

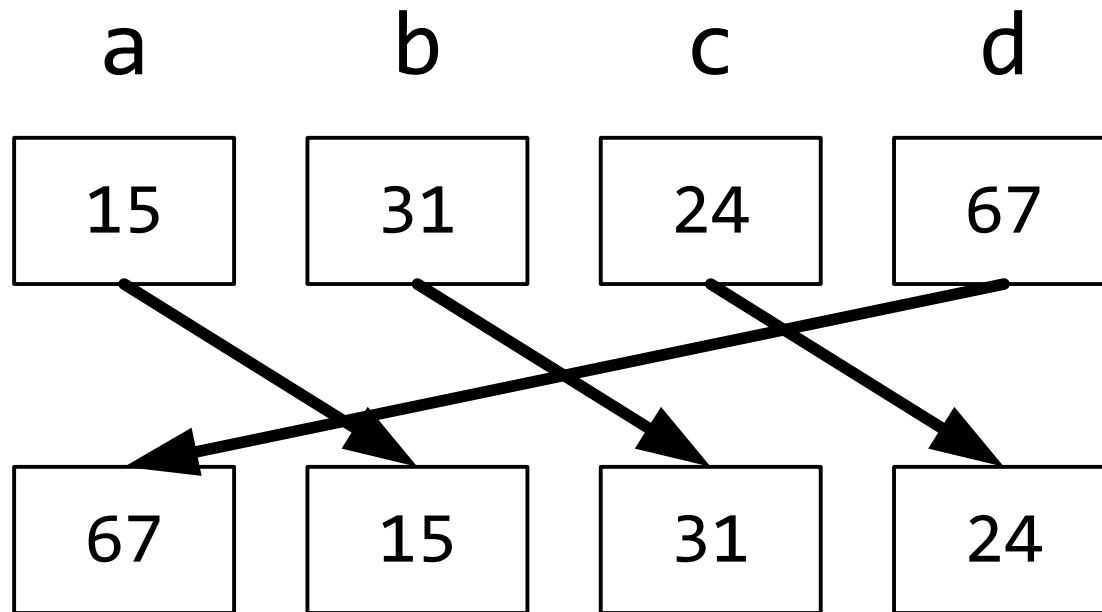
# Cプログラミング入門

## (基幹5クラス)

第11回 ポインタ, 演習

## 演習①

- ▶ 端末から入力した4つの整数を**シフト**して標準出力するプログラム pfunc4.c を作成せよ。



(シフトの様子)

