PSCP Assignment-6
Due Date: March 5th, 2021 Section-C

1)Use a double-subscripted array to solve the following problem. A company has four salespeople (1 to 4) who sell five different products (1 to 5). Once a day, each salesperson passes in a slip for each different type of product sold. Each slip contains the following:

a) The salesperson number
b) The product number
c) The total dollar value of that product sold that day

Thus, each salesperson passes in between 0 and 5 sales slips per day. Assume that the information from all of the slips for last month is available. Write a program that will read all this information for last month's sales and summarize the total sales by salesperson by product. All totals should be stored in the double-subscripted array sales. After processing all the information for last month, print the results in tabular format with each of the columns representing a particular salesperson and each of the rows representing a particular product. Cross total each row to get the total sales of each product for last month; cross total each column to get the total sales by salesperson for last month. Your tabular printout should include these cross totals to the right of the totaled rows and to the bottom of the totaled columns.

2) Two frogs are sitting at the bottom of a flight of 10 steps and debating in how many ways then can jump up the stairs. They can jump one, two or three steps at once. For example, they can cover the 10 steps by jumping (3,3,3,1) or (2,3,2,1,2) or other suitable combinations of steps. Their mathematics is not very strong and they approach you for help in order to find out the total number of possibilities they have to reach the top. Please provide them with a general solution (not only for 10 but for general n steps) in the form of a program Note that the order of the steps is important here, i.e., (3,3,3,1) is treated distinct from (1,3,3,3) for example.

3) An integer matrix of *m* rows and *n* columns is said to be sparse if more than *(m×n)/2* elements are zero. Write a program in C++ to display the non-zero elements of a sparse matrix in the triplet format: <*row_index,col_index, value*>.

Example: Input: Output:

$$\begin{bmatrix} 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$< 0,0,8 >$
$< 1,1,2 >$
$< 1,4,3 >$
$< 3,0,5 >$

4) Given an array of integers. Find a peak element in it. An array element is peak if it is NOT smaller than its neighbors. For corner elements, we need to consider only one neighbor. For example, for input array {5, 10, 20, 15}, 20 is the only peak element. For input array {10, 20, 15,

2, 23, 90, 67}, there are two peak elements: 20 and 90. Write a program to print all the peak elements in a given array.

5) A rich man died. In his will, he has divided his gold coins among his 5 sons, 5 daughters and a manager. According to his will: First give one coin to manager. 1/5th of the remaining to the elder son. Now give one coin to the manager and 1/5th of the remaining to second son and so on. After giving coins to 5th son, divide the remaining coins among five daughters equally. All should get full coins. Find the minimum number of coins he has? Write a program to display the number of coins that each of his sons, daughters and the manager got?

6) The Legendre polynomial can be calculated by means of the formulae:

$P_0 = 1,$

$P_1 = x,$

$P_n = [(2n-1)/n] \times P_{n-1} - [(n-1)/n] \times P_{n-2}$

Where n=2,3,4... and x is any floating point number between -1 and 1. Write a program using ***recursive function*** to generate the first n Legendre polynomials Let the values of n and x be the input parameters.

7) Write and test a function that creates Pascal's Triangle in the square matrix that is passed to it. For example, if the 2-D array ***x*** and the integer ***4*** are passed to the function, then it would load the following into ***x***.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 1 & 3 & 3 & 1 & 0 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

8) In the theory of games and economic behaviour, founded by John von Neumann, certain games can be represented by a single two-dimensional array, called the *payoff matrix*. Players can obtain optimal strategies when the payoff matrix has a saddle point. A saddle point is an entry in the matrix that is both the minimax and maximin. The minimax of a matrix is the minimum of the column maxima, and the maximin is the maximum of the row minima. The optimal strategies are possible when these two values are equal. Write a program that prints the minimax and the maximin of a given matrix.