「データ構造」 第 11 回 資料 & 宿題 (*Jul. 7, 2004,* 奥乃)

1 Heap – siftup and siftdown

ヒープは $x[1]$ **から** $x[n]$ **で構成.**

```
 1 /* root = 1
 2    value(i) = x[i] *
 3    leftchild(i) = 2*i
 4    rightchild(i) = 2*1+1
 5    parent(i) = i/2
 6    null(i) = (i < 1) or (i > n)
 7 */
 8
 9   /* heap 成立条件 heap(1, n):
10      配列 x[1..n] がヒープとして性質を満足する.    */
11
12
13 define siftup(n)
14    /* 実行前     n > 0 && heap(1, n-1) */
15    /* 実行後     heap(1, n) */
16
17 i = n
18 loop
19       /* ループ不変条件: heap(1, n) が成立.
20          ただし, i とその親の間は成立しない可能性あり. */
21
22   if i == 1
23      break
24   p = i/2
25   if x[p] <= x[i]
26      break
27   swap(p, i)
28   i = p
29
30
31 define siftdown(n)
32    /* 実行前    heap(2, n) && n >= 0 */
33    /* 実行後    heap(1, n) */
34
35 i = n
36 loop
37       /* ループ不変条件: heap(1, n) が成立.
38          ただし, i とその子 (個数は 0 〜 2) の間は成立しない可能性あり. */
39
40    c = 2*i
41    if c > n
42       break
43    /* c は i の左側の子 */
44    if c+1 <= n
45       /*c c+1 は i の右側の子 */
46       if x[c+1] < x[c]
47          c = c+1
48    /* c は i の小さい方の子 */
49    if x[i] <= x[c]
50       break
51    swap(c, i)
52    i = c
53
54
55 define insert(t)
56    if n >= maxsize
57       /* report-error */
58    n = n+1
59    x[n] = t
60    /* heap(1, n-1) */
61    siftup(n)
62    /* heap(1, n) */
63
64
```

```
65 define extractmin()
66    /* extractmin finds the minimum element in the set, deletes it
67       and restructures the array to have the heap property. */
68
69    if n < 1
70       /* report-error */
71    t = x[1]
72    x[1] = x[n]
73    n = n-1
74    /* heap(2, n) */
75    siftdown(n)
76    /* heap(1, n) */
77   return t
78
```

2 単純な Heapsort

データを insert し, 次に, extractmin で小さい順に取り出す.

$\Longrightarrow$ **元のデータが格納されている配列がもう一つ必要.**

3 改良版 Hearsort — 配列を 1 つしか使わない

配列 $x[1..n]$ に格納されているデータを大きいもの順に並べる.

```
81 define heapsort(n)
82    /* ループ不変条件: heap(1, i-1) */
83    for (i = 2; i >= n; i++)
84       siftup(i)
85    /* heap(1, n) */
86
86    for (i = n; i >= 2; i--)
87       swap(1, i)
62       siftdown(i-1)
63
```

4 宿題 — 次回の講義前 (10:30 まで) に教壇に 提出すること

1 部分配列 $x[1..12]$ の値が, 24, 40, 30, 59, 47, 33, 44, 70, 80, 52, 99, 38 であるヒープを, 2 分木で図示せよ.

2 問 1 の配列に $x[13]$ に 20 が挿入された時の `siftup` の様子を記述せよ.

3 問 1 の配列に $x[1]$ が 41 で置換された時の `siftdown` の様子を記述せよ.

4 配列 $x[1..n]$ がヒープとして満たすべき制約式を記述せよ.

5 部分配列 $x[l..u]$ がヒープとして満たすべき制約式を記述せよ.

6 n 個のデータを Heapsort で整列するために, `insert` と `extractmin` が呼ばれる回数はそれぞれ何回か.

また, Hearsort の最悪の計算量を求めよ.

7 改良版 Heapsort を小さいもの順に並べるには, どこを変更すればよいか.

8 【随意】 `siftup` の 22 行目の比較を, sentinel (番兵) を使用し, 毎回行なわないようにプログラムを変更せよ.