# 演習①

---

```
#include <stdio.h>

void rot1(int *pa, int *pb, int *pc, int *pd){
    

?

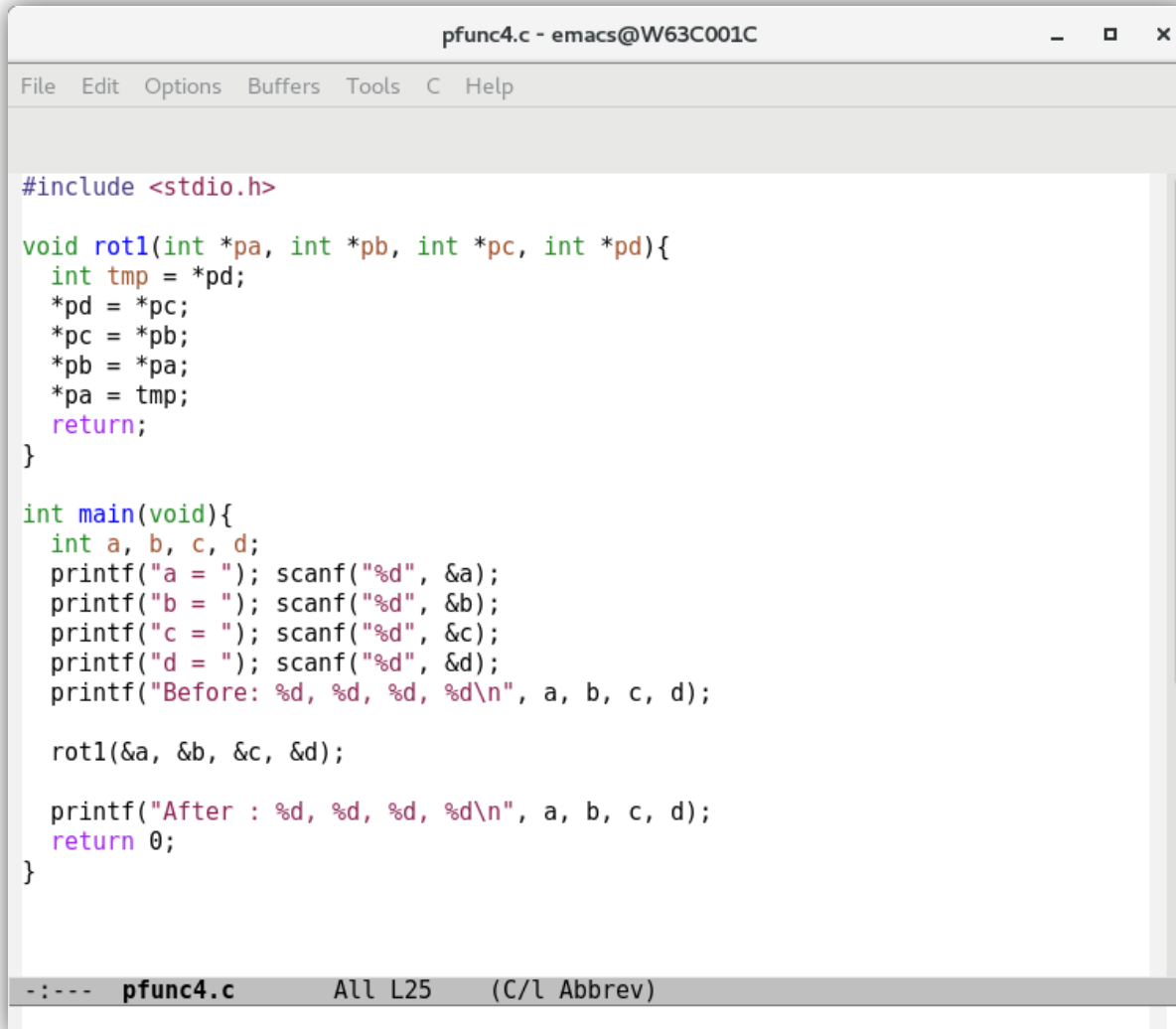

    return;
}

int main(void){
    int a, b, c, d;
    printf("a = "); scanf("%d", &a);
    printf("b = "); scanf("%d", &b);
    printf("c = "); scanf("%d", &c);
    printf("d = "); scanf("%d", &d);
    printf("Before: %d, %d, %d, %d\n", a, b, c, d);

    rot1(&a, &b, &c, &d);

    printf("After : %d, %d, %d, %d\n", a, b, c, d);
    return 0;
}
```

