

2021 2 학기
자료구조개론
중간고사

Caution

1. Following lines are all included in the program, but omitted due to page limitation

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

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

```
#include <math.h>
```

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

2. All variables and arrays in program are initialized with zeros at the beginning.

ex 1) `int i; double d;` => `i` and `d` is initialized with 0

ex 1) `char s[100];` => all the values are 0 at the start

ex 2) `char s = (char*)malloc(100);` -> all the values in `s` is initialized with 0 when allocated

Q1. Solve the problem related to the following problem

```
#define SWAP(x, y, t) ((t) = (x), (x) = (y), (y) = (t))
void sort(int list[], int n)
{
    int i, j, temp;
    int for_count = 0;
    for(i = 1; i < n; i++) {
        for (j = i - 1; j >= 0; j--) {
            if(list[j + 1] < list[j])
                SWAP(list[j + 1], list[j], temp);
            for_count++;
            (A)
        }
    }

    Printf("%d\n", for_count); (B)
}
```

```
void main(void) {
    int n = 15;
    int list[] = {1, 3, 15, 17, 3, 2, 7, 11, 9, 10, 5, 8, 3, 5, 3};
    sort(list, n);
}
```

- What is time complexity of the sort program? (array len \rightarrow n)
 - $\Theta(1)$
 - $\Theta(n)$
 - $\Theta(\log n)$
 - $\Theta(n \log n)$
 - $\Theta(n^2)$
- What is the name of this sort method
 - Selection sort
 - Insertion sort
 - Bubble sort
 - Quick sort
 - Merge sort
- What is the value of list[12] after sort
 - 1
 - 3
 - 9
 - 11
 - 15
- What is the value of list[3] after sort
 - 3
 - 5
 - 9
 - 15
 - 17
- When i is 10 and j is 6, what is the value of list[4] when program is at (A)
 - 7
 - 20
 - 9
 - 10
 - 1
- When i is 7 and j is 3, what is the value of list[2] when program is at (A)
 - 5
 - 20
 - 3
 - 10
 - 1

g. What is the output of the printf at **(B)**

① 86

② 97

③ 105

④ 113

⑤ 120

h. How many times **SWAP** called in this program

① 48

② 49

③ 50

④ 51

⑤ 52

Q2. Solve the problem related to the following problem

<pre>#define MAX_TERMS 101 #define MAX_COL 10 typedef struct { int col; int row; int value; } term;</pre>	<table><tr><th></th><th>row</th><th>col</th><th>value</th></tr><tr><td>a[0]</td><td>7</td><td>8</td><td>12</td></tr><tr><td>a[1]</td><td>1</td><td>0</td><td>-3</td></tr><tr><td>a[2]</td><td>4</td><td>3</td><td>24</td></tr><tr><td>a[3]</td><td>0</td><td>2</td><td>13</td></tr><tr><td>a[4]</td><td>3</td><td>1</td><td>-2</td></tr><tr><td>a[5]</td><td>0</td><td>3</td><td>4</td></tr><tr><td>a[6]</td><td>0</td><td>6</td><td>-5</td></tr><tr><td>a[7]</td><td>5</td><td>2</td><td>34</td></tr><tr><td>a[8]</td><td>2</td><td>2</td><td>-79</td></tr></table>		row	col	value	a[0]	7	8	12	a[1]	1	0	-3	a[2]	4	3	24	a[3]	0	2	13	a[4]	3	1	-2	a[5]	0	3	4	a[6]	0	6	-5	a[7]	5	2	34	a[8]	2	2	-79
	row	col	value																																						
a[0]	7	8	12																																						
a[1]	1	0	-3																																						
a[2]	4	3	24																																						
a[3]	0	2	13																																						
a[4]	3	1	-2																																						
a[5]	0	3	4																																						
a[6]	0	6	-5																																						
a[7]	5	2	34																																						
a[8]	2	2	-79																																						
<pre>void fasttranspose(term a[], term b[]) { int rowTerms[MAX_COL], startPos[MAX_COL]; int i, j, numCol = a[0].col, numTerms = a[0].value; b[0].row = numCol; b[0].col = a[0].row; b[0].value = numTerms; if (numTerms > 0) { for(i=0; i<numCol; i++) rowTerms[i] = 0; for(i=1; i<=numTerms; i++) rowTerms[a[i].col]++; (A) startPos[0] = 1; for(i=1; i<numCol; i++) startPos[i] = startPos[i-1] + rowTerms[i-1]; (B) for(i=1; i<=numTerms; i++) { j=startPos[a[i].col]++; b[j].row = a[i].col; b[j].col = a[i].row; b[j].value = a[i].value; } (C) } }</pre>	<pre>int main(int argc, char *argv[]) { term a[MAX_TERMS], b[MAX_TERMS]; /* a is initialized with upper value */ fasttranspose(a, b); }</pre>																																								

