1. Differentiate between computability and complexity based on their objectives.

2. With a given example describe complexity theory and computability theory

3. Define the following with a given example

i. A set

ii. Power set

iii. Members of a set

iv. Subset

v. Proper subset

vi. Infinite set

vii. Finite set

viii. Unordered pair

ix. Union of a set

x. Intersection of a set

xi Complement of a set

xii. Difference of a set

xiii. Symmetric difference of a set

4. Consider the universal set U. U = {1,2,3,.....,9} A = {2,3,5,6,9}, B = {0,2,4, 8,....,20}, C = {1,3,5,...} and D = {1,2,3,4,5,6,7,8,9}. Find

i. B∩C, A∩D, A∩C

ii. B^c, D^c, C^c

iii. A⊕C, B⊕C, C⊕D, D⊕B

iv. A∪B, B∪D, C ∪D

v. A-B, D-C, C-B

5. Define the following

i. Alphabet

ii. Words

iii. Length of a word

iv. Substring

v. Initial segment

vi. Concatenation of strings

vii. Language

6. Suppose A = {a,c,e,i,o,n,t} with strings v = concatenation and u = catetioncona

i. Find the length of word in v

ii. Concatenate v and u

iii. Reverse string u

iv. Find substring of string m, if m = cacontatione

7. Consider the given word s = abracaba

i. Find the substring of s

ii. find the initial segment of s

8. Draw the Venn diagram to represent the following:

i. Let the set START – t be the set of all English words that start with letter “t” and the set END – z of English words that ends with “Z”.

ii. The intersection between words that start with “t” and the ones the ends with “Z”.

iii. The union of the words that start with “j” and ends with “Z”.

9. Given the following set of strings chosen from ∑^\*, L⊆∑^\* and ∑ = {0,1}. Find the Language that generates

i. Set of all strings containing 11 in it

ii. Set of all strings starting with 0 and ending with 1

iii Set of all strings with length 2 iv. Set of all strings ending with 10; and

10. If A denotes an alphabet, A^\* is a set of strings over A. Find the possible alphabet for the following languages:

i. L={en,nouh,ugh}

ii. L={bear,rear,2200}

iii. The language of all binary strings.

11. Given the following set of strings chosen from A^\*, L⊆A^\* and A = {o, p}.

Find the Language that generates i. Set of all strings ending with p ii. Set of all strings with equal no of o’s and p’s; and

iii. Set of all strings starting with p and ending with o

iv. Set of all strings with length 4

12. Let ∑ = {a, b}, describe the Languages that produce the following set of strings.

i. {b^n a| n>0}

ii. { a^n ab^m | n≥0,m>0}

iii. { b^n b^m a^m | n≥0,m≥0}

iv. {ab, abb, aab, aaa, abbb,.....}

13. Let A = {a, b}, describe the Languages that produce the following set of strings.

i. {b, ba, bab, baa, bbb,.....}

ii. {bab, babb, bbab, bbabb, babbb, bbbab...}

iii. {b, bb, bbb, bbbb, ....}

iv. {ab, aabb, aaabbb,.....}

14. If L,L1, L2 are languages over ∑ , then define the following operations: .

Union .

intersection .

Difference .

complement .

Positive closure .

star operation .

Multiplication .

Power

1. The goal or objective of computability theory is to determine which problems, or classes of problems, can be solved in each model of computation. It is also concerned with the classification of problems by difficulty

WHILE

The complexity of a problem or the objective of complexity theory is to describe whether a problem can be solved using algorithms, and how much resources (in form of time and space) it will take to solve a problem algorithmically

2. Computability theory deals primarily with the question of whether a problem is

solvable at all on a computer. The statement that the halting problem cannot be

solved by a Turing machine is one of the most important results in computability

theory, as it is an example of a concrete problem that is both easy to formulate and

impossible to solve using a Turing machine. Much of computability theory builds on

the halting problem result.

Complexity Theory is part of the theory of computation dealing with the resources

required during computation to solve a given problem. The most common resources

are time (how many steps does it take to solve a problem) and space (how much

memory does it take to solve a problem). Other resources can also be considered,

such as how many parallel processors are needed to solve a problem in parallel.

Complexity theory differs from computability theory, which deals with whether a

problem can be solved at all, regardless of the resources required.

3.

I) A set is a group or collection of related objects or numbers, considered as an entity unto

itself. Examples include the set of all students in a school, the set of cars in a parking lot.

(ii)  power set of any set S is the set of all subsets of S, including the empty set and S

itself, variously denoted as P(S),(S). Eg If S is the set {x, y, z}, then the subsets

of S are

{}

{x}

{y}

{z}

{x, y}

{x, z}

{y, z}

{x, y, z}

and hence the power set of S is {{}}, {x}, {y}, {z}, {x, y}, {x, z}, {y, z}, {x, y, z}}

(iii) members of a set: elements that makes up a set. For example, {x,y,z} are the elements in set S.