# 演習① ~プログラム例~



```
pfunc4.c - emacs@W63C001C
File Edit Options Buffers Tools C Help

#include <stdio.h>

void rotl(int *pa, int *pb, int *pc, int *pd){
    int tmp = *pd;
    *pd = *pc;
    *pc = *pb;
    *pb = *pa;
    *pa = tmp;
    return;
}

int main(void){
    int a, b, c, d;
    printf("a = "); scanf("%d", &a);
    printf("b = "); scanf("%d", &b);
    printf("c = "); scanf("%d", &c);
    printf("d = "); scanf("%d", &d);
    printf("Before: %d, %d, %d, %d\n", a, b, c, d);

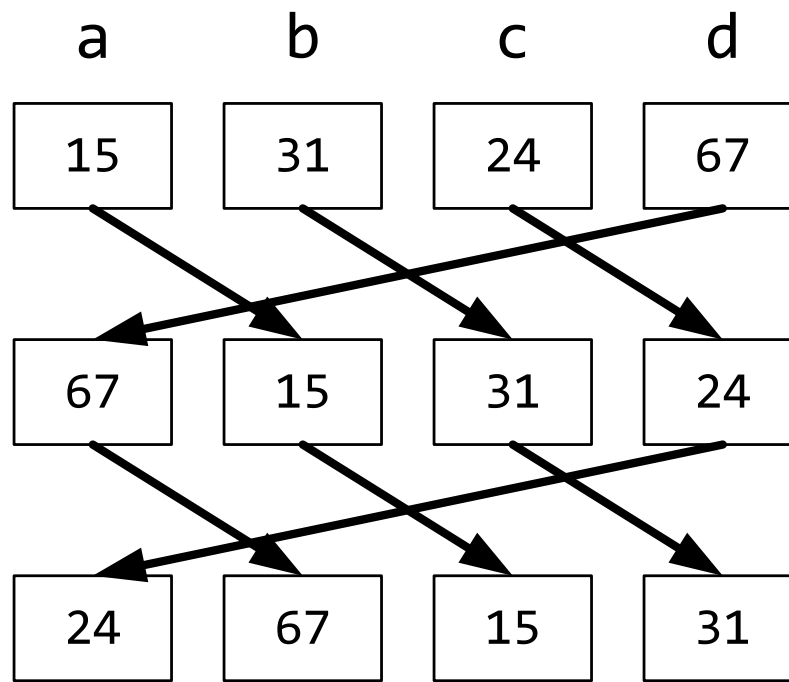
    rotl(&a, &b, &c, &d);

    printf("After : %d, %d, %d, %d\n", a, b, c, d);
    return 0;
}

-:--- pfunc4.c      All L25      (C/l Abbrev)
```

## 演習②

- ▶ 端末から入力した4つの整数を**N回シフトして標準出力**するプログラム pfunc5.c を作成せよ。
- ▶ ただし, 回数Nも端末から入力するものとする。



(例) **N=2**の場合.

## 演習②

```
#include <stdio.h>
```

```
void rot1(int *pa, int *pb, int *pc, int *pd){  
    ...  
    ...  
}
```

} 演習①を流用

```
void rotN(int *pa, int *pb, int *pc, int *pd, int N){
```

?

} 複数回rot1を呼び出す  
ように記述

```
    return;  
}
```

```
int main(void){  
    int a, b, c, d, N;  
    printf("a = "); scanf("%d", &a);  
    printf("b = "); scanf("%d", &b);  
    printf("c = "); scanf("%d", &c);  
    printf("d = "); scanf("%d", &d);  
    printf("Before: %d, %d, %d, %d\n", a, b, c, d);  
    printf("N = "); scanf("%d", &N);  
  
    rotN(&a, &b, &c, &d, N);  
  
    printf("After : %d, %d, %d, %d\n", a, b, c, d);  
    return 0;  
}
```

## 演習② ~プログラム例~

```
pfunc5.c - emacs@W63C001C
File Edit Options Buffers Tools C Help

#include <stdio.h>

void rot1(int *pa, int *pb, int *pc, int *pd){
    int tmp = *pd;
    *pd = *pc;
    *pc = *pb;
    *pb = *pa;
    *pa = tmp;
    return;
}

void rotN(int *pa, int *pb, int *pc, int *pd, int N){
    int i;
    for(i = 0 ; i < N ; i++){
        rot1(pa, pb, pc, pd);
    }
    return;
}

int main(void){
    int a, b, c, d, N;
    printf("a = "); scanf("%d", &a);
    printf("b = "); scanf("%d", &b);
    printf("c = "); scanf("%d", &c);
    printf("d = "); scanf("%d", &d);
    printf("Before: %d, %d, %d, %d\n", a, b, c, d);
    printf("N = "); scanf("%d", &N);
    -:- pfunc5.c      Top L19      (C/l Abbrev)
```