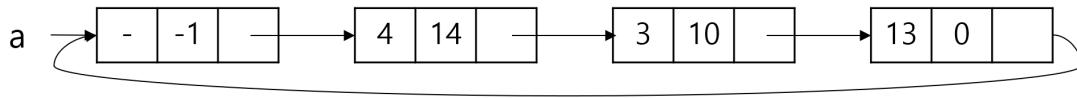
- a. What is the value of `b[2].row` after `fasttranspose`?
① 0 ② 1 ③ 2 ④ 3 ⑤ 8
- b. What is the value of `b[6].value` after `fasttranspose`?
① 12 ② 24 ③ -2 ④ -5 ⑤ -79
- c. What is the value of `b[7].value` after `fasttranspose`?
① -3 ② 13 ③ 4 ④ 34 ⑤ -79
- d. What is the value of the `rowTerms[1]` when program is at (A)
① 0 ② 1 ③ 2 ④ 3 ⑤ 4
- e. What is the value of the `rowTerms[5]` when program is at (A)
① 0 ② 1 ③ 2 ④ 3 ⑤ 4
- f. What is the value of the `startPos[2]` when program is at (B)
① 4 ② 5 ③ 6 ④ 7 ⑤ 8
- g. What is the value of the `startPos[6]` when program is at (B)
① 9 ② 10 ③ 11 ④ 12 ⑤ 13
- h. What is the value of the `startPos[2]` when program is at (C)
① 9 ② 10 ③ 11 ④ 12 ⑤ 13
- i. What is the value of the `startPos[6]` when program is at (C)
① 9 ② 10 ③ 11 ④ 12 ⑤ 13
- j. What is the time complexity of this program (`termNum` -> `n`)?
① $\Theta(1)$ ② $\Theta(n)$ ③ $\Theta(\log n)$ ④ $\Theta(n \log n)$ ⑤ $\Theta(n^2)$

Q3. Solve the problem related to the following problem

```
#define COMPARE(x,y) ((x)<(y)?-1:((x)==(y)?0:1))
typedef struct polyNode *polyPointer;
struct polyNode {
    int coef;
    int expon;
    polyPointer link;
};
polyPointer cpadd (polyPointer a, polyPointer b);
int length (polyPointer last) {
    polyPointer temp;
    int count = 0;
    if (last) {
        temp = last;
        do {
            count++;
            temp = temp->link;
        } while (temp != last);
    }
    return count;
}
void attach(float coefficient,
            int exponent, polyPointer *ptr) {
    polyPointer temp;
    temp =
    (polyPointer)malloc(sizeof(struct polyNode));
    temp->coef = coefficient;
    temp->expon = exponent;
    (*ptr)->link = temp;
    *ptr = temp;
}
void main() {
    polyPointer a,b, c;
    polyPointer init_a,init_b;
    /* a, b is initialized with format below */
    /* check the actual value in problem */
    c= cpadd(a, b);
    printf("%d\n", length(c)); (A)
}
```

```
polyPointer cpadd (polyPointer a, polyPointer b) {
    polyPointer startA, c, lastC;
    int sum, done = 0;
    int else_count = 0;
    startA = a;
    a = a->link;
    b = b->link;
    c = (polyPointer)malloc(sizeof(struct polyNode));
    c->expon = -1;
    lastC = c;
    do {
        switch (COMPARE(a->expon, b->expon)) {
            case -1:
                attach(b->coef,b->expon,&lastC);
                b = b->link;
                break;
            case 0:
                if (startA == a) done = 1;
                else {
                    sum = a->coef + b->coef;
                    if (sum) attach(sum,a->expon,&lastC);
                    else else_count++;
                    a = a->link; b = b->link;
                }
                break;
            case 1:
                attach(a->coef,a->expon,&lastC);
                a = a->link;
        }
    } while (!done);
    lastC->link =c;
    printf("else_count : %d\n", else_count); (B)
    return c;
}
```

