

[B2] 分割コンパイル

Copyright © 2020-2024, by Takeshi Kawabata

コンパイルとリンク

(人間) テキスト
ソースプログラム

main.c

func.c

機械語プログラムと
データ領域は、最終
的に固定したメモリ
アドレスに配置

コンパイル

(コンピュータ) バイナリ (2進数)
オブジェクトプログラム

main.o

100番地

機械語プログラム

200番地

データ記憶領域

func.o

100番地

機械語プログラム

200番地

データ記憶領域

再配置可能なアドレスを付与

リンク

実行プログラム

a.exe

100番地

機械語 (main)

200番地

データ記憶領域

300番地

機械語 (func)

400番地

データ記憶領域

実行するためにメモリの
アドレスを確定(固定)する

分割コンパイル

(人間) テキスト
ソースプログラム

(コンピュータ) バイナリ (2進数)
オブジェクトプログラム 実行プログラム

main.c

\$ gcc -c main.c

main.o

func.c

\$ gcc -c func.c

func.o

\$ gcc main.o func.o

a.exe

libc.a (ライブラリ)

(実行) \$./a

\$ gcc main.c func.c

\$ gcc main.c func.o

\$ gcc main.o func.c

\$ gcc main.c func.c -o progname.exe

prog_name.exe

コンパイル

リンク

(実行) \$./prog_name

ライブラリとヘッダファイル

- ライブラリ: 既に誰かが作ってくれた関数の集合
- 「#include <ヘッダ名.h>」記述は、プログラムのその位置に「/usr/include/ヘッダ名.h」を挿入する
- ライブラリ関数を利用するためには、対応するヘッダファイルをインクルードする必要がある

```
#include <stdio.h> /* 入出力ライブラリ */  
#include <stdlib.h> /* 標準機能ライブラリ */  
#include <string.h> /* 文字列処理ライブラリ */
```

ヘッダファイルの場所

```
#include <ヘッダ名.h>
```

- プログラムのその位置に、既に用意された「`/usr/include/ヘッダ名.h`」を挿入する

```
#include “ヘッダ名.h”
```

- プログラムのその位置に、
同じディレクトリにある「ヘッダ名.h」を挿入する
⇒ 構造体や型を自分で定義する場合

疑似乱数ライブラリ

- 乱数
 - 与えられた範囲で一様分布する確率変数の実現値
- 疑似乱数ライブラリ
 - 実用上乱数とみなしうる整数列を、コンピュータを用いて生成(引数のない関数 `int rand()`)
 - 範囲: $0 \leq \text{rand()} \leq \text{RAND_MAX}$

```
#include <stdlib.h> /* 標準機能ライブラリ */
srand(seed);        /* 疑似乱数の種を設定 */
value = rand();      /* 疑似乱数を1個生成 */
```