## 演習③

- ▶ 端末から入力した整数a, bをもとに**a÷bの商と余り**を標準出力するプログラム pfunc6.c を作成せよ。

```
#include <stdio.h>

void div(int a, int b, int *pq, int *pr){
    

?


    return;
}

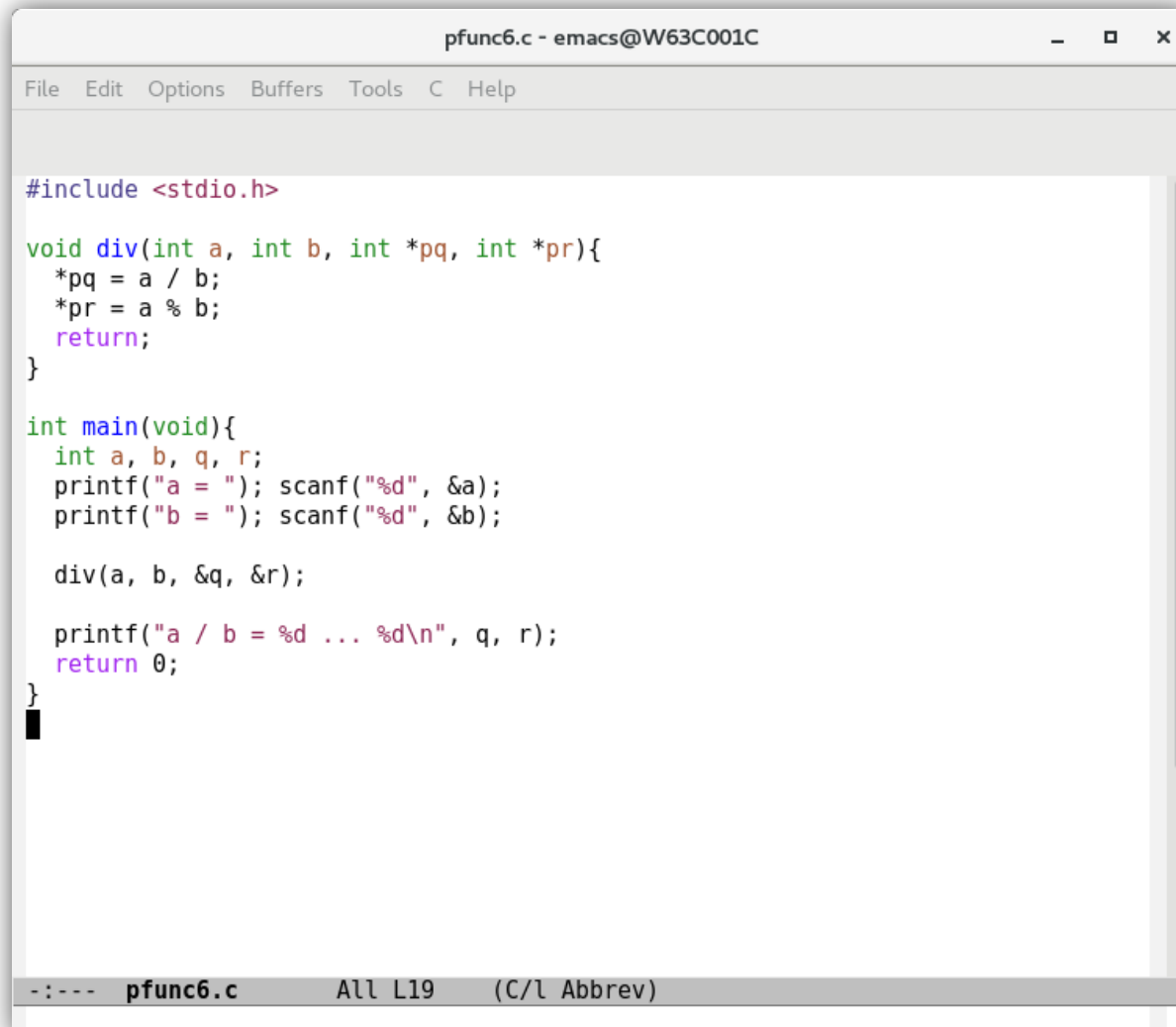
int main(void){
    int a, b, q, r;
    printf("a = "); scanf("%d", &a);
    printf("b = "); scanf("%d", &b);

    div(a, b, &q, &r);

    printf("a / b = %d ... %d¥n", q, r);
    return 0;
}
```



