

## 第9章 メモリの管理

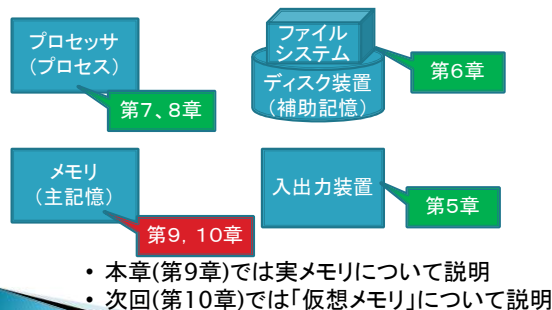
花田 英輔

(このPowerPointは渡辺名誉教授作成のものを花田が一部改題した)

1

## コンピュータの重要な資源

資源名と教科書での説明



2

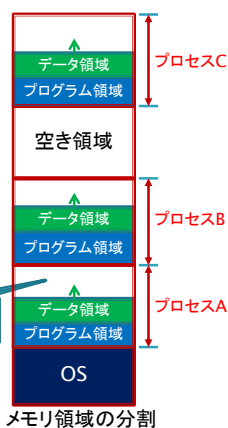
## メモリ資源(教科書9.1)

### メモリ(主記憶) = 様々な用途に分割使用

- 各利用者とOSに領域を割当て
- 1利用者の各プログラムに領域を割当て
- 1プログラムの手続き部分とデータ部分に領域を割当て

**メモリ ≠ ディスク**

ディスクは「外部記憶」もしくは「ストレージ」

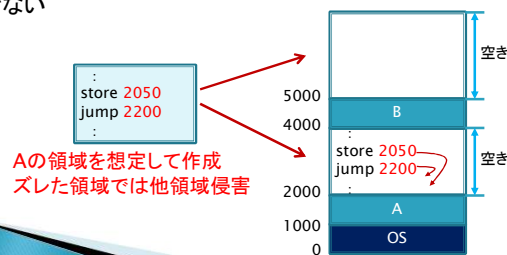
必要に応じ  
増える余地

3

## メモリへのプログラムの配置(教科書9.2)

### プログラムは、メモリ上に自由に配置したい

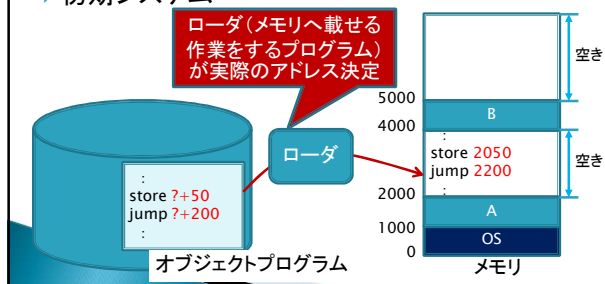
- しかしデータ取得アドレスやジャンプ先アドレスがズレ  
⇒コンパイル時に想定したアドレスからズレて配置できない