$$A(x) = 4x^{14} + 3x^{10} + 13$$



a. If $A(x) = 4x^{14} + 3x^{10} + 13$, $B(x) = -2x^{14} + x^7 + 25x^2$,

what is output of the printf at line **(A)**

- ① 5 ② 6 ③ 7 ④ 8 ⑤ 9

b. If $A(x) = 4x^{14} + 3x^{10} + 13$, $B(x) = -2x^{14} + x^7 + 25x^2$,

what is output of the printf at line **(B)**

- ① 0 ② 1 ③ 2 ④ 3 ⑤ 4

c. If $A(x) = 4x^{14} + 3x^{10} + 13$, $B(x) = -2x^{14} + x^7 + 25x^2$,

what is the value of `c->link->link->link->link->link->expon`

- ① 0 ② 3 ③ 9 ④ 19 ⑤ 20

d. If $A(x) = 2x^{25} + 16x^7 - 3x^2 + x$, $B(x) = 2x^{10} - 33x^3$,

what is output of the printf at line **(A)**

- ① 5 ② 6 ③ 7 ④ 8 ⑤ 9

e. If $A(x) = 2x^{25} + 16x^7 - 3x^2 + x$, $B(x) = 2x^{10} - 33x^3$,

what is output of the printf at line **(B)**

- ① 0 ② 3 ③ 9 ④ 19 ⑤ 20

f. If $A(x) = 2x^{25} + 16x^7 - 3x^2 + x$, $B(x) = 2x^{10} - 33x^3$,

what is the value of `c->link->link->link->link->link->expon`

- ① 0 ② 2 ③ 4 ④ 8 ⑤ 14

g. If $A(x) = x^3 + 3x^2 + 3x^1 + 1$, $B(x) = -x^3 - 2x^2 - 1$,

what is output of the printf at line **(A)**

- ① 0 ② 1 ③ 2 ④ 3 ⑤ 4

h. If $A(x) = x^3 + 3x^2 + 3x^1 + 1$, $B(x) = -x^3 - 2x^2 - 1$,

what is output of the printf at line **(B)**

- ① 0 ② 1 ③ 2 ④ 3 ⑤ 4

i. If $A(x) = x^3 + 3x^2 + 3x^1 + 1$, $B(x) = -x^3 - 2x^2 - 1$,

what is the value of c->link->link->link->link->link->expon

- ① -1 ② 0 ③ 1 ④ 2 ⑤ 3

j. What is time complexity of the pmatch program? (A(x) term -> n, B(x) term -> m)

- ① $\Theta(1)$ ② $\Theta(n)$ ③ $\Theta(n+m)$ ④ $\Theta(n*m)$ ⑤ $\Theta(n^2)$

Q4. Solve the problem related to the following problem

<pre> #define MAX_STACK_SIZE 1000 #define MAX_ROWS 8 #define MAX_COLS 8 #define EXIT_ROW 6 #define EXIT_COL 6 typedef struct __element{ int row; int col; int dir; }element; typedef element elements; typedef struct __offset{ int vert; int horiz; }offset; elements stack[MAX_STACK_SIZE]; offset move[8]; int maze[MAX_ROWS][MAX_COLS]; int mark[MAX_ROWS][MAX_COLS]; int top; void path(void); element pop(void){ return stack[top--]; } void push(element e){ stack[top++] = e; } int main(void) { path(); } </pre>	<pre> void path(void){ /* output a path through the maze if such a path exists*/ int i, row, col, nextRow, nextCol, dir = 0; int found = false; int total_tries = 0; int right_path_tries = 0; element position; mark[1][1]=1; top=0; stack[0].row=1; stack[0].col=1; stack[0].dir=0; while (top>-1 && !found) { position = pop(); row = position.row; col = position.col; dir= position.dir; while (dir< 8 && !found) { /* move in direction dir*/ nextRow= row + move[dir].vert; nextCol= col + move[dir].horiz; Ⓐ if (nextRow==EXIT_ROW && nextCol==EXIT_COL) found = true; else if (!maze[nextRow][nextCol] && !mark[nextRow][nextCol]){ mark[nextRow][nextCol] = 1; position.row= row; position.col= col; position.dir= ++dir; push(position); row = nextRow; col = nextCol; dir= 0; right_path_tries++; } </pre>
---	---

