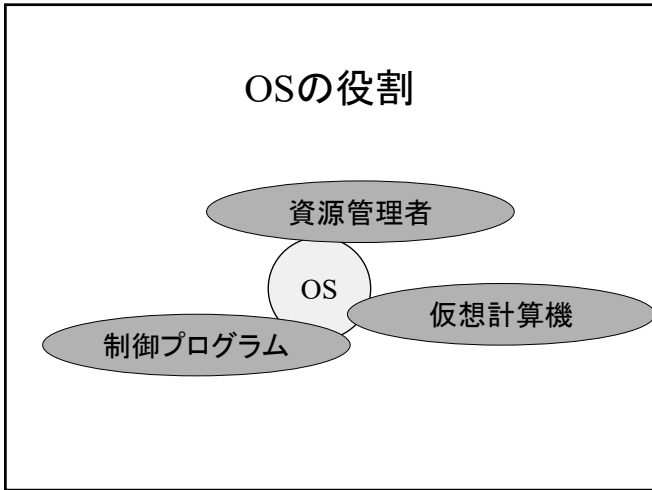
16



17



18



19

OSの役割: 資源管理者 (resource manager, resource allocator)

- 資源
 - ハードウェア: CPU, メモリ, IO装置, ハードディスク
 - ソフトウェア: プログラム, データ等
- 目的
 - 共有資源の状態を維持
 - APからのアクセスを管理, 資源を適正に利用させる
- 要求
 - 効率
 - 公平性

公平性を追求すると
効率が下がる
逆も同じ

↻ トレードオフ

20

OSの役割: 資源管理者

- 資源管理(resource manager)
 - ハードウェア資源の管理
- 情報管理(information manager)
 - ソフトウェア資源の管理

21

OSの役割: 資源管理者

22

効率と公平性

大通り: 3600台/時間
路地: 1台/時間

この信号をどう設定する?

