

Indian Institute of Technology Kharagpur

AUTUMN Semester, 2015

COMPUTER SCIENCE AND ENGINEERING

CS19001: Programming and Data Structure Laboratory

Assignment – 8

Full Marks: 12

Time allowed: 3 hours

INSTRUCTIONS: Please see the questions and write C programs step by step. Ensure proper indentations to improve the readability of your code. All these features are necessary and absence will lead to deduction of marks.

Please do not forget to upload files to *Moodle* before you leave.

Structures in C

1. A polynomial can be represented as an array of structure in the following manner: each element of the array represents one term of the polynomial, and is a structure that contains the coefficient (a float) and the exponent(an integer). Let such a structure be named `PolyTerm`. For convenience, you can store the terms of the polynomial in decreasing order of exponent in the array. Given two such polynomials (entered by the user), write a complete C program with a function `void mult_poly (PolyTerm poly_1[], int max_exp_poly_1, PolyTerm poly_2[], int max_exp_poly_2, PolyTerm poly_result[])` to calculate and store the product polynomial (for the polynomials `poly_1` and `poly_2`) in the array `poly_result`. Your program should print the coefficients of the terms of the product polynomial, in decreasing order of exponent, e.g. if the product polynomial is $x^3 + 3x^2 + 3$, the program should print: 1 3 0 3. If you use dynamic memory allocation in your program, make sure to free all dynamically allocated memory before your program exits. (4 marks)
2. Write a C program that takes two distances in the inch-feet system, and stores in data members of two structure variables. Then, the program calculates and displays the sum of the two distances again in the inch-feet system. If the user enters an inch value greater than 12, proper corrections are made. (Hint: 1 foot = 12 inches). (3 marks)

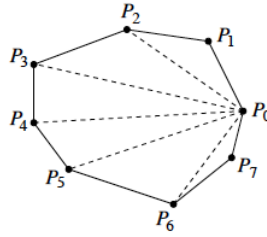


Figure 1: Triangulation of a convex polygon.

3. A *convex polygon* is a simple polygon in which each interior angle is less than 180° . In this exercise, you are required to compute the area of a convex polygon. Perform the following steps:

- (a) Define a data type `point` consisting of double-precision floating-point fields x and y .
- (b) Next, define a static array of `MAX` structures of type `point` as the data type `cpolygon`, using the following statement:

```
typedef point cpolygon [MAX] ;
```

where you are assuming that a convex polygon can have maximum `MAX` sides. You can define `MAX` to be a macro of reasonable value (e.g. 15). Note that a convex polygon is to be stored in an array of points as a counterclockwise sequence of the vertices of the polygon.

- (c) Complete the function `double triangleArea(cpolygon P, int i, int j, int k)` that accepts a variable of data type `cpolygon`, and three indices i, j, k representing vertices P_i, P_j, P_k of the polygon stored in the input array. Note that the `cpolygon` datatype directly represents an array. The function computes and returns the area of the triangle $P_iP_jP_k$ using the formula:

$$\Delta = \sqrt{s(s-a)(s-b)(s-c)}$$

where a, b, c are the lengths of the three sides of the triangle and $s = (a+b+c)/2$ is the *semi-perimeter* of the triangle. You may use math library functions.

- (d) In order to compute the area of the convex polygon, the polygon is first triangulated as shown in Fig. 1. The area of each triangle is calculated using the function `triangleArea()`. The sum of these areas is returned as the area of the input polygon. Write a function `double polygonArea(cpolygon P, int num_sides)` that performs this task. (5 marks)