## 演習③ ~プログラム例~



```
pfunc6.c - emacs@W63C001C
File Edit Options Buffers Tools C Help

#include <stdio.h>

void div(int a, int b, int *pq, int *pr){
    *pq = a / b;
    *pr = a % b;
    return;
}

int main(void){
    int a, b, q, r;
    printf("a = "); scanf("%d", &a);
    printf("b = "); scanf("%d", &b);

    div(a, b, &q, &r);

    printf("a / b = %d ... %d\n", q, r);
    return 0;
}
```

-:--- pfunc6.c All L19 (C/l Abbrev)

## 演習④

- ▶ 配列の各要素を2倍に変更する関数 twice を作成せよ。
- ▶ プログラム名：parray1.c

```
#include <stdio.h>
```

```
void twice(int *a){
```

?

```
    return;
```

```
}
```

```
int main(void){
```

```
    int i, a[5] = {11, 22, 33, 44, 55};
```

```
    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
```

```
    printf("\n");
```

```
    twice(a);
```

```
    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
```

```
    printf("\n");
```

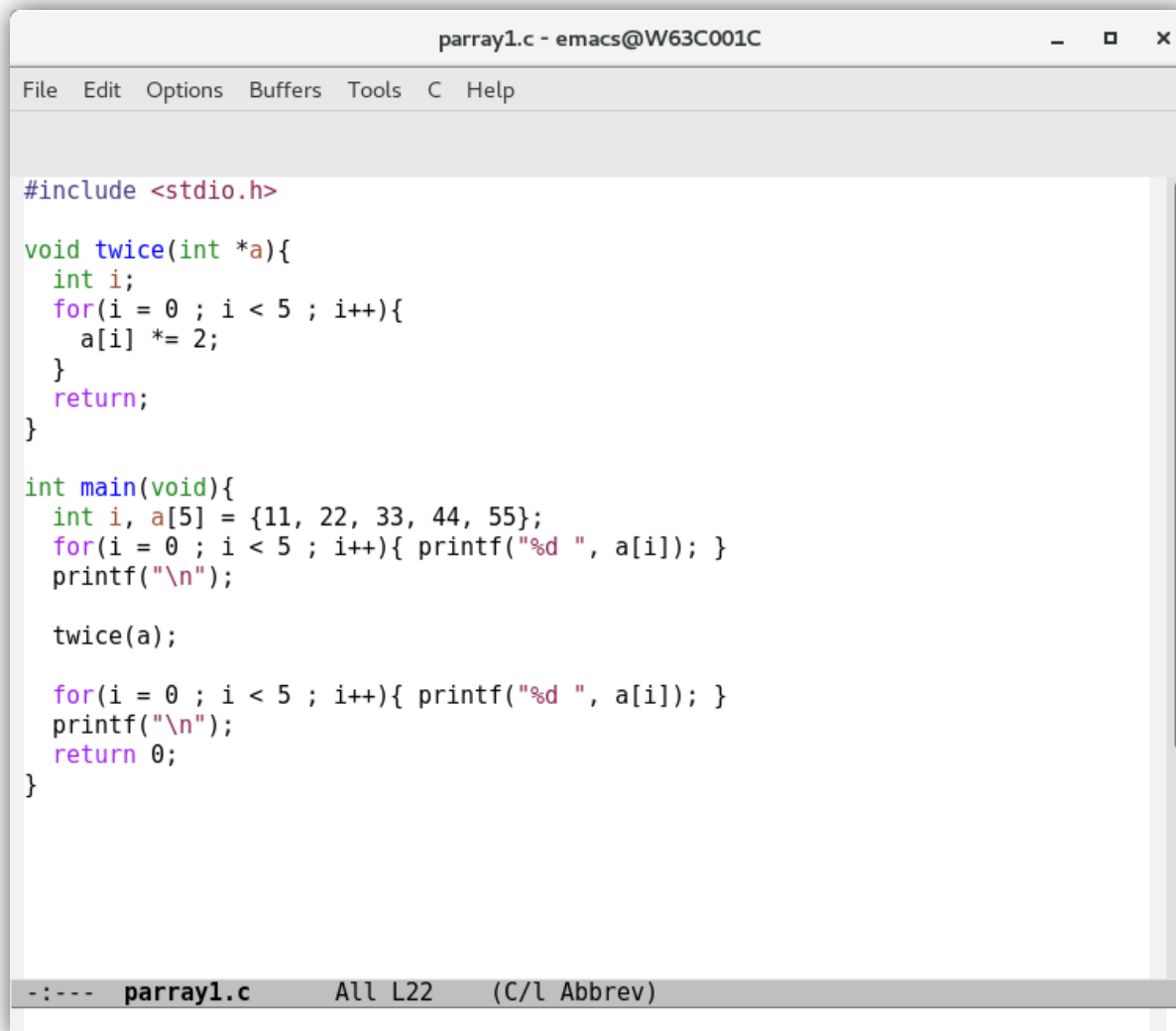
```
    return 0;
```

```
}
```

要素数は5で固定



## 演習④ ~プログラム例~



```
parray1.c - emacs@W63C001C
File Edit Options Buffers Tools C Help

#include <stdio.h>

void twice(int *a){
    int i;
    for(i = 0 ; i < 5 ; i++){
        a[i] *= 2;
    }
    return;
}

int main(void){
    int i, a[5] = {11, 22, 33, 44, 55};
    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
    printf("\n");