23

効率と公平性: 公平性優先

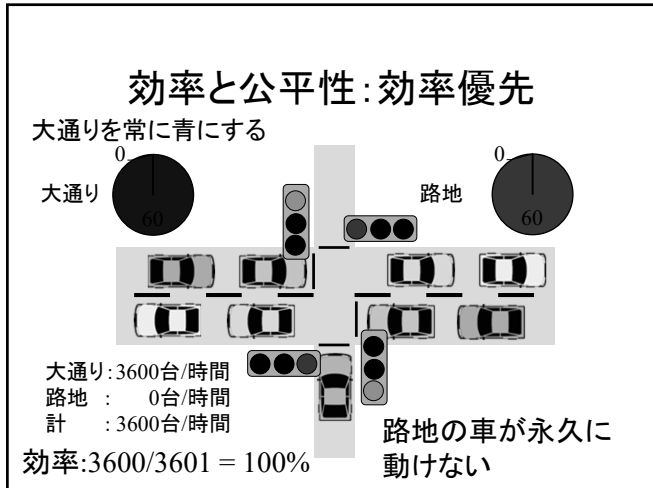
赤と青を完全に半々にする

大通り: 1800台/時間
路地: 1台/時間
計: 1801台/時間

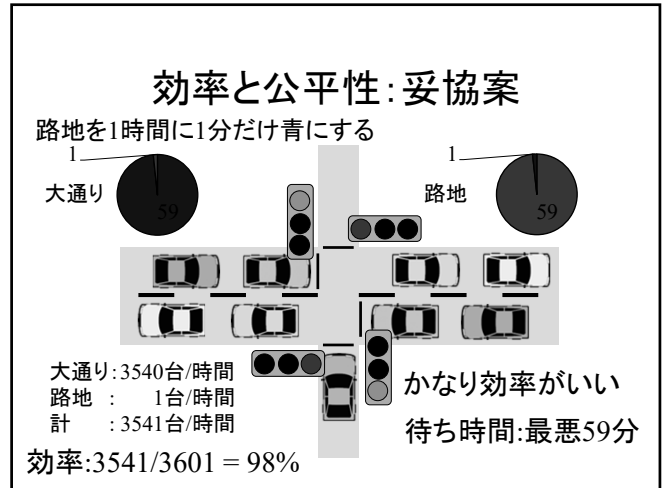
効率: $1801/3601 = 50\%$

公平だが効率が悪い

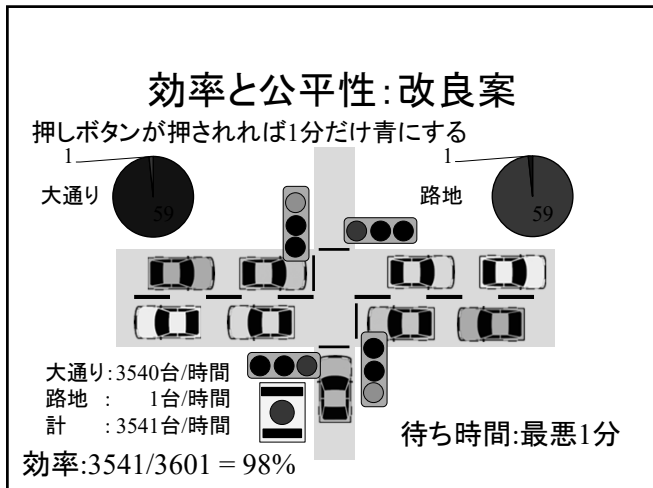
24



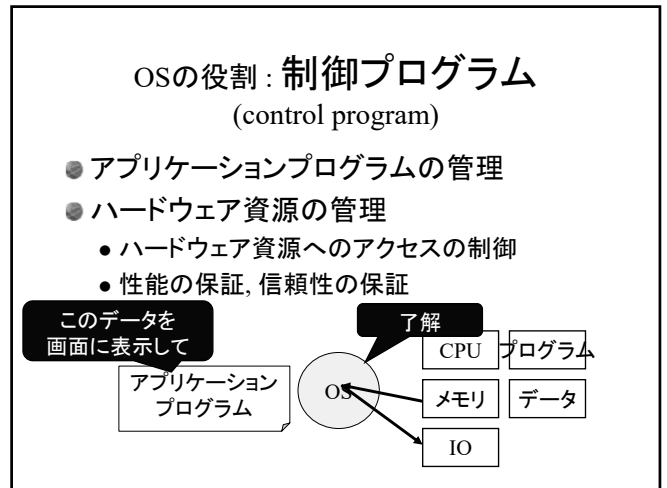
25



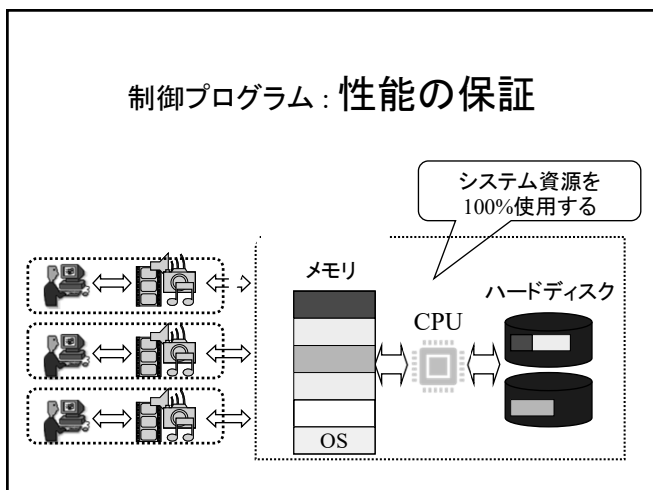
26



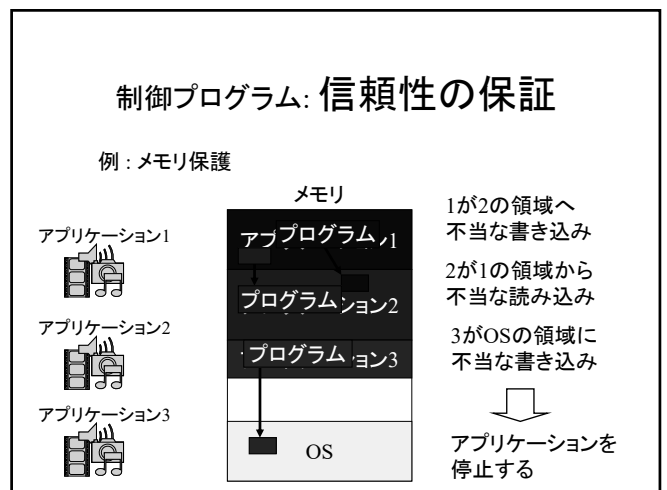
27



28



29



30

OSの役割: **仮想計算機**(拡張計算機)
(virtual machine, extended machine)

- ハードウェアを抽象化
- 抽象概念に対するアクセス手段の提供
- プログラミング環境の提供

31

仮想計算機:
計算機ハードウェアの抽象化

ユーザーから見たデータ
使い易さを提供する

実際のHDの中身

32

仮想計算機:
計算機ハードウェアの抽象化

33

仮想計算機:
計算機ハードウェアの抽象化

例: データのコピー

物理特性を気にしなくて良い

それぞれ物理特性が異なる

34

仮想計算機:
抽象概念に対するアクセス手段の提供

ハードウェア	抽象概念	
メモリ	仮想アドレス空間	ハードウェアに直接アクセスさせない
ディスク装置	ファイル	
IO装置		不正アクセスの抑止
ネットワーク	プロセス	
CPU		使い易さの提供

35

仮想計算機:
使い易さの提供

- ユーザーがプログラムを簡単に実行できる

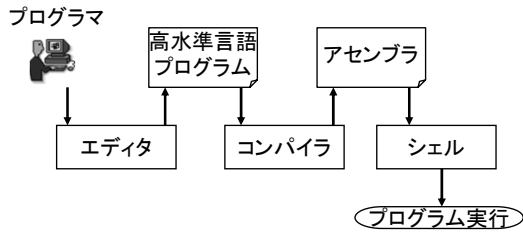
コマンド入力によりプログラムを実行可能

アイコンクリックによりプログラムを実行可能

36

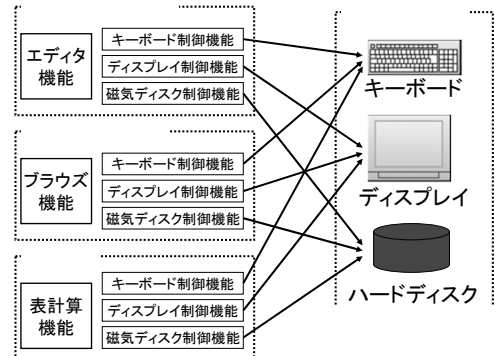
仮想計算機： プログラミング環境の提供

- コンパイラ, エディタ, シェル等と協調



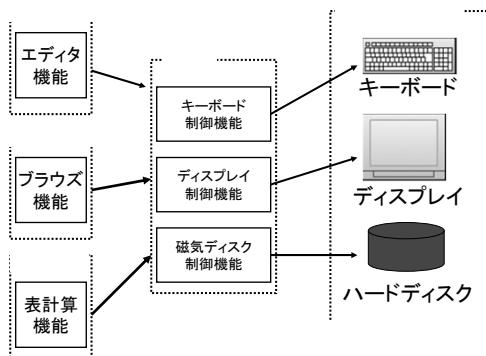
37

もしOSが無かったら...



