**DESIGN AND ANALYSIS OF ALGORITHMS**

1. **Ex1**

Algorithm ***F(n)***

**//The algorithm caculate the sum of formula**

**//Input: n**

**//Output: Sum of formula**

**sum 0**

**for i 2 to n do**

**sum = sum + 1/(1-i\*\*i)**

**return sum**

1. **Ex2**

Algorithm ***F(n)***

**//The algorithm caculate the sum of formula**

**//Input: n**

**//Output: sum of formula**

**sum 0**

**for i 1 to n do**

**//m is an arbitrary parameter**

**sum = sum +**

**return sum**

1. **Ex3**

Algorithm ***F(n)***

**//The algorithm caculate the sum of formula**

**//Input: n**

**//Output: sum of formula**

**sum 0**

**for i 1 to n do**

**sum = sum + i\*\*3**

**return sum**

1. **Ex4**

Algorithm ***Factorial(n)***

**//The algorithm find the factorial of a positive integer n**

**//Input: n**

**//Output: sum of formula**

**fac = 1**

**for i 1 to n do**

**fac = fac \* i**

**return fac**

1. **Ex5**

Algorithm ***isDistinct(arr[0..n-1])***

**//The algorithm check whether all elements in an array are distinct**

**//Input: an array arr[0..n-1] of comparable elements**

**//Output: True if all elements in an array are distinct otherwise**

**s an empty set**

**for each element in arr do**

**add element to s**

**return True if length(s)= length(arr) else False**

1. **Ex6**

Algorithm ***FindMax(arr[0..n-1])***

**//The algorithm find the maximum element in an array**

**//Input: an array arr[0..n-1] and all all elements in the array are //unique**

**//Output: max element of an array**

**max first element of an array**

**for each element in arr do**

**if element > max**

**max element**

**return max**

1. **Ex7**

Algorithm ***Multiply2Matrices(matrix1, matrix2)***

**//The algorithm multiply two matrices**

**//Input: two multipliable matrices**

**//Output: a result matrix**

**result an empty list**

**if two matrices are multipliable do**

**for i row(matrix1) do**

**line an empty list**

**for j col(matrix2) do**

**sum 0**

**for k row(matrix1) do**

**sum = sum + matrix1[i][k] \* matrix2[k][j]**

**append sum to line**

**append line to result**

**return result**

**else**

**return ‘Dimension error: can not multiply’**

1. **Ex8**

Algorithm ***MultiplyMatrixWithNumber(matrix, number)***

**//The algorithm multiply a matrix with a number**

**//Input: a matrix, a number**

**//Output: a result matrix**

**result copy(matrix)**

**for i row(matrix) do**

**for j col(matrix) do**

**result[i][j] = number \* matrix[i][j]**

**return result**

1. **Ex9**

Algorithm ***Subtract2Matrices(matrix1, matrix2)***

**//The algorithm subtract two matrices**

**//Input: a matrix1, matrix2**

**//Output: a result matrix**

**result a zeros matrix with similar shape to matrix1**

**if shape(matrix1) = shape(matrix2) do**

**for i row(matrix1) do**

**for j col(matrix1) do**

**result[i][j] = matrix1[i][j] – matrix2[i][j]**

**return result**

**else:**

**return ‘Dimension error: can not subtract’**

1. **Ex10**

Algorithm ***Add2Matrices(matrix1, matrix2)***

**//The algorithm add two matrices**

**//Input: a matrix1, matrix2**

**//Output: a result matrix**

**result a zeros matrix with similar shape to matrix1**

**if shape(matrix1) = shape(matrix2) do**

**for i row(matrix1) do**

**for j col(matrix1) do**

**result[i][j] = matrix1[i][j] + matrix2[i][j]**

**return result**

**else:**

**return ‘Dimension error: can not add**