    twice(a);

    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
    printf("\n");
    return 0;
}

-:--- parray1.c    All L22    (C/l Abbrev)
```

## 演習⑤

- ▶ 配列の要素を逆順に変更する関数 reverse を作成せよ。
- ▶ プログラム名：parray2.c

```
#include <stdio.h>
```

```
void reverse(int *a){
```

?

```
    return;
```

```
}
```

```
int main(void){
```

```
    int i, a[5] = {11, 22, 33, 44, 55};
```

```
    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
```

```
    printf("\n");
```

```
    reverse(a);
```

```
    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
```

```
    printf("\n");
```

```
    return 0;
```

```
}
```

要素数は5で固定



## 演習⑤ ~プログラム例~



The screenshot shows an Emacs editor window titled "parray2.c - emacs@W63C001C". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "C", and "Help". The code is as follows:

```
#include <stdio.h>

void reverse(int *a){
    int i, b[5];
    for(i = 0 ; i < 5 ; i++){
        b[i] = a[i];
    }
    for(i = 0 ; i < 5 ; i++){
        a[i] = b[4-i];
    }
    return;
}

int main(void){
    int i, a[5] = {11, 22, 33, 44, 55};
    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
    printf("\n");

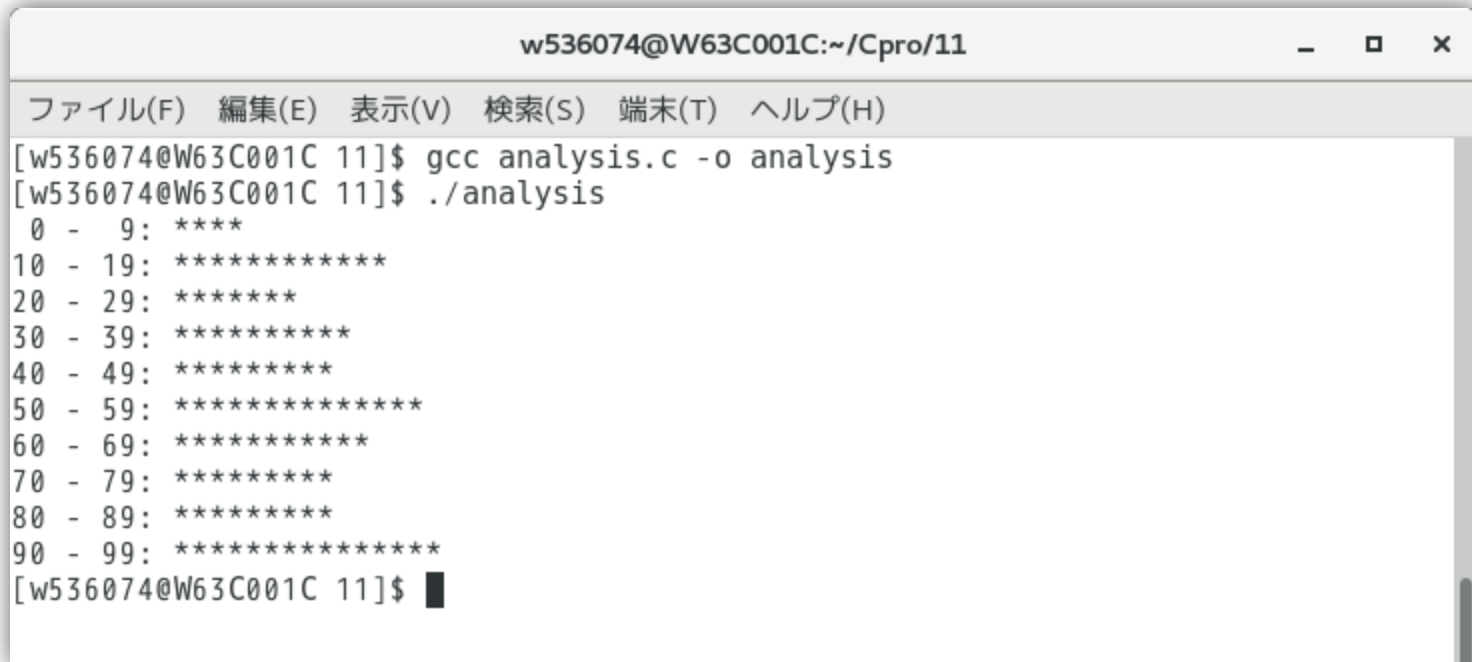
    reverse(a);

    for(i = 0 ; i < 5 ; i++){ printf("%d ", a[i]); }
    printf("\n");
    return 0;
}
```

