

1	2	3	4	5	
/8	/10	/10	/10	/7	/45

Name _____

Student Number _____

Instructor: S. Qiao

SFWR ENG 2S03/COMP SCI 2S03

Day Class

Duration of examination: 2 hours

McMaster University Final Examination

December 2011

This examination paper includes 6 pages and 5 questions. You are responsible for ensuring that your copy of the paper is complete. Bring any discrepancy to the attention of your invigilator.

SPECIAL INSTRUCTIONS: This paper must be returned with your answers. Use of McMaster standard (Casio-FX991) calculator only is allowed.

I. Assume that the function `Mystery` has been defined as follows:

```
int Mystery(int m, int n) {
    if (n == 0) {
        return 1;
    } else {
        return m * Mystery(m, n - 1)
    }
}
```

(a) (4 marks) What is the value of `Mystery(2, 3)`?

This function computes m^n .

The value of `Mystery(2, 3)` is 8.

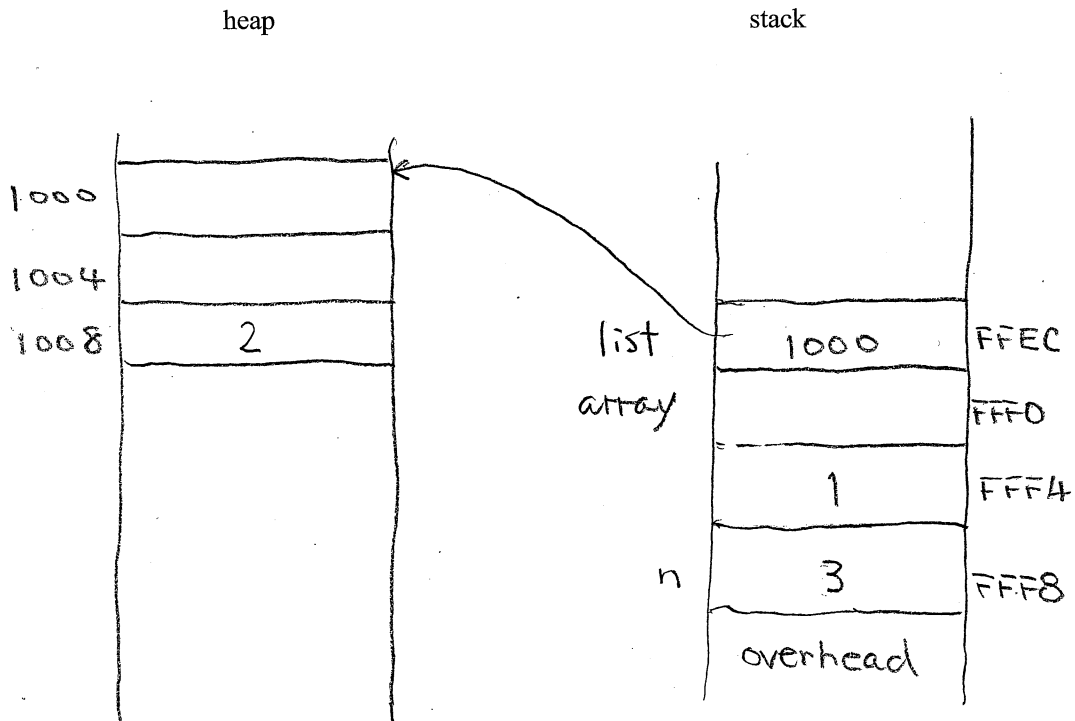
(b) (4 marks) What is the computational complexity of `Mystery(m, n)` in term of the big- O notation, where N is the value of the argument n , assuming $n \geq 0$?

$O(N)$

2. Draw a heap-stack diagram showing how memory is allocated for each of the following problems. Use both explicit addresses and arrows.

