# プログラミング入門 II　演習報告書

課題番号：6

国田 将人
学籍番号: 201111361
s1111361@coins.tsukuba.ac.jp

2012-02-20

# 課題 1

## リスト

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

struct node {
  int data;
  struct node *left;
  struct node *right;
};

struct node *insert_data(int x, struct node *p);
int search_tree(int x, struct node *p);
void print_tree(struct node *p);

int main(int argc, char *argv[])
{
  FILE *fp;
  int i, x;
  struct node *root;

  if (argc != 2) {
    printf("missing file argument¥n");
    return 1;
  }

  fp = fopen(argv[1], "r");
  if (fp == NULL) {
    printf("can't open %s¥n", argv[1]);
    return 1;
  }

  root = NULL;

  for (i = 0; i < 20; i++) {
    fscanf(fp, "%d", &x);
    root = insert_data(x, root);
  }

  print_tree(root);

  while(1) {
    scanf("%d", &x);
    if (x <= 0)
      break;
    if (search_tree(x, root) == 1)
      printf("%d: Found¥n", x);
    else
      printf("%d: Not found¥n", x);
  }

  fclose(fp);

  return 0;
}
```

（次ページに続く）

```
struct node *insert_data(int x, struct node *p)
{
  if (p == NULL) {
    p = (struct node *)malloc(sizeof(struct node));
    if (p == NULL) {
      printf("Out of memory¥n");
      exit(1);
    }
    p->data = x;
    p->left = NULL;
    p->right = NULL;

    return p;
  }

  if (x == p->data)
    return p;

  if (x < p->data)
    p->left = insert_data(x, p->left);
  else
    p->right = insert_data(x, p->right);

  return p;
}

void print_tree(struct node *p)
{
  if (p == NULL)
    return;

  print_tree(p->left);
  printf("%d¥n", p->data);
  print_tree(p->right);
}

int search_tree(int x, struct node *p)
{
  if (p == NULL)
    return 0;

  if (p->data == x)
    return 1;
  else if (p->data > x)
    return search_tree(x, p->left);
  else
    return search_tree(x, p->right);
}
```

## 実行結果

```
cosmos10:6_120213 s1111361$ ./ex6-1 prog2-ex6-data.txt
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
64
64: Not found
3
3: Found
-1
cosmos10:6_120213 s1111361$
```

# 課題 2

## リスト

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

struct node {
  int data;
  struct node *left;
  struct node *right;
};

struct node *insert_data(int x, struct node *p);
int sum_tree(struct node *p);

int main(int argc, char *argv[])
{
  FILE *fp;
  int i, x;
  struct node *root;

  if (argc != 2) {
    printf("missing file argument¥n");
    return 1;
  }

  fp = fopen(argv[1], "r");
  if (fp == NULL) {
    printf("can't open %s¥n", argv[1]);
    return 1;
  }

  root = NULL;

  for (i = 0; i < 20; i++) {
    fscanf(fp, "%d", &x);
    root = insert_data(x, root);
  }

  fclose(fp);

  printf("sum: %d¥n", sum_tree(root));

  return 0;
}
```

（次ページに続く）

```c
struct node *insert_data(int x, struct node *p)
{
  if (p == NULL) {
    p = (struct node *)malloc(sizeof(struct node));
    if (p == NULL) {
      printf("Out of memory¥n");
      exit(1);
    }
    p->data = x;
    p->left = NULL;
    p->right = NULL;

    return p;
  }

  if (x == p->data)
    return p;

  if (x < p->data)
    p->left = insert_data(x, p->left);
  else
    p->right = insert_data(x, p->right);

  return p;
}

int sum_tree(struct node *p)
{
  if (p == NULL)
    return 0;

  return sum_tree(p->left) + p->data + sum_tree(p->right);
}
```

## 実行結果

```
cosmos10:6_120213 s1111361$ ./ex6-2 prog2-ex6-data.txt
sum: 171
```

prog2-ex6-data.txt にある整数から重複した数を除いて合計すると 171 になった。これにより、この結果が正しいことが確認した。