The status bar at the bottom shows "-:--- parray2.c All L25 (C/l Abbrev)".

## 演習⑥（余力がある人向け）

- ▶ 0~99の乱数を100個生成し，10段階のヒストグラムを表示するプログラム analysis.c を作成せよ。
- ▶ 実行例




```
w536074@W63C001C:~/Cpro/11
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C001C 11]$ gcc analysis.c -o analysis
[w536074@W63C001C 11]$ ./analysis
 0 - 9: ****
10 - 19: *****
20 - 29: *****
30 - 39: *****
40 - 49: *****
50 - 59: *****
60 - 69: *****
70 - 79: *****
80 - 89: *****
90 - 99: *****
[w536074@W63C001C 11]$
```

## 演習⑥（余力がある人向け）

### ▶ プログラムの一部

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define SIZE 100
```

二つの関数を  
自作する



```
void analyze_data(int *data, int *score, int width){ ... }
void print_score(int *score, int num){ ... }
```

```
int main(void){
    srand((unsigned)time(NULL));
    int i, data[SIZE];
    for(i = 0 ; i < SIZE ; i++){
        data[i] = rand() % 100;
    }
    int score[10];
    for(i = 0 ; i < 10 ; i++){ score[i] = 0; }

    analyze_data(data, score, 10);
    print_score(score, 10);
    return 0;
}
```

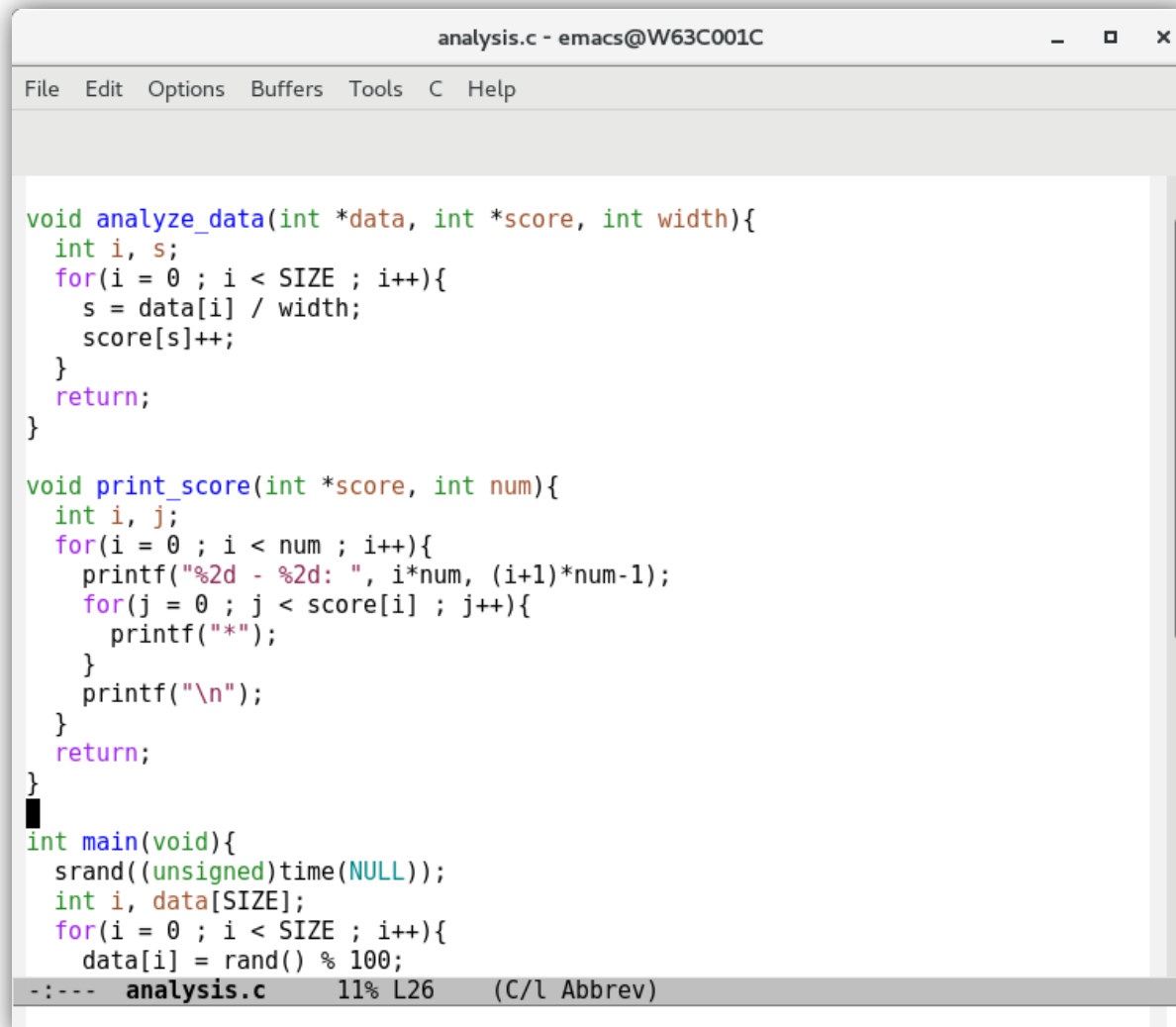