(a) (5 marks)

```
void main() {
    int n = 3;
    int array[2];
    array[1] = 1
    int *list = new int[n];
    list[2] = 2;
    <----- diagram at this point
}
```



(b) (5 marks)

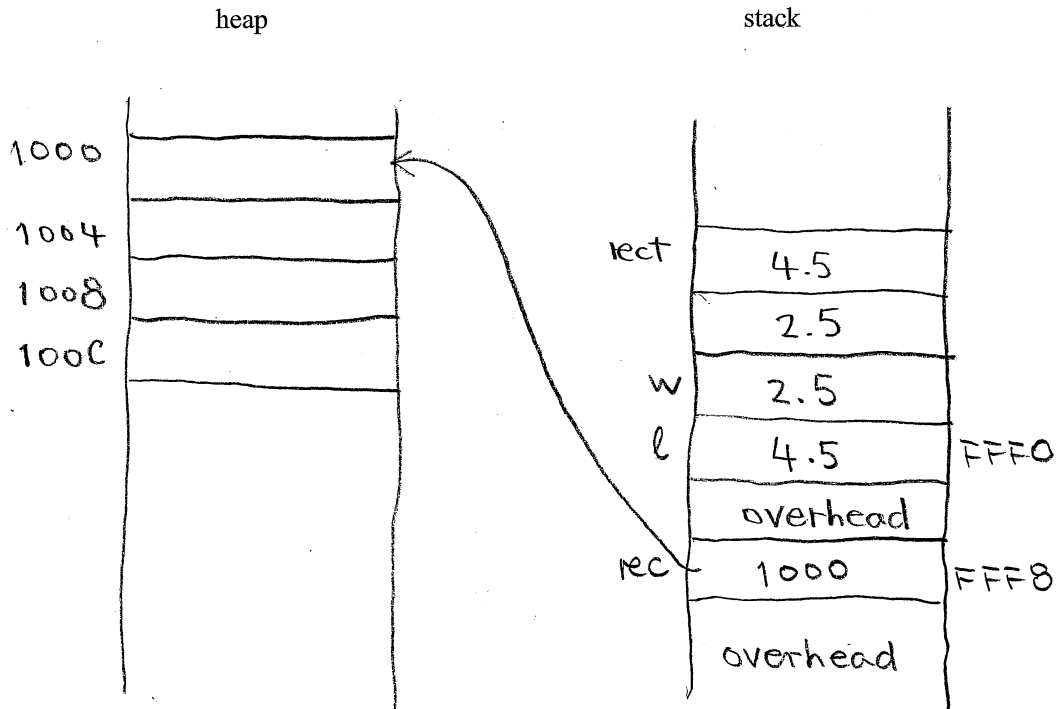
```

struct rectangleT {
    float length;
    float width;
}

rectangleT MakeRectangle(float l, float w);

int main() {
    rectangleT *rec = new rectangleT[2];
    rec[1] = MakeRectangle(4.5, 2.5);
    return 0;
}

rectangleT MakeRectangle(float l, float w) {
    rectangleT rect;
    rect.length = l;
    rect.width = w;
    <---- diagram at this point
    return rect;
}
    
```



3. (10 marks) Complete the following implementation of the selection sort algorithm.

```
/*
 * Function Sort
 * -----
 * Sorts a array of effective size n using
 * the selection sort algorithm.
 */

void Sort(int array[], int n) {
    int rh;
    int lh;

    for (lh = 0; lh < n-1; lh++) {
        rh = lh;
        for (int i = lh+1; i < n; i++) {
            if (array[i] < array[rh]) {
                rh = i;
            }
        }
        int tmp = array[lh];
        array[lh] = array[rh];
        array[rh] = tmp;
    }
}
```

4. (10 marks) Assuming the type definition:

```
struct cellT {  
    int value;  
    cellT *next;  
};
```

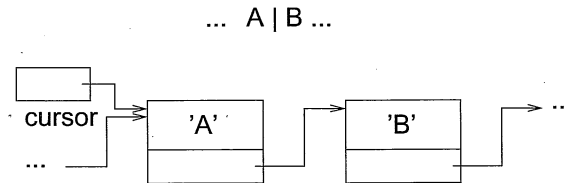
Write a function `PrintLinkedList (cellT *head)` which takes a pointer to a linked list (without dummy cell) as input and prints the value of each cell of the list in order.

```
void PrintLinkedList (cellT *head) {  
    cellT *ptr;  
    for (ptr = head; ptr != NULL; ptr = ptr->next) {  
        cout << ptr->value << endl;  
    }  
}
```

5. (7 marks) Assuming the type definition:

```
struct cellT {
    char ch;
    cellT *next;
};
```

The following figure represents an editor buffer implemented by linked list.



Complete the function

```
void EditorBuffer::insertCharacter(char ch) {
```

```
    cellT *cp = new cellT;
```

```
    cp->ch = ch;
```

```
    cp->next = cursor->next;
```

```
    cursor->next = cp;
```

```
    cursor = cp;
```

```
}
```

END