

課題 4-1~4-5 のプログラム例

```
#include <stdio.h>

/* 関数のプロトタイプ宣言 */
void show_range2(int *ptr, int s, int e);
int sum_range(int *ptr, int s, int e);
int find_max(int *ptr, int s, int e);
void check_even(int *ptr);
void check_bigger(int *ptr1, int *ptr2);
void reverse(int *ptr1, int *ptr2);

/* 指定した範囲内の配列の要素を出力する */
void show_range2(int *ptr, int s, int e)
{
    int i;
    for(i=s; i<=e; i++) {
        printf("*ptr+%d: %d, ptr+%d: %p\n",
            i, *(ptr+i), i, ptr+i);
    }
}

/* 指定した範囲内の配列の要素の合計を求める */
int sum_range(int *ptr, int s, int e)
{
    int i, result;
    result = 0;
    /* s 番目から e 番目までを繰り返し加算する */
    for(i=s; i<=e; i++) {
        result += *(ptr+i);
    }
    return result;
}

/* 指定した範囲内の配列の要素の最大値を求める */
int find_max(int *ptr, int s, int e)
{
    int i, result;
    /* 最大値の初期値を代入する */
    result = *(ptr+s);
    /* s 番目から e 番目までを繰り返し比較する */
    for(i=s+1; i<=e; i++) {
        /* これまでの最大値より大きい場合は
           最大値を更新する */
        if(result < *(ptr+i)) {
            result = *(ptr+i);
        }
    }
    return result;
}
```

```
/* 配列の要素が偶数かどうかを判定する */
void check_even(int *ptr)
{
    int i;
    /* 各要素を繰り返し判定する */
    for(i=0; i<=5; i++) {
        /* 偶数でなかった場合は
           ptr+i の参照先に 0 を代入する */
        if(*(ptr+i) % 2 != 0) {
            *(ptr+i) = 0;
        }
    }
}

/* 2つの配列の要素の大きさを判定する */
void check_bigger(int *ptr1, int *ptr2)
{
    int i;
    for(i=0; i<=5; i++) {
        /* 2つの要素を比較し、
           比較結果に応じて 1 か 2 を代入する */
        if(*(ptr1+i) > *(ptr2+i)) {
            *(ptr1+i) = 1;
        } else {
            *(ptr1+i) = 2;
        }
    }
}

/* 配列の要素を逆順にする */
void reverse(int *ptr1, int *ptr2)
{
    int i;
    /* ptr1 の要素を ptr2 の後ろから代入する */
    for(i=0; i<=5; i++) {
        *(ptr2+4-i) = *(ptr1+i);
    }
}

int main(void)
{
    int test[5] = {80, 60, 55, 22, 75};
    int test2[5] = {76, 85, 47, 92, 68};
    int ans[5] = {0, 0, 0, 0, 0};
    int a1[5] = {8, 4, 2, 9, 0};
    int a2[5] = {5, 6, 1, 3, 7};
    int a3[5] = {1, 4, 2, 5, 3};
    float value[5] = {12.5, 18.1, 13.2, 17.8, 15.0};
    int round[5];
```

```
/* sum_range() の動作確認 */
printf("sum_range: %d\n",
       sum_range(test, 1, 3));
printf("sum_range: %d\n",
       sum_range(test2, 2, 4));
/* find_max() の動作確認 */
printf("find_max: %d\n",
       find_max(test, 0, 4));
printf("find_max: %d\n",
       find_max(test2, 0, 2));
/* check_even() の動作確認 */
check_even(test);
show_range2(test, 0, 4);

check_even(test2);
show_range2(test2, 0, 4);
/* check_bigger() の動作確認 */
check_bigger(a1, a2);
show_range2(a1, 0, 4);
/* reverse() の動作確認 */
reverse(a2, ans);
show_range2(ans, 0, 4);
reverse(a3, ans);
show_range2(ans, 0, 4);

return 0;
}
```