38

OSがあれば...



39

OSの役割

- 資源管理者(resource manager)
 - ハードウェア資源,ソフトウェア資源の管理, ユーザへの割り当て
- 制御プログラム(control program)
 - アプリケーションプログラム, ハードウェアの管理
 - 性能の保証, 信頼性の保証
- 仮想計算機(virtual machine)
 - プログラミング環境を実権
 - “使い易さ”の提供

40

OSの歴史

第0世代	1940年代
第1世代	1950年代
第2世代	1960年代前半
第3世代	1960年代後半
第4世代	1980年代

41

OSの歴史(第0世代)

- 第0世代(1940年代)
 - OSは存在しない
 - ユーザは機械語でプログラムを作成, コードの配線やパネルスイッチで入力
- 機械語
- 1と0のみで記述
 - ハードに依存 (計算機ごとに異なるプログラムが必要)

0101 B409
0102 BA09
0105 CD21
0107 CD20
:

42

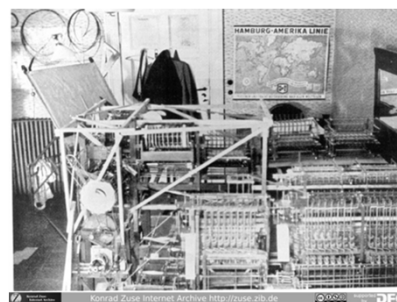
Z1

● Z1

- 1938年 コンラッド・ツーゼ作
- 22ビットの機械式メモリ64個
- 22ビットの2進浮動小数点演算
- 世界最初のプログラム可能な計算機
 - ...の予定だったが誤作動が多く信頼性にかける
 - 使用済映画用フィルムでプログラム
- 1940年空爆により破壊される

43

Z1

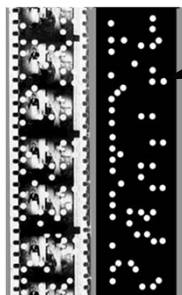


ツーゼの計算機 Z1 [1]

[1] Konrad Zuse Internet Archive, <http://zuse.zib.de/>

44

Z1のプログラムフィルム



使用済フィルムに
穴を空けて
プログラム

Z1のプログラムフィルム [1]

[1] https://upload.wikimedia.org/wikipedia/commons/6/6f/Zuse_Z1.jpg

45

Atanasoff-Berry Computer

● ABC

- 1942年 ジョン・アタナソフとクリフォード・ベリー制作
- 50ビットの固定小数点演算
- 世界初の電子計算機
- 29元連立方程式を解ける
- ほぼ出来上がっていたが、アタナソフが戦時研究に駆り出されている間に解体された

46

Atanasoff-Berry Computer



Atanasoff-Berry Computer (複製) [1]

[1] https://upload.wikimedia.org/wikipedia/commons/9/90/Atanasoff-Berry_Computer.jpg

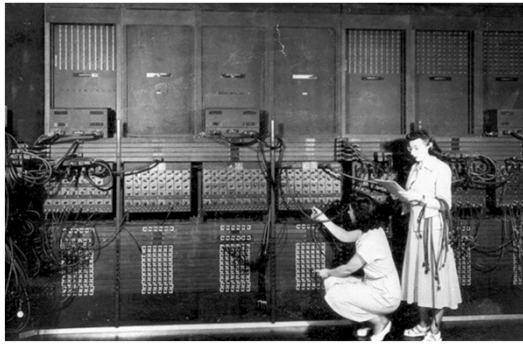
47

Electronic Numerical Integrator and Computer

● ENIAC

- 1946年 ジョン・モークリーとジョン・エッカート制作
- 弾道表作製計算機
- 真空管式電子計算機
- プログラムは配線によりセット
 - 配線作業は1週間必要
- ~~「世界初の電子計算機」~~
 - 世界初は Atanasoff-Berry Computer

48



ENIACのプログラミング風景[1]

[1] John W. Mauchly and the Development of the ENIAC Computer, An Exhibition in the Department of Special Collections, Van Pelt Library, University of Pennsylvania, <http://www.library.upenn.edu/exhibits/rbm/mauchly/jwmintro.html>

49

最初の計算機

年代	名称	備考	考案者・製作者
1938年	Z1	最初のプログラム可能な計算機	コンラッド・ツェーゼ
1941年	Z3		
1942年 未完成	ABC	最初の電子式計算機	ジョン・アタナソフ クリフォード・ベリー
1943年	コロサス	最初の真空管式大規模機械	アラン・チューリング トム・フラワーズ
1944年	ASCC	最初に実用化されたプログラム可能な計算機	ハワード・エイケン
1946年	ENIAC	最初の実用化された電子式計算機	ジョン・モークリー ジョン・エッカート
1948年	SSEM	最初に稼働したフォン・ノイマン型計算機	フレデリック・ウィリアムズ トム・キルバーン
1949年	EDSAC	最初に実用化されたフォン・ノイマン型計算機	モーリス・ウィルクス
1951年	EDVAC	最初にフォン・ノイマン型を提唱した計算機	ジョン・モークリー ジョン・エッカート

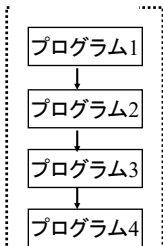
50

OSの歴史(第1世代)

● 第1世代(1950年代)