(iv) Subset: is a set A is a subset of a set B, or equivalently B is a superset of A, if A is contained in B. That is, all elements of A are also elements of B. A and B may be equal.

(v) Proper subset : A proper subset of a set A is a subset of A that is not equal to A. In other words, if B is a proper subset of A, then all elements of B are in A but A contains at least one element that is not in B.

(vi) Infinite set:  set is said to be an infinite set whose elements cannot be listed if it has an unlimited (i.e. uncountable) by the natural number 1, 2, 3, 4, ………… n.

(vii) Finite set: a finite set is a set that has a finite number of elements. Informally, a finite set is a set which one could in principle count and finish counting.

(viii) Unordered pair:  is a set of the form {a, b}, i.e. a set having two elements a and b with no particular relation between them.

(ix) Union of a set: The u un ni io on n of two s se et ts s A and B is the s se et t of elements which are in A, in B, or in both A and B.

For example, if A = {1, 3, 5, 7} and B = {1, 2, 4, 6, 7} then A ∪

B = {1, 2, 3, 4, 5, 6, 7}.

(x) Intersection of a set:  i in nt te er rs se ec ct ti io on n of two sets A and B, denoted by A ∩  B, is the

set containing all elements of A that also belong to B (or equivalently, all elements

of B that also belong to A). For example, The intersection of the sets {1, 2, 3} and {2,

3, 4} is {2, 3}.

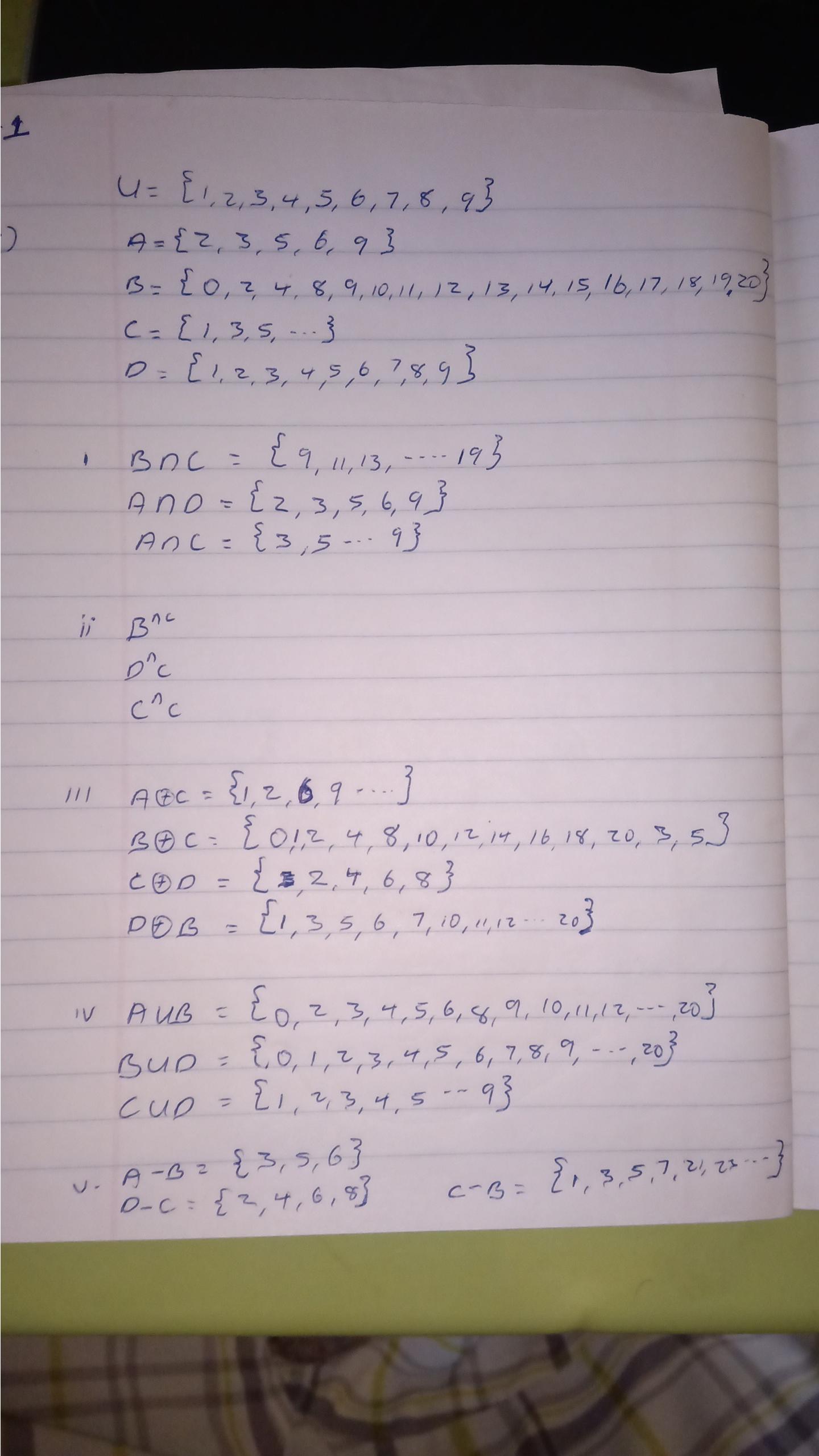
(xi) Complement of a set: refers to elements not in A. When all sets under consideration are considered to be subsets of a given set U, the absolute compliment of A is the set of elements in U but not in A.