## 演習⑥（余力がある人向け）

---

- ▶ `analyze_data(int *data, int *score, int width){ ... }`
  - ▶ データを解析する関数
  - ▶ `score[0]`, `score[1]`, ..., `score[9]`を計算する
  - ▶ `score[0]` → 0~9の範囲のデータ個数（階級0）
  - ▶ `score[1]` → 10~19の範囲のデータ個数（階級1）
  - ▶ `width` → 階級の幅（10）
- ▶ `print_score(int *score, int num){ ... }`
  - ▶ 解析結果を表示する関数
  - ▶ `num` → 階級数（階級0～階級9）



## 演習⑥ ~プログラム例~



```
analysis.c - emacs@W63C001C
File Edit Options Buffers Tools C Help

void analyze_data(int *data, int *score, int width){
    int i, s;
    for(i = 0 ; i < SIZE ; i++){
        s = data[i] / width;
        score[s]++;
    }
    return;
}

void print_score(int *score, int num){
    int i, j;
    for(i = 0 ; i < num ; i++){
        printf("%2d - %2d: ", i*num, (i+1)*num-1);
        for(j = 0 ; j < score[i] ; j++){
            printf("*");
        }
        printf("\n");
    }
    return;
}

int main(void){
    srand((unsigned)time(NULL));
    int i, data[SIZE];
    for(i = 0 ; i < SIZE ; i++){
        data[i] = rand() % 100;
    }
}

-:--- analysis.c 11% L26 (C/l Abbrev)
```