

第13回 C++プログラミング III

課題出題日: 2023 年 1 月 12 日 (木)

課題提出期限: 2023 年 1 月 18 日 (水) 24:00

第13回講義で扱ったヒープソートに関するプログラムを作成する。CoursePowerで配布されるスケルトンファイル(prac13_skel.cpp)を各自ダウンロードし、ポイントとなる箇所(1)–(7)を埋めることでプログラムを完成させよ。プログラムを実行し、入力データ(data13.csv)に含まれている“数値”を昇順にソートした以下のような実行例が得られることを確かめること。

```
$ cat data13.csv
23,1,12,29,10,16,20,21,8,26,4,11,30,18,24
$ g++ -std=c++17 prac13.cpp
$ ./a.out
array before sorting
23,1,12,29,10,16,20,21,8,26,4,11,30,18,24
array after sorting
1,4,8,10,11,12,16,18,20,21,23,24,26,29,30
```

実行例において、\$はプロンプトを意味する。また与えられた全てのファイルのテキストエンコーディングはUnicode (UTF-8)で統一されている。入力データ(data13.csv)はCoursePowerで配布されているのでダウンロードし使用すること。

以下にprac13_skel.cppを添付する。

```
1 // ヒープソート実装のプログラム
2 #include <iostream>
3 #include <fstream>
4
5 // 関数プロトタイプ宣言
6 void downheap(int a[], int left, int right);
7 void swap(int &a, int &b);
8
9 // ヒープソートで配列 a をソートする関数
10 void heap_sort(int a[], int n){
11     // (i) 配列 a をヒープ化
12     // (1) 処理を完成させる
13
14     // (ii) ヒープ再構築を用いたソート
15     // (2) 処理を完成させる
16 }
17
18 // downheap 処理: a[left] から a[right] をヒープ化させる
19 void downheap(int a[], int left, int right){
20     // a[left] を下流の適切な位置まで下ろし、葉に達するとループを抜ける
21     int parent{left};
22     int child{};
23     while( parent < (right+1)/2 ) {
24         // 左子の添字
25         int cl = // (3) 完成させる
26         // 右子の添字
27         int cr = // (4) 完成させる
28
29         // 右子があるか確認しつつ、大きい方の子の添え字を child へセットする
30         // (5) 処理を完成させる
31
32         // 子が親以下の値であればループを抜ける
33         // (6) 処理を完成させる
```

```

34
35     // 大きい方の子と親の値を交換し、下流に移る
36     // (7) 処理を完成させる
37 }
38 }
39
40 // 配列の内容の表示させる関数
41 void print_array(int a[], int n){
42     for (int i{}; i < n; i++) {
43         if (i != n-1) {
44             std::cout << a[i] << ",";
45         } else {
46             std::cout << a[i];
47         }
48     }
49     std::cout << "\n";
50 }
51
52 // 2つの int 型の変数の入れ替えを行う関数
53 void swap(int& a, int& b){
54     int tmp;
55     tmp = a;
56     a = b;
57     b = tmp;
58 }
59
60 int main(){
61     // ファイル読み込み : data13.csv は CoursePower 上で配布
62     std::ifstream fin("data13.csv");
63     // エラーチェック
64     if (!fin) std::cerr << "cannot open file" << "\n";
65
66     // データファイルを空読みしデータ要素数 size を取得する
67     int size{};
68     std::string st;
69     while (std::getline(fin, st, ',')) size++;
70
71     // ファイルポインタを初期位置に戻す
72     fin.clear();
73     fin.seekg(0, fin.beg);
74
75     // 配列の動的確保
76     int* array = new int[size];
77     int i{};
78     while (std::getline(fin, st, ',')) {
79         array[i] = std::stoi(st);
80         i++;
81     }
82
83     // ソート前の結果を表示
84     std::cout << "array before sorting\n";
85     print_array(array, size);
86
87     // ヒープソートを実行
88     heap_sort(array, size);
89
90     // ソート後の結果を表示
91     std::cout << "array after sorting\n";
92     print_array(array, size);
93
94     return 0;
95 }

```