4

## 再配置ローダによる方法

- オブジェクトプログラムではアドレスは未定
- メモリへのロード時に決定
- 初期システム

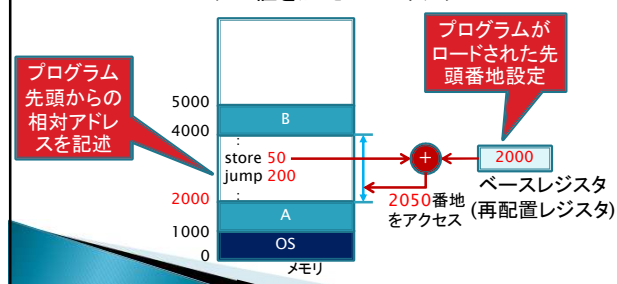


5

## 再配置レジスタによる方式

### ハードウェア変更

- ベースレジスタ(再配置レジスタ)の追加
- ベースレジスタの値を加えてメモリアクセス

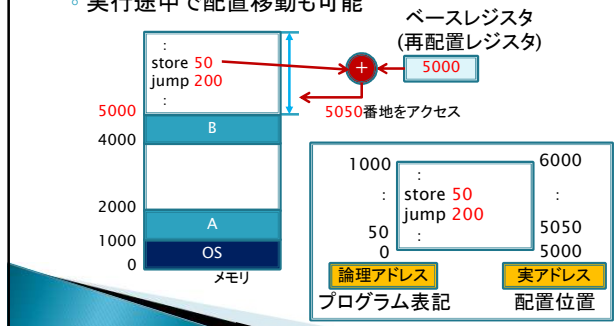


6

## 再配置レジスタによる方式

## ▶ 5000番地に載せると以下の通り

- 実行途中で配置移動も可能

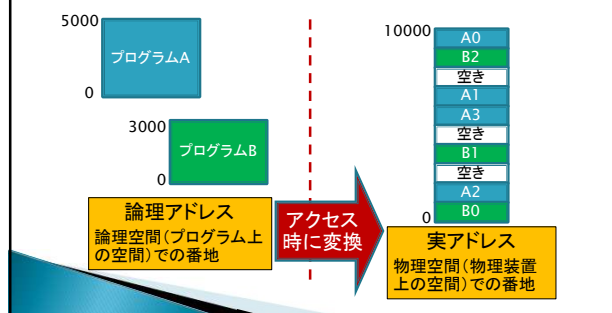


7

## 仮想記憶による方法

## ▶ 記憶装置上の扱いとプログラム上の扱いを分離

- ベースレジスタ方式の発展形、次章で説明

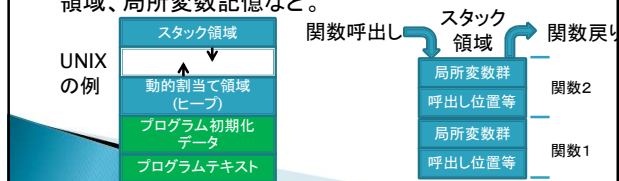


8

## プロセスのメモリ領域の管理(教科書9.3)

## ▶ プロセスごとに以下の領域が割当て

- プログラムの**手続き**部分=サイズ固定(コードなど)
- プログラムの**固定データ**部分=サイズ固定(大域変数など)
- **動的割当て**部分=サイズ可変。プログラム中で動的確保(malloc/free, new/delete)。ヒープと呼ぶ。
- **スタック**領域=サイズ可変。関数呼出し時の情報退避領域、局所変数記憶など。



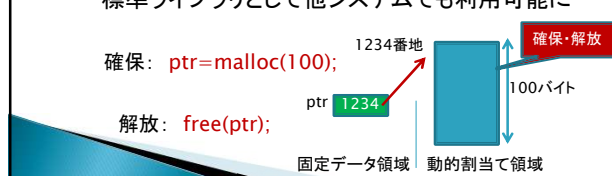
9

## メモリ領域の確保・解放機能(教科書9.4)

## ▶ 動的割当て部分のメモリ確保と解放の機構

⇒ C言語標準ライブラリ: malloc/free

- 元々はUNIX用
- 言語機能ではなくライブラリ関数の一つ (C++では、new/deleteが言語機能として実装)
- C言語標準化(ANSI/ISO)に伴い、標準ライブラリとして他システムでも利用可能に

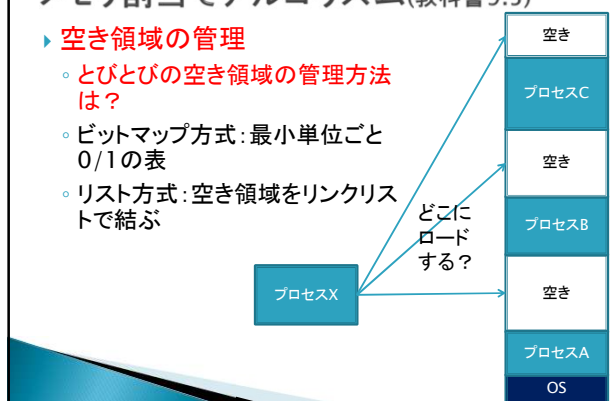


10

## メモリ割当てアルゴリズム(教科書9.5)

## ▶ 空き領域の管理

- とびとびの空き領域の管理方法は?
- ビットマップ方式: 最小単位ごと0/1の表
- リスト方式: 空き領域をリンクリストで結ぶ

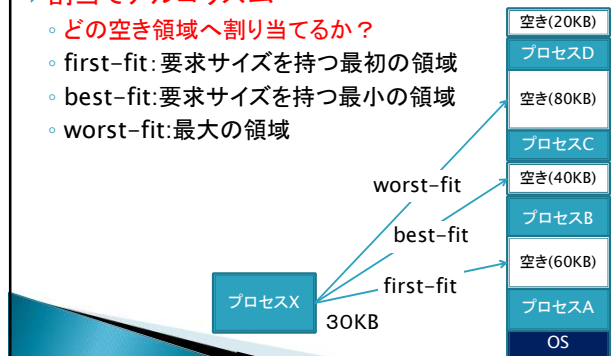


11

## メモリ割当てアルゴリズム

## ▶ 割当てアルゴリズム

- どの空き領域へ割り当てるか?
- first-fit: 要求サイズを持つ最初の領域
- best-fit: 要求サイズを持つ最小の領域
- worst-fit: 最大の領域

