This assignment is to be done individually.

Important Note: The university policy on academic dishonesty (cheating) will be taken very seriously in this course. You may not provide or use any solution, in whole or in part, to or by another student.

You are encouraged to discuss the concepts involved in the questions with other students. If you are in doubt as to what constitutes acceptable discussion, please ask! Further, please take advantage of office hours offered by the instructor and the TA if you are having difficulties with this assignment.

DO NOT:

- Give/receive code or proofs to/from other students

- Use search engines to find solutions for the assignment

DO:

- Meet with other students to discuss assignment (it is best not to take any notes during such meetings, and to re-work assignment on your own)

- Use online resources (e.g. Wikipedia) to understand the concepts needed to solve the assignment

Submission Instructions:

- You may type or write your answer as long as it is readable.

- Submit two files on CourSys

  1. `report.pdf`, which contains a write-up of your solutions to the assignment
  2. `lozenge.c`, which contains the code you wrote in Problem 2.

Problem 1

a) Consider the code snippet given below. The variable `sum` is of type `int`. Will this compile? Explain why or why not (1 sentence). (3 marks)

```
1  int sum = 2 + '2';
```

b) What will be the value of the result variable after calling `hcf` function? (5 marks)

```
1  #include <stdio.h>
2
3  int hcf(int *x, int *y) {
4      *x = *y;
5
6      int a = *x;
7      int b = *y;
8
9    while (b != 0) {
10      int tmp = b;
11        b = a % b;
12        a = tmp;
13    }
14    return a;
15 }
16
17 int main() {
18     int a = 54;
19     int b = 24;
20     int result = hcf(&a, &b);
21      return 0;
22 }
```

c) Get the best Big-O estimate that you can and briefly justify your answers (1-2 sentences). Consider running times for all the operations contained within the loop body and ignore the running times for initializer, entry condition and increment. (5 marks)

```
1  #include <stdio.h>
2
3  int main() {
4      int i = 0;
```

```
 5        int n = 10;
 6        int j;
 7        while (i < n) {
 8            i++;
 9            j = i;
10            while (i < n) {
11                printf("hello %d\n", i);
12                i++;
13            }
14            i = j;
15        }
16        return 0;
17 }
```

Problem 2

Create a lozenge using programming. (12 marks)

input: n (where $n > 0$)

output: for valid n lozenge with n star on each side, otherwise print "Give me a valid input".

sample 1:

input: 3

output:

```
1     *
2    * *
3   * * *
4    * *
5     *
```

sample 2:

input: 6

output:

```
1        *
2       * *
```

```
 3 |     *  *  *
 4 |   *  *  *  *
 5 | *  *  *  *  *
 6 |*  *  *  *  *  *
 7 | *  *  *  *  *
 8 |  *  *  *  *
 9 |   *  *  *
10 |    *  *
11 |     *
```