

アルゴリズムとデータ構造 授業中練習問題12

次のプログラムは「構造体のクイックソートの実現例」(教科書の List6-9 を構造体版に拡張)である。このプログラムに関して、以下の問いに答えなさい。さらに、このプログラムを入力し、自分のパソコンでコンパイル、実行できることを確認してください。なお、プログラムの日本語部分は、英語、ローマ字に変更してかまいません。

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

#define swap(type, x, y) do {type t; t = x; x = y; y = t;} while(0)

/*--- 会員データ ---*/
typedef struct {
    int no; /* 番号 */
    char name[20]; /* 氏名 */
} Member;

/*--- 会員の番号の昇順比較関数 ---*/
int AscendingMemberNoCmp(const Member *x, const Member *y) {
    return x->no < y->no ? -1: x->no > y->no ? 1: 0;
}

/*--- 会員の番号の降順比較関数 ---*/
int DescendingMemberNoCmp(const Member *x, const Member *y) {
    return x->no < y->no ? 1: x->no > y->no ? -1: 0;
}

/*--- 会員の氏名の昇順比較関数 ---*/
int AscendingMemberNameCmp(const Member *x, const Member *y) {
    return strcmp(x->name, y->name);
}

/*--- 会員の氏名の降順比較関数 ---*/
int DescendingMemberNameCmp(const Member *x, const Member *y) {
    return strcmp(y->name, x->name);
}

/*--- 会員データ (番号と氏名) の表示 (改行あり) ---*/
void PrintLnMember(const Member *x) {
    printf("%d %s\n", x->no, x->name);
}

/*--- 全データの表示 ---*/
void Print(const Member *data, int n) {
    int i;

    for(i=0; i < n; i++)
        PrintLnMember(data+i);
}

/* --- クイックソート --- */
void quick(Member *a, int left, int right,
           int compare(const Member *y, const Member *z)) {
    int pl = left;
    int pr = right;
    Member x = a[(pl+pr)/2];

    do {
        while(compare(&x, a+pl)>0) pl++;
        while(compare(a+pr, &x)>0) pr--;
        if ( pl <= pr ) {
            swap(Member, a[pl], a[pr]);
            pl++;
        }
    } while (pl < pr);
}
```

```

    pr--;
}
} while(pl <= pr );
←
if ( left < pr ) quick(a, left, pr,    compare);    ①
if ( pl < right) quick(a, pl,    right, compare);
}

/*--- メニュー ---*/
typedef enum {
    TERMINATE, ASCEND_NO, ASCEND_NAME,
    DESCEND_NO, DESCEND_NAME, PRINT_ALL
} Menu;

/*--- メニュー選択 ---*/
Menu SelectMenu(void) {
    int i, ch;
    char *mstring[] = {
        "番号で昇順ソート", "名前で昇順ソート",
        "番号で降順ソート", "名前で降順ソート",
        "データを表示"
    };
    do {
        for (i = TERMINATE; i < PRINT_ALL; i++) {
            printf("(%2d) %-18.18s  ", i + 1, mstring[i]);
            if ((i % 3) == 2)
                putchar('¥n');
        }
        printf("( 0) 終了  :");
        scanf("%d", &ch);
    } while (ch < TERMINATE || ch > PRINT_ALL);

    return (Menu)ch;
}

/*--- メイン ---*/
int main(void) {
    Menu menu;
    Member data[] = {
        {1, "takashi"}, {3, "kouji"}, {5, "umeda"}, {7, "satoshi"},
        {6, "noyuri"}, {2, "daisuke"}, {0, "motoko"}, {4, "agemi"},
        {9, "ito"}, {2, "ohta"}
    };
};

int ndata = sizeof(data)/sizeof(data[0]);

do {
    switch (menu = SelectMenu()) {
        case ASCEND_NO : /* 番号で昇順にソート */
            quick(data, 0, ndata-1, AscendingMemberNoCmp);
            break;
        case ASCEND_NAME :/* 名前で昇順にソート */
            quick(data, 0, ndata-1, AscendingMemberNameCmp);
            break;
        case DESCEND_NO : /* 番号で降順にソート */
            quick(data, 0, ndata-1, DescendingMemberNoCmp);
            break;
        case DESCEND_NAME :/* 名前で降順にソート */
            quick(data, 0, ndata-1, DescendingMemberNameCmp);
            break;
    }
} while (menu != PRINT_ALL);
}

```

```
case PRINT_ALL : /* 全データを表示 */
    Print(data, ndata);
    break;
}
} while (menu != TERMINATE);

return 0;
}
```

- 1) このプログラムの動作直後に, 数字の1を入力しました. このとき, 次の間に答えなさい.
 - (ア) main 関数から呼び出された quick 関数の枢軸に選択された要素を示しなさい.
 - (イ) main 関数から呼び出された quick 関数で, swap(Member, a[pl], a[pr])が呼び出される回数は何回ですか.
 - (ウ) main 関数から呼び出された quick 関数の矢印①の場所における, pl, pr, left, right の値はいくらですか.
 - (エ) quick 関数はソートが終わるまで何回呼ばれますか.
- 2) このプログラムの動作直後に, 数字の4を入力しました. このとき, 次の間に答えなさい.
 - (ア) main 関数から呼び出された quick 関数の枢軸に選択された要素を示しなさい.
 - (イ) main 関数から呼び出された quick 関数で, swap(Member, a[pl], a[pr])が呼び出される回数は何回ですか.
 - (ウ) main 関数から呼び出された quick 関数の矢印①の場所における, pl, pr, left, right の値はいくらですか.
 - (エ) quick 関数はソートが終わるまで何回呼ばれますか.