(xii) Difference of a set: The difference of two sets, written A - B is the set of all

elements of A that are not elements of B. The difference operation, along with union and intersection, is an important and fundamental set theory operation.

(xiii) Symmetric difference of a set :  is the s set of elements which are in either of the sets and not in their intersection.

4.



5. An alphabet is a finite non-empty set. The elements of an alphabet are called

the letters or symbols of the alphabet.

(ii) Words: Suppose a set A is finite and A is viewed as a character set or an alphabet. Then a finite sequence over A is called a string or word

(iii) Length of words: Length of a word is denoted as |w| and is defined as the

number of positions for the symbol in the string.

(iv) Substring: is a contiguous sequence of characters within a string.

(v) Initial segment: initial segment is often referred to as the set of strictly preceding elements.

(vi) Concatenation of strings:

Let w1 and w2 be two strings then w1w2 denotes their concatenation w. The

concatenation is formed by making a copy of w1 and followed by a copy of w2.

For example w1 = xyz, w2 = uvw

then w = w1w2 = xyzuvw

(vii) Language: A language is a set of strings, chosen from some Σ\* or we can say- ‘ ‘A

language is a subset of Σ\* ‘ ‘. A language which can be formed over ‘ Σ ‘ can

be finite or infinite.

6. A = {a,c,e,i,o,n,t}

v = concatenation

u = catetioncona

(i) the length of v= 13

(ii) reverse string u= anocnoitetac

(iii) concatenate v and u= concatenationcatetioncona

(iv) substring of m if m=cacontatione

∑0={λ}

∑1= { c,a,c,o,n,t,a,t,I,o,n,e}

∑2= {ca,ac,co,on,nt,ta,at,ti,io,on,ne}

∑3= {cac,aco,con,,ont,tat,tio,ion,one}

∑4={caco,acon,cont,onta,tati,atio,ione}

∑5={cacon,acont,conta,ontat,ntati,tatio,ation,tione}

∑6={cacont,aconta,contat,ontati,ntatio,tation,atione}

∑7={caconta,acontat,contati,ontatio,ntation,tatione}

∑8={cacontat,acontati,contatio,notation,ntatione}

∑9={cacontati,acontatio,contation,ontatione,}

∑10={cacontatio,acontation,contatione}

∑11={cacontation,acontatione}

∑12={cacomtatione}

∑= { λ,c,a,c,o,n,t,a,t,I,o,n,e, ca,ac,co,on,nt,ta,at,ti,io,on,ne, cac,aco,con,,ont,tat,tio,ion,one, caco,acon,cont,onta,tati,atio,ione, cacon,acont,conta,ontat,ntati,tatio,ation,tione,cacont,aconta,contat,ontati,ntatio,tation,atione,caconta,acontat,contati,ontatio,ntation,tatione, cacontat,acontati,contatio,notation,ntatione, cacontati,acontatio,contation,ontatione, cacontatio,acontation,contatione, cacontation,acontatione, cacomtatione}

7. substring of s={abracaba}

(i) ∑0={λ}

∑1={a,b,r,a,c,a,b,a}

∑2={ab,br,ra,ca,ab,ba}

∑3={abr,bra,rac,aca,cab,aba}

∑4={abar,brac,raca,acab,caba}

∑5={abrac,braca,racab,acaba}

∑6={abraca,bracab,racaba}

∑7= {abracab,bracaba}

∑8={abracaba}

∑= { λ, a,b,r,a,c,a,b,a, ab,br,ra,ca,ab,ba, abr,bra,rac,aca,cab,aba, abar,brac,raca,acab,caba, abrac,braca,racab,acaba, abraca,bracab,racaba, abracab,bracaba, abracaba}

8.

9.

10.

11.

12.

13. {b^n a^m b^p|n>0,m>=0,p>=0}

(ii) {b^n a b^m|n>0,m>=0}

(iii) { b^n|n>0}

(iv) {a^n b^m| n>0,m>0}

14