

# システムプログラムレポート

2024年12月6日  
学籍番号：35714121  
名前：福富隆大

1. 7-3ページのe7-1.cと7-14ページのe7-3.cをそれぞれgccでコンパイルし、オブジェクトファイルe7-1.oとe7-3.oを作成しなさい

作成したファイル内容、結果について

```
gcc -c e7-1.c
gcc -c e7-3.c
```

上記のコマンドでオブジェクトファイルを作成した。

2. e7-1.oとe7-3.oをそれぞれobjdumpコマンドで逆アセンブルし、gccがどのようなアセンブリコードを生成しているかを確認しなさい。

実行結果

e7-1.o:       ファイル形式 elf64-x86-64

セクション .text の逆アセンブル:

```
0000000000000000 <max1>:
  0:  f3 0f 1e fa          endbr64
  4:  55                   push   %rbp
  5:  48 89 e5             mov    %rsp,%rbp
  8:  8b 05 00 00 00 00    mov    0x0(%rip),%eax          # e
<max1+0xe>
 e:  89 05 00 00 00 00    mov    %eax,0x0(%rip)          # 14
<max1+0x14>
14:  8b 15 00 00 00 00    mov    0x0(%rip),%edx          # 1a
<max1+0x1a>
1a:  8b 05 00 00 00 00    mov    0x0(%rip),%eax          # 20
<max1+0x20>
20:  39 c2               cmp    %eax,%edx
22:  7e 0d             jle    31 <max1+0x31>
24:  8b 05 00 00 00 00    mov    0x0(%rip),%eax          # 2a
<max1+0x2a>
2a:  89 05 00 00 00 00    mov    %eax,0x0(%rip)          # 30
<max1+0x30>
30:  90                   nop
31:  90                   nop
```

32:	5d	pop	%rbp
33:	c3	ret	

e7-3.o:        ファイル形式 elf64-x86-64

セクション .text の逆アセンブル:

```
0000000000000000 <max2>:
  0:  f3 0f 1e fa      endbr64
  4:  55               push  %rbp
  5:  48 89 e5         mov   %rsp,%rbp
  8:  8b 05 00 00 00 00 mov   0x0(%rip),%eax    # e
<max2+0xe>
  e:  89 05 00 00 00 00 mov   %eax,0x0(%rip)    # 14
<max2+0x14>
 14:  8b 15 00 00 00 00 mov   0x0(%rip),%edx    # 1a
<max2+0x1a>
 1a:  8b 05 00 00 00 00 mov   0x0(%rip),%eax    # 20
<max2+0x20>
 20:  39 c2           cmp   %eax,%edx
 22:  7e 0e          jle   32 <max2+0x32>
 24:  8b 05 00 00 00 00 mov   0x0(%rip),%eax    # 2a
<max2+0x2a>
 2a:  89 05 00 00 00 00 mov   %eax,0x0(%rip)    # 30
<max2+0x30>
 30:  eb 01          jmp   33 <max2+0x33>
 32:  90             nop
 33:  90             nop
 34:  5d             pop   %rbp
 35:  c3             ret
```

作成したファイル内容、結果について

max2はgotoを使っているのでアセンブリコードにjmp命令があるが、max1はgotoを使っていないのでない。行数的にはmax2の方が多くなったが、アセンブリする時の実行時間は圧倒的にmax2の方が早かった。

講義に対する感想・質問・意見

今回の内容は普段よく使う条件分岐だったので比較的分かりやすかった。OSやコンピュータアーキテクチャでやったパイプライン処理の話も出てきて、他の授業の知識と繋げることができ面白かった。