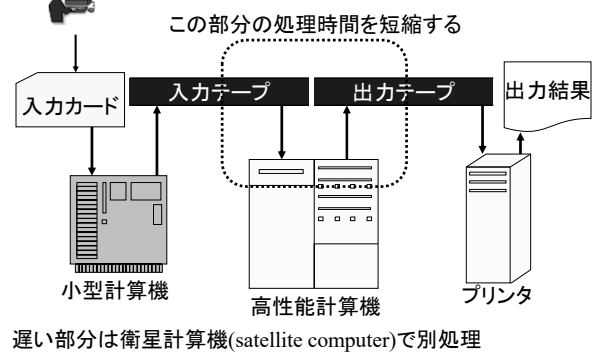
- バッチ処理(batch processing)
 - 複数のプログラムを連続して処理

同様なジョブを一括処理(batch)することによりセットアップ時間を軽減



51

OSの歴史(第1世代): バッチ処理の流れ



52

OSの歴史(第2世代)

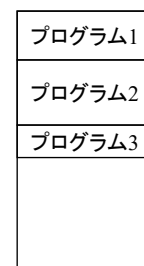
● 第2世代(1960年代前半)

- マルチプログラミング(multiprogramming)
 - 並行処理(concurrent processing)
 - 複数のプログラムを切り替えて実行
 - タイムシェアリング(time-sharing)
 - 複数のユーザの仕事进行处理
 - 仮想記憶(virtual address)
 - 主記憶よりも大きい記憶空間の提供

53

OSの歴史(第2世代): マルチプログラミング (multiprogramming)

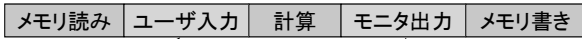
- 複数のプログラムを同時にメモリに置く



54

OSの歴史(第2世代): マルチプログラミング

- プログラムにはユーザの入力やIO処理が含まれる



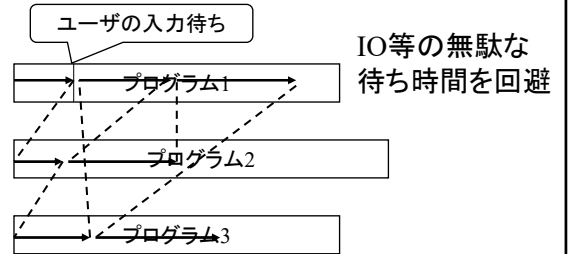
待っている間はプロセッサはアイドル(何もしない)状態

↓
待ち時間の間に他のプログラムを行える

55

OSの歴史(第2世代): マルチプログラミング

- CPUはプログラムを高速に切り替えて実行



56

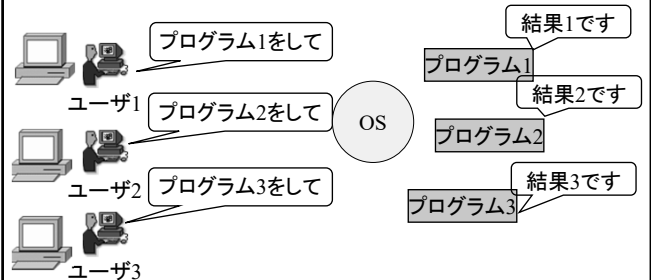
OSの歴史(第2世代): マルチプログラミング

- マルチプログラミングに必要な能力
 - システムに備えられたIOルーチン
 - メモリ管理
 - 複数のプログラムをメモリに割り当てる
 - CPUスケジューリング
 - 複数のプログラムからどのプログラムを次に実行するか決定する
 - デバイスの割り当て

57

OSの歴史(第2世代): タイムシェアリング (time-sharing)

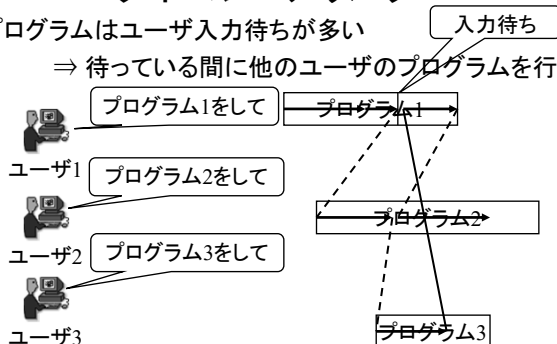
- 複数のユーザの仕事に対話的に処理



58

OSの歴史(第2世代): タイムシェアリング

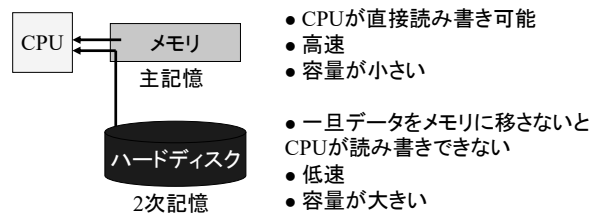
プログラムはユーザ入力待ちが多い
⇒ 待っている間に他のユーザのプログラムを行う



59

OSの歴史(第2世代): 仮想記憶 (virtual address)