<pre>/* Continued from path function */ else{ ++dir; } total_tries++; } /* while (dir< 8 & !found) */ } /* while (top>-1 && !found) */ if (found) { printf("The path is:\n"); printf("row col\n"); for (i=0; i<top; i++) printf("%2d%5d\n", stack[i].row, stack[i].col); printf("%2d%5d\n", row, col); printf("%2d%5d\n", EXIT_ROW, EXIT_COL); } else{ printf("The maze does not have a path\n"); } printf("total_tries : %d\n", total_tries); ❷ printf("right_path_tries : %d\n", right_path_tries); ❸ }</pre>	<p>Maze</p> <table><tr><th>Indices</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>2</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>3</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr><tr><td>4</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>5</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>6</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>7</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table> <p>Move</p> <table><tr><th></th><th>vert</th><th>horiz</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>2</td><td>1</td><td>0</td></tr><tr><td>3</td><td>1</td><td>-1</td></tr><tr><td>4</td><td>0</td><td>-1</td></tr><tr><td>5</td><td>-1</td><td>-1</td></tr><tr><td>6</td><td>-1</td><td>0</td></tr><tr><td>7</td><td>-1</td><td>1</td></tr></table>	Indices	0	1	2	3	4	5	6	7	0	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1	1	2	1	0	1	0	0	1	1	1	3	1	0	1	0	1	1	0	1	4	1	0	1	1	0	1	0	1	5	1	0	0	1	0	0	1	1	6	1	1	0	1	0	0	0	1	7	1	1	1	1	1	1	1	1		vert	horiz	0	0	1	1	1	1	2	1	0	3	1	-1	4	0	-1	5	-1	-1	6	-1	0	7	-1	1
Indices	0	1	2	3	4	5	6	7																																																																																																					
0	1	1	1	1	1	1	1	1																																																																																																					
1	1	0	0	0	1	0	1	1																																																																																																					
2	1	0	1	0	0	1	1	1																																																																																																					
3	1	0	1	0	1	1	0	1																																																																																																					
4	1	0	1	1	0	1	0	1																																																																																																					
5	1	0	0	1	0	0	1	1																																																																																																					
6	1	1	0	1	0	0	0	1																																																																																																					
7	1	1	1	1	1	1	1	1																																																																																																					
	vert	horiz																																																																																																											
0	0	1																																																																																																											
1	1	1																																																																																																											
2	1	0																																																																																																											
3	1	-1																																																																																																											
4	0	-1																																																																																																											
5	-1	-1																																																																																																											
6	-1	0																																																																																																											
7	-1	1																																																																																																											

What is result of the printf at line ②

a. What point is not in the path

- ① (1, 2) ② (1, 3) ③ (3, 3) ④ (5, 5) ⑤ (6, 5)

b. What is the value of position.dir when nextRow is 1 and nextCol is 3 at ②

- ① 0 ② 1 ③ 2 ④ 3 ⑤ 4

c. What is the value of position.dir when nextRow is 4 and nextCol is 5 at ②

- ① 0 ② 1 ③ 2 ④ 3 ⑤ 4

d. What is the result of the printf at line **(B)**

- ① 12 ② 13 ③ 14 ④ 15 ⑤ 16

e. What is the result of the printf at line **(C)**

- ① 6 ② 7 ③ 8 ④ 9 ⑤ 10

f. Suppose `maze[5][5] = 1`, then

What point is not in the path

- ① (1, 2) ② (1, 3) ③ (2, 3) ④ (5, 4) ⑤ (6, 5)

g. Suppose `maze[5][5] = 1`, then

What is the value of `position.dir` when `nextRow` is 1 and `nextCol` is 2 at **(A)**

- ① 0 ② 1 ③ 2 ④ 3 ⑤ 4

h. Suppose `maze[5][5] = 1`, then

What is the value of `position.dir` when `nextRow` is 3 and `nextCol` is 5 at **(A)**

- ① 0 ② 1 ③ 2 ④ 3 ⑤ 4

i. Suppose `maze[5][5] = 1`, then

What is the result of the printf at line **(B)**

- ① 12 ② 13 ③ 14 ④ 15 ⑤ 16

j. Suppose `maze[5][5] = 1`, then

What is the result of the printf at line **(C)**

- ① 6 ② 7 ③ 8 ④ 9 ⑤ 10