# システムプログラム第2回レポート

2024年10月8日

学籍番号:35714121

名前:福富隆大

① 講義資料内のe2-4m.c, e2-4.cの2つのファイルを作成して分割でコンパイルし,これらをリンクさせて実行ファイルを作成しなさい。

#### 実行結果

スライドの通りの以下の手順でコンパイルして実行した。

```
gcc -c e2-4m.c -o e2-4m.o
gcc -c e2-4.c -o e2-4.o
gcc e2-4m.o e2-4.o -o e2-4-x64.ex
./e2-4-x64.exe 3 5
```

#### 結果は以下のようになった。

```
v1=3 v2=5
v1=3 v2=5 v3=8
```

### 画像の下の部分のターミナルに実行結果あります

```
エクスプローラー

> ソース管理リポジトリ
                                                                                                                                            35714121.pdf M
                                                                                                                                                                                                                                                                                        System Program > 02 > C e2-4m.c > main(int, char * [])
      univer... ⋧ n ↔ ✓ ♡ 👯 …
         ~ ソース管理
        > 変更

■ 357 19121.put System Fr... M
                                                                #Include <stdloin.h>
int v1,v2,v3;

void sum(void);
int main(int argc, char *argv[]){
    if (argcl=3){
        printf("ERR: enter %s v1 v2 \n",argv[0]);
        return 1;
    }
          C e2-4.c System Program/02 M
C e2-4m.c Syst... The Market Syst...
         e2-4m.o System Progra... M
                                                                   v1=atoi(argv[1]);
        V UNIVERSITYLECTURE
                                                                  v2=atol(agv[1]);
v2=atol(agv[1]);
printf("v1=%d v2=%d \n", v1, v2);
sum();
printf("v1=%d v2=%d v3=%d \n", v1, v2, v3);
          > programming1
           > programming2
           > programming3
           v 😇 System Program
            > 🖿 01
                e2-4-x64.exe
e2-4.c
                 > ■ 03
             > • 04
           問題 12 出力 <u>ターミナル</u> ポート コメント
                                                                                                                                                                                                                                                                     ∑ zsh - 02 + ∨ □ 🛍 🚍
             ● (base) Fukutomi®dS-MocSook-Pro @2 % ./e2-4-x64.exe 3 5

√1-3 √2-5 √2-5

√2-5 √2-5 √2-5

( base) Fukutomi®dS-MocSook-Pro @2 % ■

in* → ② ② ② △ 12 ½ ② Git Graph 44 Words, 338 Chars, 17 Lines, 1 Paragraph, ~0m13s re
```

#### 作成したファイル内容

```
// e2-4m.c
#include <stdio.h>
#include <stdlib.h>
int v1, v2, v3;
void sum(void);
int main(int argc, char *argv[])
{
if (argc!=3){
printf("ERR: enter %s v1 v2\forall n", argv[0]);
return 1;}
v1=atoi(argv[1]);
v2=atoi(argv[2]);
printf("v1=%d v2=%d\u00e4n",v1,v2);
printf("v1=%d v2=%d v3=%d\u00e4n",v1,v2,v3);
return 0;
}
```

```
//e2-4.c
extern int v1,v2,v3;
void sum(void)
{
    v3=v1+v2;
}
```

② 上記を基に、四則演算を行うプログラムを作成する。演算ごとに個別ファイルとして作成し、①と同様に実行ファイルを作成しなさい

#### 実行結果

四則演算をするコードをそれぞれファイルを分けて書いた。 スライドの例ではexternを使っていたが、今回は関数プロトタイプ宣言を用いている。 ファイルが多いので以下のようにまとめてコンパイルした。

```
gcc -c add.c subtract.c multiply.c divide.c main.c
gcc -o calculator main.o add.o subtract.o multiply.o divide.o
./calculator
```

結果は以下のようになった(3と5はキーボード入力してます)

```
2つの数を入力してください: 3 5
加算: 8
減算: -2
```

乗算: 15 除算: 0.60

## 画像の下の部分のターミナルに実行結果あります

```
    □ universityLecture

                                                                                                                                                                                                                                                         # ## □ ···
~ ソース管理リポジトリ
int add(int-a, int-b);
int subtract(int-a, int-b);
int multiply(int-a, int-b);
float-divide(int-a, int-b);
        ✓ 変更

© e2-4-x64.exe System Pr... M
                                                         C main.c System... ** カナリ
        〉開いているエディター
       V UNIVERSITYLECTURE
#
                                                              printf("加算: %d\n", add(num1, num2));
printf("流算: %d\n", subtract(num1, num2));
printf("崇算: %d\n", multiply(num1, num2));
printf("除算: %.2f\n", divide(num1, num2));
           calculator divide.c
             ♂ divide.o□ e2-4-x64.exeC e2-4.c
              😭 e2-4.o
              main.o
               multiply.o

☆ subtract.o

♪ 実行結果 1.png

             03
           > • 04
        > アウトライン
         問題 11 出力 ターミナル ポート コメント
      (base) futuation(例3-MacBook-Pro 02 % ./calculator

つの数を入力してください: 3 5

加算: -2

実算: -5

険算: 0.50

(base) / futuation(例3-MacBook-Pro 02 % []
```

#### 作成したファイル内容

```
//main.c
#include <stdio.h>
int add(int a, int b);
int subtract(int a, int b);
int multiply(int a, int b);
float divide(int a, int b);
int main() {
    int num1, num2;
    printf("2つの数を入力してください: ");
    scanf("%d %d", &num1, &num2);
    printf("加算: %d\n", add(num1, num2));
    printf("減算: %d\n", subtract(num1, num2));
    printf("乗算: %d\n", multiply(num1, num2));
    printf("除算: %.2f\n", divide(num1, num2));
    return 0;
}
```

```
//add.c
#include <stdio.h>
int add(int a, int b) {
   return a + b;
}
```

```
//subtract.c
#include <stdio.h>
int subtract(int a, int b) {
   return a - b;
}
```

```
//multiply.c
#include <stdio.h>
int multiply(int a, int b) {
   return a * b;
}
```

```
//divide.c
#include <stdio.h>
float divide(int a, int b) {
    if (b != 0) {
        return (float)a / b;
    } else {
        printf("エラー: 0で割ることはできません。\n");
        return 0;
    }
}
```

## 講義に対する感想・質問・意見

今までは高水準言語のプログラミング言語と低水準言語のアセンブリ言語を別々に勉強してきたので、この講義で二つを結びつけて勉強していきたいです。

また、アセンブリコードについてもより詳しく理解していきたいです。