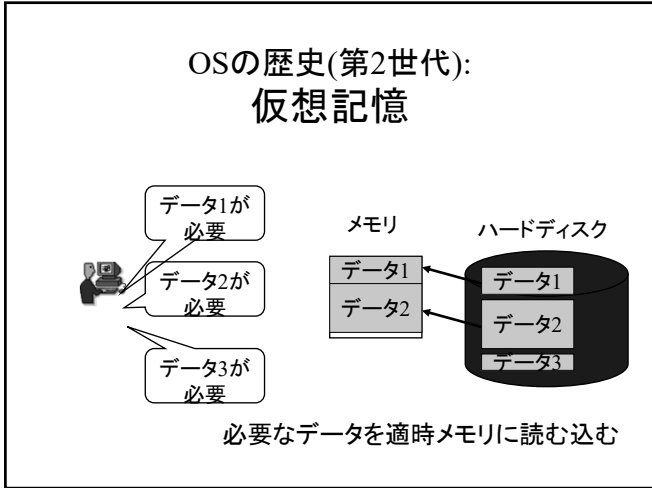
- 主記憶より大きい記憶空間の提供



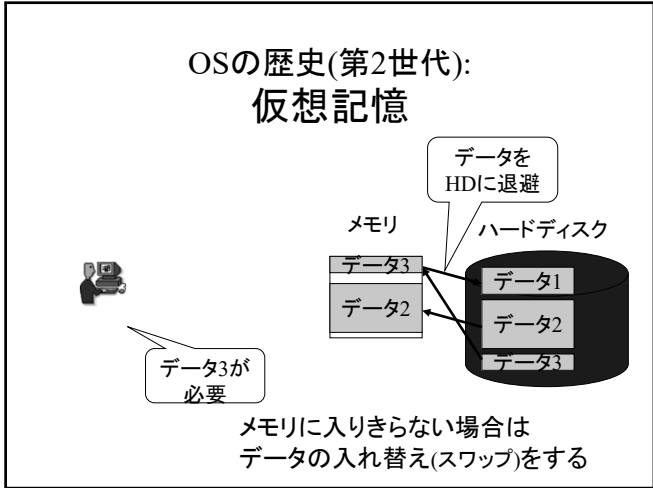
- CPUが直接読み書き可能
- 高速
- 容量が小さい

- 一旦データをメモリに移さないとCPUが読み書きできない
- 低速
- 容量が大きい

60



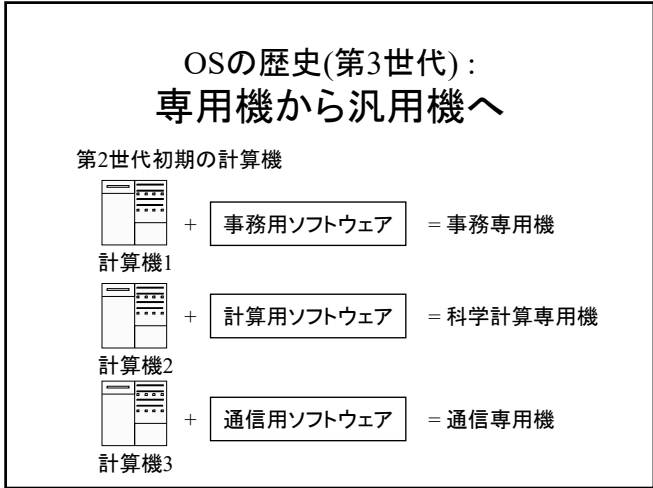
61



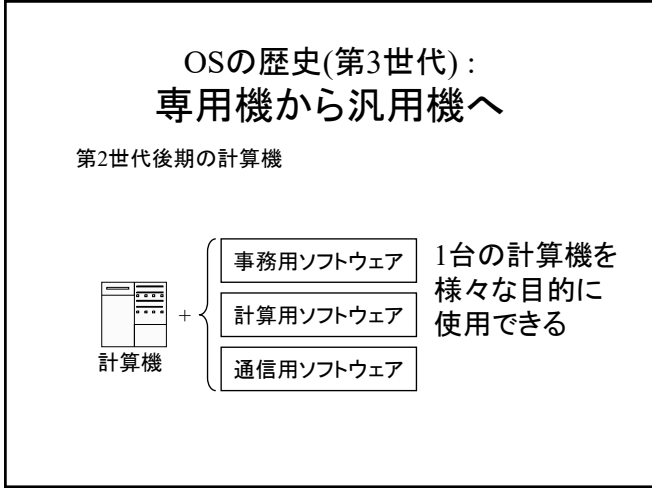
62

- ### OSの歴史(第3世代)
- 第3世代(1960年代後半)
 - 汎用大型計算機
 - IBM System/360 の OS/360
 - MULTICS
 - シミュレータ, エミュレータ

63



64



65

- ### OSの歴史(第3世代): 専用機から汎用機へ
- 多くのユーザが汎用性を求めた
 - 科学技術計算
 - 商用計算
 - 事務処理
 - 通信
- 多数の機能を持つ反面、システムが巨大化してしまった
- ### 汎用大型計算機

66

OSの歴史(第3世代) : IBM System/360

- 1964年IBMが開発
- 初の本格的OS OS/360 を搭載



System/360[1]

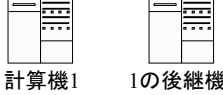
[1] System 360 From Computers to Computer Systems, IBM
<http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/system360/impacts/>

67

OSの歴史(第3世代) : IBM System/360

- IBMは System/360 の後継機を次々と発表
 - System/370, 4300, 3080, 3090

計算機ファミリー

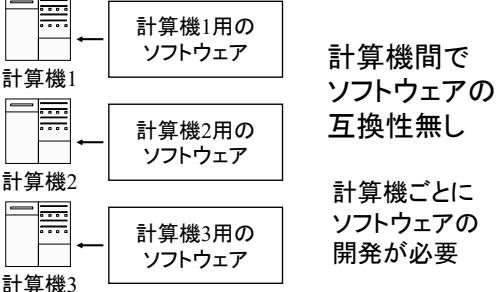


後継機は計算機1のソフトウェアをそのまま使える

68

OSの歴史(第3世代)

