ESC 101N: Fundmentals of Computing

(2010-11-IInd Semester)

End Sem Examination (Paper A)
12.15PM-3.15PM, 27th April, 2011

Combined Solutions for Paper A / B

(Including distribution of marks)

Question Mapping Between A and B									
Part A	1	2	3	4	5	6	7	8	9
Part B	4	3	2	5	6	7	8	9	1
Marks	13	10	10	10	7	6	10	10	10

Note: It has come to our notice that the corrections related to questions Q6 in Paper A / Q7 in Paper B were not announced in L7 due to some confusion in announcements. To handle this situation, the paper will now be graded from full marks 86 (not 90). Other corrections were incorporated in the solution.

1. Solution to question 1 (paper A) / question 4 (paper B)

Same for both:

(i) Type mismatch: p is pointer to int but a[0] int [1 mark]

(ii) Legal and evaluate to 1. [1 mark]

(iii) Legal evaluates to 1. [1 mark]

(iv) Legal evaluates to 1. [1 mark]

```
void makeZero(int a[], int n) {
    int *p;                                          [1 mark]
    for (p = a; p < a + n; p++)                      [2 marks]
        *p = 0;                                      [1 mark]
}
```

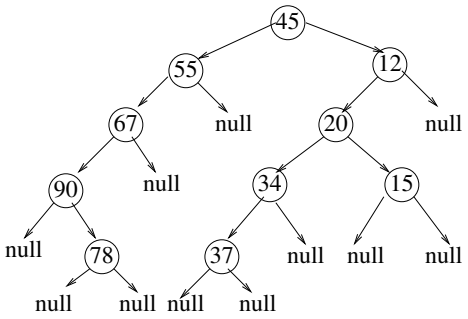
```
int searchKey(int a[], int n, int key) {
   int *p;                                           [1 mark]
   for (p = a; p < a + n; p++)                       [2 marks]
      if(*p == key)                                  [1 mark]
         return 1;                                   [1 mark]
   return 0;
}
```

2. Solution to question No. 2 (paper A) and question No. 3 (paper B).

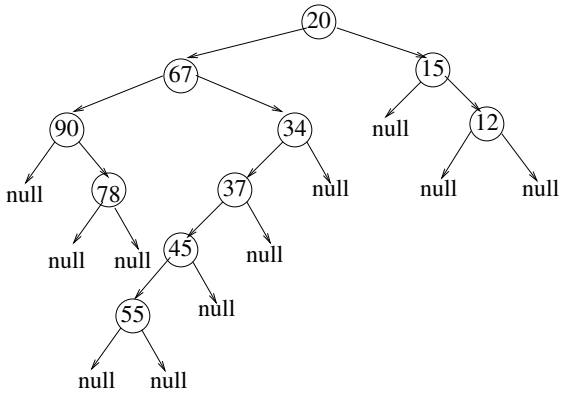
   Data for paper A (Q2): 45, 55, 12, 20, 15, 67, 90, 34, 37, 78, -1

   Data for paper B (Q3): 20, 67, 15, 90, 34, 78, 37, 45, 55, -1

```
5 marks for correct tree upto height 2.
Rest 5 marks for remaining part
If all nulls pointers are correst then 2 marks.
For part A
==========
```




```
For part B
==========
```



3. Solution to question No. 3 (paper A) and question No. 2 (paper B).

   Data for paper A : Array = {4, 5, 1, 2, 3, 6}, k = 3.

   Data for paper B: Array = {3, 5, 1, 2, 4, 6}, k = 2.

Solution to part (a):

```
?1?:  x <= a[right] / a[left]                              [1 marks]
?2?:  x >= a[left] / x <= a[right]                         [1 marks]
?3?:  a[right--] = a[left] / a[left++] = a[right]          [1 marks]
?4?:  a[right] = x / x >= a[left]                           [1 mark]
```

Solution to part (b)

```
The call sequence will be
select(a, 0, 5, 3) / select(a, 0, 5, 2)                          [2]
select(a, 0, 3, 3) / select(a, 0, 2, 2)                          [2]
select(a, 3, 3, 3) / select(a, 2, 2, 2)                           [2]
```

4. Solution to question 4 (paper A) and question 5 (paper B).

   Data for paper A: {2,5,3,1,4}.

   Data for paper B: {3.2.5.1.4}

```
2,5,1,3,4 / 3,2,1,5,4                                      [2 marks]
2,1,5,3,4 / 3,1,2,5,4                                      [2 marks]
1,2,5,3,4 / 1,3,2,5,4                                      [2 marks]
1,2,3,5,4 / 1,3,2,4,5                                      [2 marks]
1,2,3,4,5 / 1,2,3,4,5                                      [2 marks]
```

5. Soution to question 5 (paper A) and question 6 (paper B)

Same for both:

```
(a) ?1?:  1                                                [1 marks]
    ?2?:  1 + syrlength(syr(n))                            [2 marks]
(b)
     int syrLength(int n) {
        if (syrn(n) == 1)                                   [1 mark]
            return 1;                                       [1 mark]
        return 1 + syrLength(syrn(n));                     [2 marks]
    }
```

6. Solution to question 6 (paper A) and question 7 (paper B)

   Note: The correction "Assume n to be odd" (not announced in L7). So, only three parts are graded.

```
L4:  c - 'a'; / L6:  i + 'a';                                  [2 mark]
L8:  else if (c >= 'A' && c <= 'Z') (same for both)            [2 marks]
L11:  i + 'a'; / L10:  (i + k) % 26;                           [2 marks]
Solution for last two errors (scrapped from question paper, and
not graded) will be as follow:
L24:  *(s+n-1) = rotationk(*(s+n-1), k); (same for both)       [3
marks]
L25:  stringRotation(s+1, n-1, k); (same for both)             [3 marks]
```

7. Solution to question 7 (paper A) and question 8 (paper B).

   Correction announced: delete the first of the two blanks marked ?1? in the question.

Same for both:

```
?1?:  struct Rectangle *rotated = malloc(sizeof(struct
rectangle *));                                                 [2 marks]
?2?:  rotated->x1 = r->x2;                                     [2 marks]
?3?:  rotated->y1 = r->y1;                                     [2 marks]
?4?:  rotated->x2 = r->x2 + (r->y2 - r->y1);                   [2 marks]
?5?:  rotated->y2 = r->y1 + (r->x2 - r->x1);                   [2 marks]
```

8. Solution to question 8 (paper A) and question 9 (paper B).

Same for both:

```
?1?:  **a                                                      [2 marks]
?2?:  SIZE * sizeof(int *)                                     [2 marks]
?3?:  i * sizeof(int)                                          [2 marks]
?4?:  i < SIZE                                                 [2 marks]
?5?:  j <= i                                                   [2 marks]
```

9. Solution to question 9 (paper A) and question 1 (paper B)

Same for both:

Left column	Right column (Q1 in part B)
1	i / n [1 mark]
2	c / i [1 mark]
3	n / k [1 mark]
4	k / j [1 mark]
5	e [1 mark] / f [2 marks]
6	j / h [1 mark]
7	h / e [1 mark]
8	l / c [1 mark]
9	f [1 mark] / l [2 marks]