

データ構造とアルゴリズム 2007 年度期末試験問題

静岡大学工学部システム工学科

安藤 和敏

2008 年 02 月 14 日

注意事項

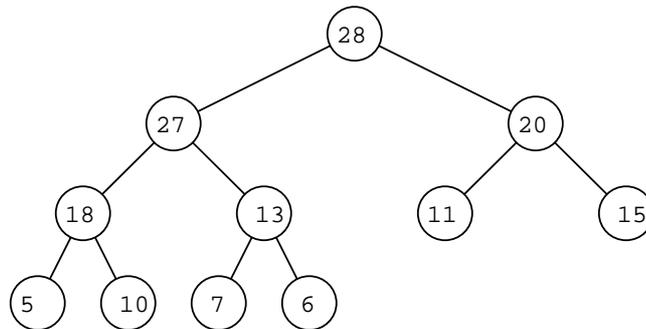
- 学生証を, 写真のある面を上にして, 机の上に置いておくこと.
- 持ち込み一切不可.
- 試験の時間は 10:20-11:40 である.
- 問題用紙は持ち帰ってよい.
- 解答及び採点の結果は, Web ページ (<http://coconut.sys.eng.shizuoka.ac.jp/algo/07/>) で公開する.

問題 1. (配点 30)

図 1 は, ヒープに対する操作 `push_heap` と `delete_maximum` の C 言語による実現である.

設問 (1) 空欄 ~ に最もよくあてはまる式または命令文を記入せよ.

設問 (2) 下図に示したヒープ H に



対して `push_heap(H, 26)` を実行した後のヒープ H を 2 分木で示せ.

設問 (3) さらにその後, `delete_maximum(H)` を実行した後のヒープ H を 2 分木で示せ.

設問 (4) さらにその後, `delete_maximum(H)` を実行した後のヒープ H を配列で示せ. (H[i] に何も入っていないときはそこは空欄にせよ.)

設問 (5) `push_heap(T, x)` の最悪時間計算量と `delete_maximum(T)` の最悪時間計算量を, オーダ記法を用いて表わせ. 配列 T の大きさは, n とせよ.

```

void push_heap(int T[],int x){
    int k;
    size++;
    T[size] = x;
    ア ;
    while(T[k]>T[k/2] && k>1) {
        イ ;
        k=k/2;
    }
}
int delete_maximum(int T[]) {
    int k,big,ret_val;
    ret_val= T[1];
    T[1] = T[size];
    T[size]=-1;
    size--;
    k=1;
    while (2*k <= size) {
        if (2*k==size) {
            if( T[k] < T[2*k]){
                SWAP(T[k],T[2*k]);
                ウ ;
            } else {
                break;
            }
        } else {
            if(T[2*k] > T[2*k+1]) {
                big = 2*k;
            } else {
                big = 2*k+1;
            }
            if(T[k]<T[big]) {
                SWAP(T[k],T[big]);
                エ ;
            } else {
                break;
            }
        }
    }
    return ret_val;
}

```

図 1: 操作 push_heap と 操作 delete_maximum の C 言語による実現

問題 2. (配点 40)

図 2 は, partition と quicksort の C 言語による実現である. ここで, partition における基準値を常に $D[\text{right}]$ としている.

設問 (1) 空欄 ~ に最もよくあてはまる式または命令文を記入せよ.

設問 (2) 以下に示される配列 D

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
D	4	9	7	8	1	3	2	5

に対して, $\text{partition}(D, 0, 7)$ を呼び出した. 関数 partition の外側の while 文の 1 回目から 3 回目の各繰り返しにおいて, $\text{if}(i < j) \text{ SWAP}(D[i], [j]);$ が実行された直後の配列 D を示せ. また, $\text{partition}(D, 0, 7)$ が終了したときの配列 D を示せ.

設問 (3) 大きさ n の配列 D に対する $\text{partition}(D, 0, n-1)$ の最悪時間計算量, 及び, 最良時間計算量をオーダ記法を用いて表わせ.

設問 (4) 設問 (2) で与えた配列 D に対して, $\text{quicksort}(D, 0, 7)$ を呼び出した. 変数 pivot_index に代入される数を答えよ.

設問 (5) 大きさ n の配列 D に対する $\text{quicksort}(D, 0, n-1)$ の最悪時間計算量, 最良時間計算量, 平均計算時間をオーダ記法を用いて表わせ.

```
int partition(int D[], int left, int right) {
    int i, j, k;

    i = left;  ;
    while(i <= j) {
        while(D[i] < D[right])  ;
        while(D[right] <= D[j] && i <= j) j--;
        if(i < j) SWAP(D[i], D[j]);
    }
    SWAP(D[i], D[right]);
    return i;
}

void quicksort(int D[], int left, int right) {
    int pivot_index;
    if(left >= right) return; //要素数が1 になったら終了
    pivot_index = partition(D, left, right);
    quicksort(D, left,  );
    quicksort(D, pivot_index + 1,  );
}
```

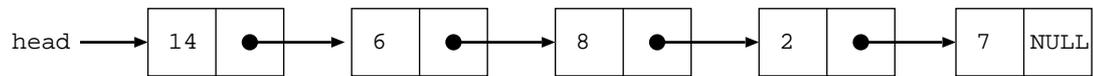
図 2: partition と quicksort の C 言語による実現

問題 3. (配点 30)

図 3 は, 連結リストに対する操作 `insert` と `delete` の C 言語による実現である.

設問 (1) 空欄 ~ に最もよくあてはまる代入文を記入せよ.

設問 (2) 下図のように, ポインタ変数 `head` が連結リストの先頭アドレスを指していると仮定しよう.



`delete(head->next)` を実行した後の連結リストを上図のように図示せよ.