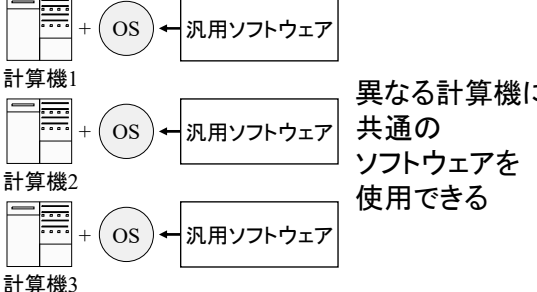
- 第2世代以前の計算機



69

OSの歴史(第3世代)

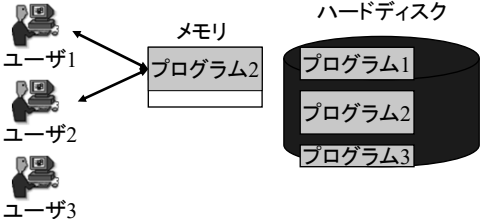
- 第3世代以降の計算機



70

OSの歴史(第3世代) : CTSS(the compatible time-sharing system)

- 1963年MITが開発
 - 複数のユーザのプログラムを切り替えながら実行可能



71

OSの歴史(第3世代) : MUTICS (multiplexed information and computing service)

- 1965年MITとベル研が共同開発
 - CTSSの流れを汲むOSの原型
 - 仮想記憶、プロセスの概念、プロテクションの概念等の重要なアイデアが盛り込まれる

しかし実用化はせず

1965年当時

- ハードウェア能力が無い
- ソフトウェア技術が無い

72

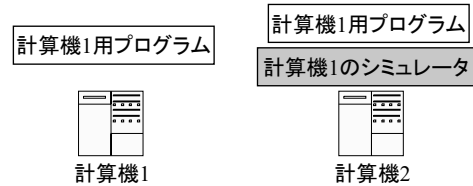
OSの歴史(第3世代): UNICS

- 1968年ベル研が開発
 - MULTICSの失敗(複雑過ぎ, 大き過ぎ)を踏まえたシンプルで小さいOS
- 後に UNIX と改名
- ただし現在は巨大化

73

OSの歴史(第3世代): シミュレータ, エミュレータ (simulator, emulator)

- 他の計算機の命令をシミュレート可能



異なる計算機で同一のソフトウェアが使用可能

74

OSの歴史(第3世代): シミュレータ, エミュレータ

	シミュレータ	エミュレータ
構成	ソフトウェア	マイクロプログラム 機械語
実行速度	遅い	速い
費用	安い	高い
開発難易度	低い	高い

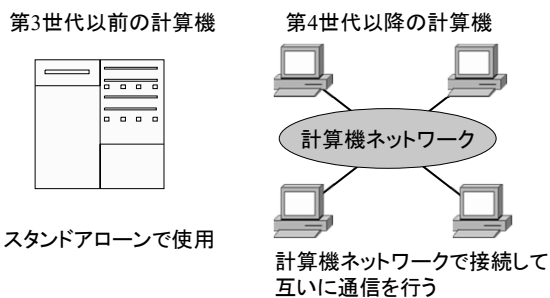
75

OSの歴史(第4世代)

- 第4世代(1970年代後半)
 - 計算機ネットワーク
 - オンライン処理
 - データベース
 - パーソナルコンピュータ
 - ワークステーション

76

OSの歴史(第4世代): 計算機ネットワーク



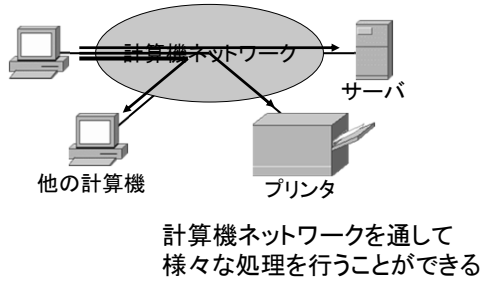
77

OSの歴史(第4世代): 計算機ネットワーク

- 1970年代初期 ハワイ大学で研究され、「Alohanet」が開発
ハワイ諸島の端末が共有無線周波数帯へ構造化アクセス
- 1972年 米Xerox社のPARCが自社PC(Alto)用に「Alto Aloha Network」
- 1973年 Alto以外のコンピュータにも対応させ、「Ethernet」と命名
- 1980年 米DEC社、米Intel社、米Xerox社の3社連合(DIX)によって、最初のイーサネット規格が発表される

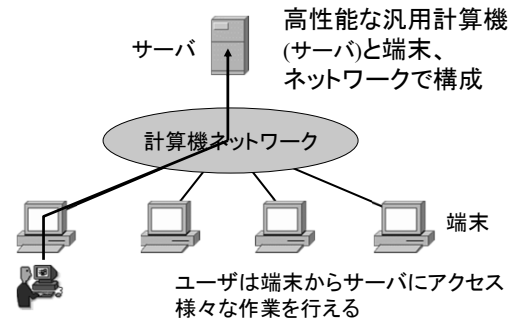
78

OSの歴史(第4世代): オンライン処理



79

OSの歴史(第4世代): ワークステーション



80

OSの歴史(第4世代): パーソナルコンピュータ

- 1975年 MITS がAltair 8800 開発
- 1977年 Apple computer がApple II 開発
- 1981年 IBMが IBM PC 5150 開発

以降様々なパーソナルコンピュータが開発される
組織に1台の時代から、個人に1台の時代へ

81

Altair 8800

- Altair 8800
 - 1974年 Micro Instrumentation and Telemetry Systems が開発
 - コンピュータキット
 - 組み立てキットで販売
 - 機体前面のスイッチでプログラム
 - CPU : Intel 8080 2MHz
 - メモリ : 256 B

82

Altair 8800



Altair 8800 [1]