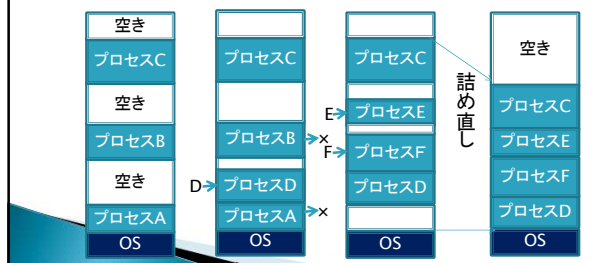


12

## 断片化

- ▶ 確保・解放を繰り返すと**小さい空き領域が多数できる**(これを「断片化」という)

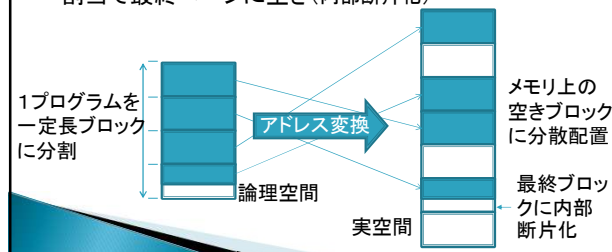
- 対策: **詰め直し**(時々停止し、割当て領域をずらしてまとめる) ⇒ **オーバーヘッドが大きくなる**



13

## ページ化による割当て

- ▶ メモリを小さい固定長ブロック(=ページ)に分け、その単位で空き領域に割当て
  - アクセス時はアドレス変換(ベースレジスタ方式類似、次章)
  - 同一サイズのため、小さい空きが分散しない
  - 割当て最終ページに空き(内部断片化)



14

## セグメント化による割当て



- セグメント: プログラム構造に依存した可変長ブロック
- ページ: プログラム構造とは独立した固定長ブロック

15

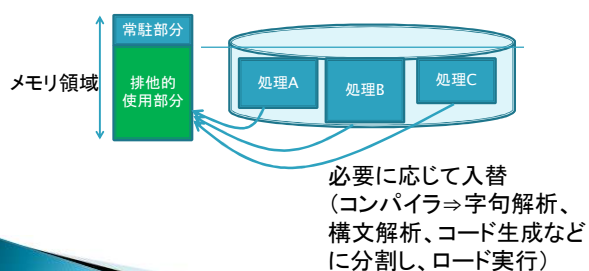
## メモリに入りきらないプログラムの実行(教科書9.6)

- ▶ メモリ容量は有限
  - ⇒ 入りきらないプログラムがあり得る
- ▶ 3つの実現方法
  1. オーバレイ方式
    - バラして実施
  2. スワッピング方式
    - 取っ替え引っ替え
  3. 仮想記憶方式
    - 秘密の場所を使用

16

## メモリに入りきらないプログラムの実行1

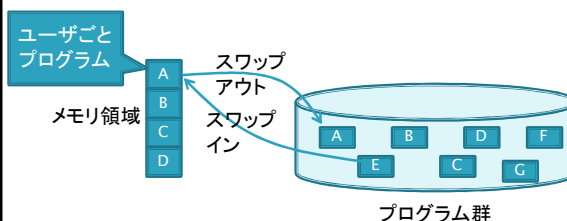
- ▶ **オーバーレイ方式**=初期システム



17

## メモリに入りきらないプログラムの実行2

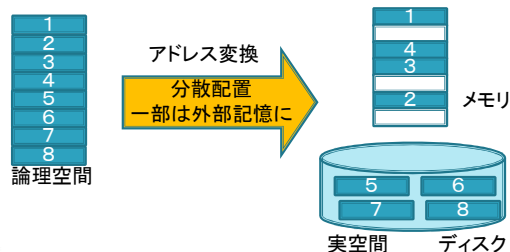
- ▶ **スワッピング方式**=TSSで使用



18

## メモリに入りきらないプログラムの実行3

- ▶ **仮想記憶方式** = 次章、乞うご期待



19

## 今回の課題

1. ベースレジスタ方式で、ベースレジスタの値が「5000」の場合について、以下の処理を実行すると実メモリ上のどのアドレスがアクセスされるかを示せ。
  2. (予習)「仮想記憶方式」とは何か、調べて記せ  
(1.の実行内容と論理メモリの状態はMoodle内ファイルを用いること)
- ▶ 今回のファイル名は“学籍番号-OS11.docx”  
(例: 24238000-OS11.docx)としてください
  - ▶ 締切: 12月26日(金) 18:00 (遅れた場合は減点)

20

## 次回の講義について

- ▶ 12月25日から1月7日までは冬休みで講義はありません
- ▶ 1月12日は**祝日**です
  - 代替日は14日なので、**次回講義は1月14日(水)の1校時**です
- ▶ 年末までに未提出課題の確認を行います
  - 未提出者リストを作成予定です
  - 今からでも間に合うので提出してください
    - ・ 課題は評価全体の20%程度を占めます
- ▶ 期末試験は2月9日になる予定です
  - 部屋が変わるようです(事前に指示します)

21