Question (1) [20 marks]

Mr Readitover, the chief editor of the Important Gazette Newspaper, has hired you to develop a text document summariser for the paper. The primary objective of this piece of software will be to automate the assessment of the many text documents that are submitted for publication in the “Dear Miss Ally” section of the newspaper. Design an algorithm solution using pseudocode that reads any given text-file and performs the following operations and output their results:

a. Determines the number of numbers 0 through 9 on each line
b. Determines the number of spaces per line.
c. Determines the number of question sentences within the file. For example, sentences similar to the following would be counted as a single question sentence:

   What do you think I should do Miss Ally?

d. Determines the number of sentences with exclamations made in the document.
e. Determines whether the file is acceptable for publication based on whether the number of lines is less than 30.

You also need to conduct error checking for file errors.

NOTE: You are not required to implement the algorithm in C! Only pseudocode is required.
Question (2) [15 marks]

(a) Consider the following C program:

```c
//Mystery Program
1. #include <stdio.h>
2. #define SIZE 5
3. int main()
4. {
5.     int array[100], minimum, size, location = 1;
6.     printf("Enter %d integers\n", SIZE);
7.     for ( int c = 0 ; c <= SIZE - 1 ; c++ )
8.         scanf("%d", &array[c]);
9.     mystery = array[0];
10.    for ( int c = 1 ; c <= SIZE - 1 ; c++ )
11.        if ( array[c] < mystery )
12.            mystery = array[c];
13.            location = c+1;
14.    }
15.    printf("Mystery element is present at location %d and its value is \n", location, mystery);
16.    return 0;
17. }
```

(i) Give the output generated by the C program if the numbers entered are 26, 67, 990, 8, 1005.

(ii) Identify and explain how SIZE in the pre-processor directives is used and what its scope
     is. Also, identify and explain the scope of the variable c at lines 8 and c at line 11.

(iii) What is the variable mystery used to calculate?

(b) Perform and show a desk trace that includes the output of the following C-code and briefly state
    what the code does.

```c
#include <stdio.h>

main()
{
    float k=1.0;
    int i = 3;

    while (k <= 3)
    {
        printf("Next number is %5.1f \n", j, k);
        k++;
    }
```

[3 marks]

PLEASE TURN OVER
Question (3) [15 marks]

a. Write a small program to output the following. Use nested for loops.

```
 * 
 *** 
 ***** 
 ******** 
 ********** 
***********
```

[6]

b. A certain computer uses 16 bits to store floating point numbers. This system uses the first bit for the sign, the next 5 bits for the exponent (in excess 8) and the remaining bits for the mantissa.

What number is represented by 0100001101010101.

[4]

(b) Review the sample code below and indicate (using the line number) those lines which have errors (logic and/or syntax) and what the error is. For each error, give the correct version of the incorrect line of code. The code is intended to take two distances in inches/feet and store in them data members of two structure variables. Then, this program should calculate the sum of two distances and display it.

```
1. include <stdio.h>
2. struct Distance{
3.   int feet;
4.   float inch;
5.   |
6.   d1,d2;
7. }
8. int main(){
9.   printf("Enter information for 1st distance\n");
10.  printf("Enter feet: ");
11.  scanf("%d", &feet_d1);
12.  printf("Enter inch: ");
13.  scanf("%f", &inch_d1);
14.  printf("\nEnter information for 2nd distance\n");
15.  printf("Enter feet: ");
16.  scanf("%d", &feet_d2);
17.  printf("Enter inch: ");
18.  scanf("%f", &inch_d2);
19.  sum.feet= feet_d1 + feet_d2;
20.  sum.inch= inch_d1 + inch_d2;
21.  /* If inch is greater than 12, changing it to feet. */
22.  if (sum.inch>12.0)
23.   {
24.     sum.inch=sum.inch-12.0;
25.     ++sum.feet;
26.   }
27.  printf("\nSum of distances=%d\"%.1f\"",sum.feet,sum.inch);
28.  return 0;
29. }
```
Question (A) [10 marks]

a) Use the declarations given below in a complete C program to achieve the following tasks:

```c
#define FILENAME "ClassData.dat"
FILE *myPt;
char fName[15];
char lName[20];
int classsize;
float examScores[5];
float sum = 0.0;
float average;
```

i. Open the file for appending. Conduct necessary checking for errors and request the number of students in the class.

ii. Print the initials of each student, ask for each of the exam scores, store in the array, provide their average score; and print the lowest and highest average in the class.

iii. Close the file and end the program.

[3+6+1]

End of Exam