[1] https://upload.wikimedia.org/wikipedia/commons/1/19/Altair_8800.jpg

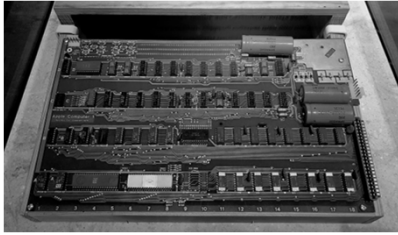
83

Apple I

- Apple I
 - 1976年 Apple が開発
 - ワンボードマイクロコンピュータ
 - キーボード等は別途必要
 - CPU : MOS 6502 1 MHz
 - メモリ : 4 KB

84

Apple I



Apple I [1]

基盤のみ モニタ・キーボードは別途必要

[1] https://upload.wikimedia.org/wikipedia/commons/1/10/Apple_1_Woz_1976_at_CHM.agr_cropped.jpg

85

Apple II

- Apple II
 - 1977年 Apple が開発
 - パーソナルコンピュータ
 - キーボードが本体に組み込まれる
 - OS : Apple DOS
 - CPU : MOS Technology 6502 1.023 MHz
 - メモリ : 4 KB

86

Apple II



本体に
キーボード

Apple II [1]

[1] https://upload.wikimedia.org/wikipedia/commons/9/98/Apple_II_typical_configuration_1977.png

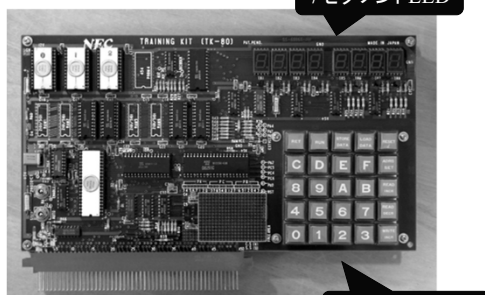
87

TK-80

- TK-80
 - 1976年 日本電気 (NEC) が開発
 - ワンボードマイクロコンピュータ
 - ボード単体で使用可能
 - 16進キーボードでプログラム
 - 7セグメントLEDで値表示
 - CPU : NEC μ PD8080A 2.048 MHz
 - メモリ : 512 B

88

TK-80



7セグメントLED

TK-80 [1]

16進キーボード

[1] <https://upload.wikimedia.org/wikipedia/commons/a/a4/TK-80.jpg>

89

PC-8001

- PC-8001
 - 1979年 日本電気 が開発
 - パーソナルコンピュータ
 - キーボードが本体に組み込まれる
 - OS : N-BASIC
 - CPU : NEC μ PD780C-1 4 MHz
 - メモリ : 16 KB

90

PC-8001



本体が
キーボードと
一体化

PC-8001 [1]

[1] https://upload.wikimedia.org/wikipedia/commons/e/e4/Nec_PC_8000_series.jpg

91

PC-9801

- PC-9801
 - 1982年 日本電気 が開発
 - パーソナルコンピュータ
 - 一時は日本国内で圧倒的シェアを占める
 - OS : MS-DOS
 - CPU : NEC μ PD8086 5 MHz
 - メモリ : 128 KB

92

PC-9801



PC-9801 [1]

[1] <https://upload.wikimedia.org/wikipedia/commons/4/4f/PC-9801-1st-001.jpg>

93

IBM PC 5150

- IBM personal computer model 5150
 - 1981年 IBM が開発
 - ビジネス用パーソナルコンピュータ
 - OS : PC DOS
 - CPU : Intel 8088 4.77 MHz
 - メモリ : 16~256 KB

94

IBM PC 5150



IBM PC 5150 [1]

[1] https://upload.wikimedia.org/wikipedia/commons/6/69/IBM_PC_5150.jpg

95

Macintosh 128K

- Macintosh 128K
 - 1984年 Apple が開発
 - 一体型パーソナルコンピュータ
 - グラフィカルユーザインタフェース
 - マウスによる操作
 - OS : System 1.0
 - CPU : MC68000 7.8336 MHz
 - メモリ : 128 KB

96



97

パーソナルコンピュータの歴史

発売年	機種	販売元	備考
1975	Altair 8800	MITS	組み立てキットで販売 スイッチでプログラム
1976	Apple I	Apple	基盤のみ
1976	TK-80	日本電気	基盤に16進キーボードと7セグメントLED 追加機器無しで使用可能
1977	Apple II	Apple	キーボードが本体に組み込み
1979	PC-8001	日本電気	日本のパソコンの先駆け
1981	IMB PC 5150	IBM	ビジネス用パソコン
1982	PC-9801	日本電気	一時は圧倒的国内シェアを占める
1984	Macintosh 128K	Apple	GUI採用 マウスで操作可能

98

計算機の世代

世代	素子	プログラミング・言語	特徴	使用者
第0世代 1940年代	機械式	ケーブルの差し入れ・機械語	黎明期 OSは存在しない	開発者
第1世代 1950年代	真空管	ケーブルの差し入れ・機械語	バッチ処理	開発者
第2世代 1960年代前半	トランジスタ	コマンドベース インタフェース・ 低水準言語	マルチプログラミング タイムシェアリング 仮想記憶 専用ソフトウェア	専門技術者
第3世代 1960年代後半	IC, LSI	GUI+ 階層メニュー・ 高水準言語	汎用大型計算機 シミュレータ, エミュレータ 汎用ソフトウェア	専門的職人
第4世代 1980年代	超LSI	GUI・ オブジェクト指向 言語	計算機ネットワーク オンライン処理 データベース パーソナルコンピュータ ワークステーション	一般人