設問 (3) その後, さらに `insert(9,head->next->next)` を実行した. 設問 (2) と同様に, この操作の後の連結リストを図示せよ.

```

typedef struct cell {
    int element;
    struct cell *next;
} cell;

cell *head;

void insert(int x, cell *p) {
    cell *temp;
    temp = (cell *)malloc(sizeof(cell));
    temp->element = x;
    if (p != NULL) {
        

|   |
|---|
| ア |
|---|

 ;
        

|   |
|---|
| イ |
|---|

 ;
    } else {
        temp->next = head;
        head = temp;
    }
}

void delete(cell *p) {
    cell *temp;
    if (head == NULL) {
        printf("Error: List is empty.\n");
        exit(1);
    }
    if (p == NULL) {
        temp = head;
        head = head->next;
        free(temp);
    } else if (p->next != NULL) {
        

|   |
|---|
| ウ |
|---|

 ;
        

|   |
|---|
| エ |
|---|

 ;
        free(temp);
    }
}

```

図 3: insert と delete の C 言語による実現

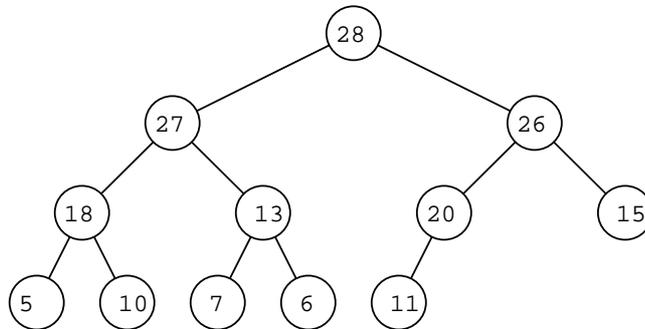
学籍 番号		氏 名	
----------	--	--------	--

問題 1(1) の解答欄 (配点 12)

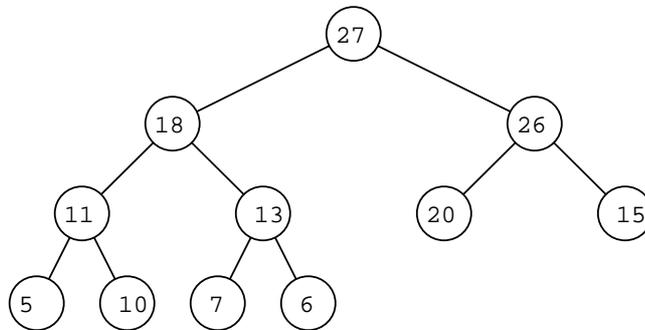
ア. _____ `k= size` _____ イ. _____ `SWAP(T[k],T[k/2])` _____

ウ. _____ `k = 2*k` _____ エ. _____ `k=big` _____

問題 1(2) の解答欄 (配点 4)



問題 1(3) の解答欄 (配点 4)



問題 1(4) の解答欄 (配点 6)

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]
H	26	18	20	11	13	6	15	5	10	7	

問題 1(5) の解答欄 (配点 4)

	最悪時間計算量
<code>push_heap(T, x)</code>	$O(\log n)$
<code>delete_maximum(T)</code>	$O(\log n)$

学籍 番号		氏 名	
----------	--	--------	--

問題 2(1) の解答欄 (配点 12)

ア. _____ `j = right-1` _____ イ. _____ `i++` _____

ウ. _____ `pivot_index-1` _____ エ. _____ `right` _____

問題 2(2) の解答欄 (配点 12)

		[0][1][2][3][4][5][6][7]								
1 回目の while ループが終了するとき	D	<table border="1"><tr><td>4</td><td>2</td><td>7</td><td>8</td><td>1</td><td>3</td><td>9</td><td>5</td></tr></table>	4	2	7	8	1	3	9	5
4	2	7	8	1	3	9	5			
		[0][1][2][3][4][5][6][7]								
2 回目の while ループが終了するとき	D	<table border="1"><tr><td>4</td><td>2</td><td>3</td><td>8</td><td>1</td><td>7</td><td>9</td><td>5</td></tr></table>	4	2	3	8	1	7	9	5
4	2	3	8	1	7	9	5			
		[0][1][2][3][4][5][6][7]								
3 回目の while ループが終了するとき	D	<table border="1"><tr><td>4</td><td>2</td><td>3</td><td>1</td><td>8</td><td>7</td><td>9</td><td>5</td></tr></table>	4	2	3	1	8	7	9	5
4	2	3	1	8	7	9	5			
		[0][1][2][3][4][5][6][7]								
partition が終了したとき	D	<table border="1"><tr><td>4</td><td>2</td><td>3</td><td>1</td><td>5</td><td>7</td><td>9</td><td>8</td></tr></table>	4	2	3	1	5	7	9	8
4	2	3	1	5	7	9	8			

問題 2(3) の解答欄 (配点 6)

最悪時間計算量	最良時間計算量
$O(n)$	$O(n)$

問題 2(4) の解答欄 (配点 4)

4

問題 2(5) の解答欄 (配点 6)

最悪時間計算量	最良時間計算量	平均計算時間
$O(n^2)$	$O(n \log n)$	$O(n \log n)$

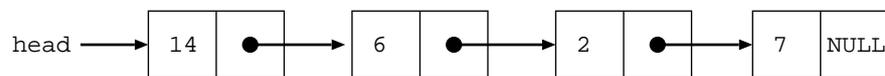
学籍 番号		氏 名	
----------	--	--------	--

問題 3(1) の解答欄 (配点 12)

ア. temp->next = p->next イ. p->next = temp

ウ. temp = p->next エ. p->next = p->next->next

問題 3(2) の解答欄 (配点 9)



問題 3(3) の解答欄 (配点 9)