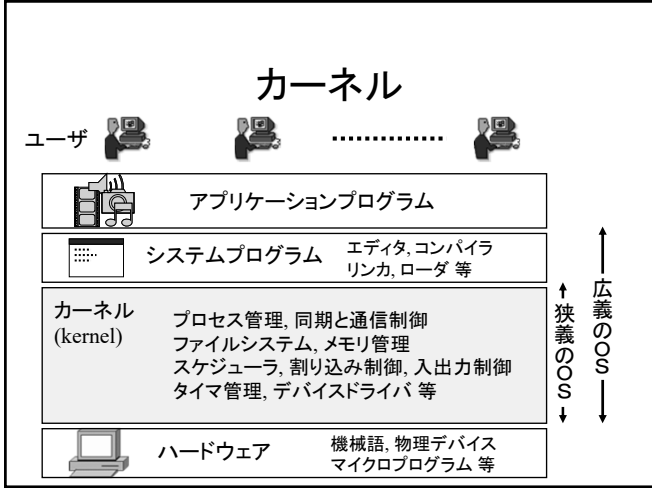
99

カーネル(kernel)

- OSの基本的なサービス
 - 資源の割付と保護
 - プログラムの実行
 - 入出力操作
 - ファイル操作

カーネル(kernel)
基本的サービスを行うOSの根幹
制御プログラムとしての役割

100



101

カーネルの特徴

- カーネルの特徴
 - 割込みにより起動
 - カーネルモード(特権モード)で動作
 - アプリケーションプログラムから記憶保護
 - プログラム実行の管理
 - 計算機資源の管理
 - 特別な特権命令でアプリケーションプログラムに戻る

102

カーネル

- カーネルが行うサービス
 - プロセス管理
 - 同期と通信制御
 - ファイルシステム
 - メモリ管理
 - スケジューラ
 - 割り込み制御
 - 入出力制御
 - タイマ制御
 - デバイスドライバ

次週以降
これらについて
順次講義します

103

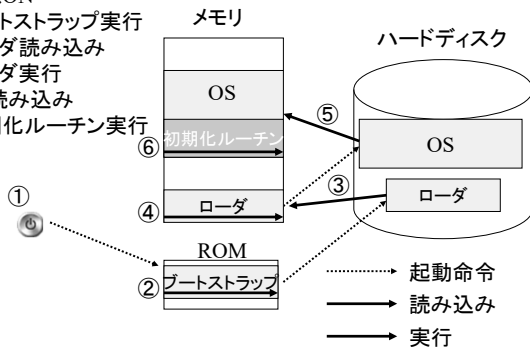
OSの起動と終了

- OSはブートストラップとローダにより起動
 - ブートストラップ(bootstrap)
 - ブートROM上に書き込まれている
 - ハードディスクからローダを読み出し、起動する
 - ローダ(loader)
 - ハードディスク上に書き込まれている
 - OSの初期化ルーチンを実行する

104

OSの起動と終了

1. 電源ON
2. ブートストラップ実行
3. ローダ読み込み
4. ローダ実行
5. OS読み込み
6. 初期化ルーチン実行



105

OSの起動と終了

- OSの終了はシャットダウン(shutdown)プログラムで実行
 - シャットダウンプログラム
 1. 新規プロセスの生成, ログインを停止
 2. 実行中のプロセスが終了したことを確認
 3. システムプログラムのファイルを閉じる
 4. 停止
 いきなり電源を切ってはいけない

106

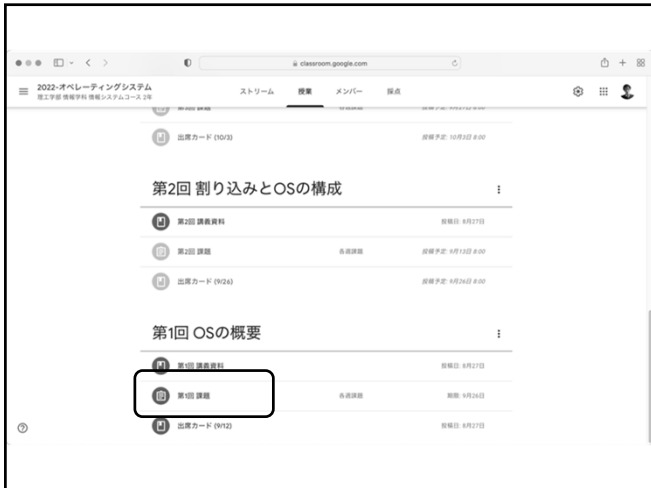
課題テスト

- 毎週 GoogleClassroom上で課題テストを行う
 - 授業後～翌週の授業開始まで
- GoogleClassroomでオペレーティングシステム
 - ⇒ 授業
 - ⇒ その回の課題と辿る

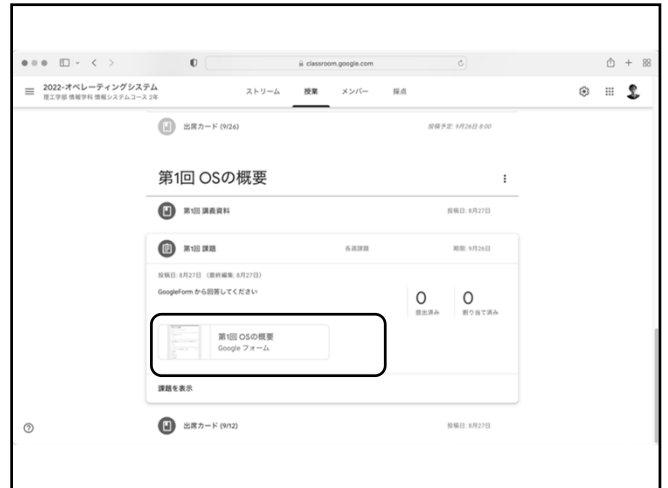
107



108



109



110



111