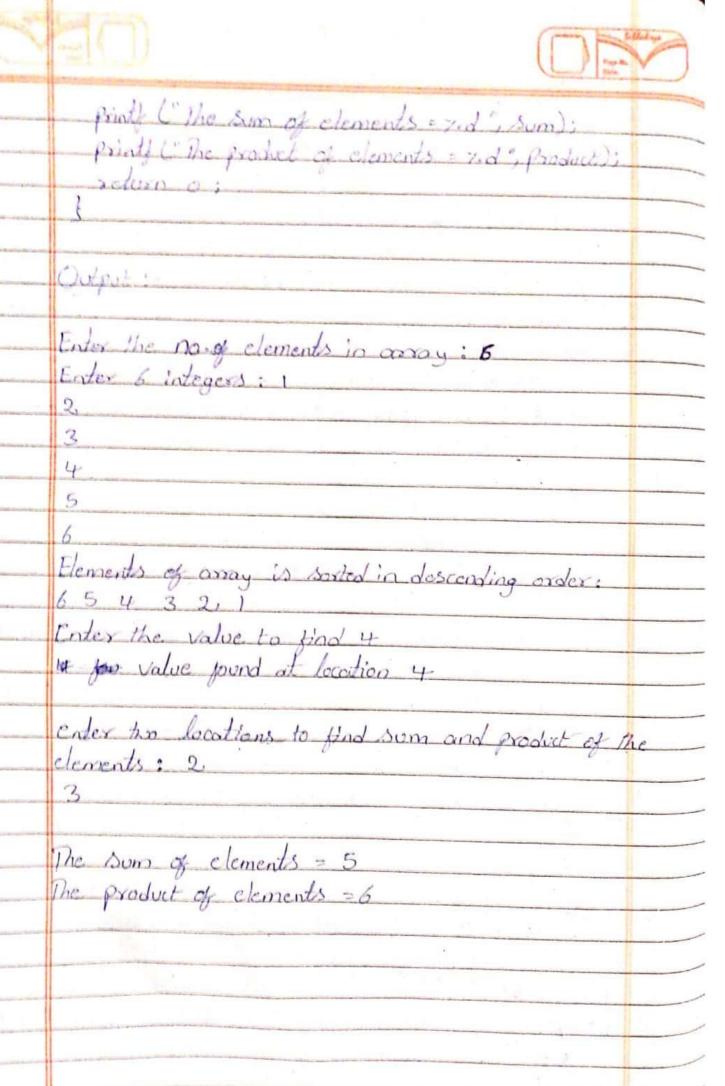
	-
luke the clements from the user and sort them	in_
debeending wider and do the following	
Dustry binny search find the clement and the	location
in the array where the dement is asked from	USex.
DISK the user lo enter any two locations print	
and product of values at those locations in the	
orsay	
Crogramis	
1) include & stdio.h >	
int male (.)	
intil las bink mid in the author to	
int i, low, high, mid, restray, arx [100], top, i,	one, 12
print ("enter the no-of elements in array: ");	
Scanf (" " do, An);	
and Canton of Colors	
frintf ("enter 1.d integer:", n);	
1px (1=0 \$ 120 \$ 1+1)	
pr(ico) izris itt)	-
par Cuitisiansitt	
1/ Carreiz carres).	
tmp = ary [1];	
arr Lil = arr Lil s	
an Lij : temtup;	
}	
ζ Scanned with	1

```
prints C'clements of army is sorted in descending outs: );
  Kinf ("1.d", an (1);
printf ("enter value to knd:");
 Scant ("7.2", 4 Key);
 high = n-1;
 mid = Clan + high)/2;
 While (low = high)
   if Carr Emid 7 > Key)
   else if Carrimid) = = Key)
     Printy ("Value pound at location Yed", Key, midths
   high = mid - 1;
   mid = (lan + high)/2;
  if (law shigh)
   Printy ("not found in the list", Key);
   Print ("\n");
   Printf ("coder the location to find the Dum and product :"
   Scanf ("1.d", dans);
   Scanf (" ted", 4 tho);
   Sum = (ary [anc] + arr Etwal);
   Product = (arr [enc] * arr [two]);
```

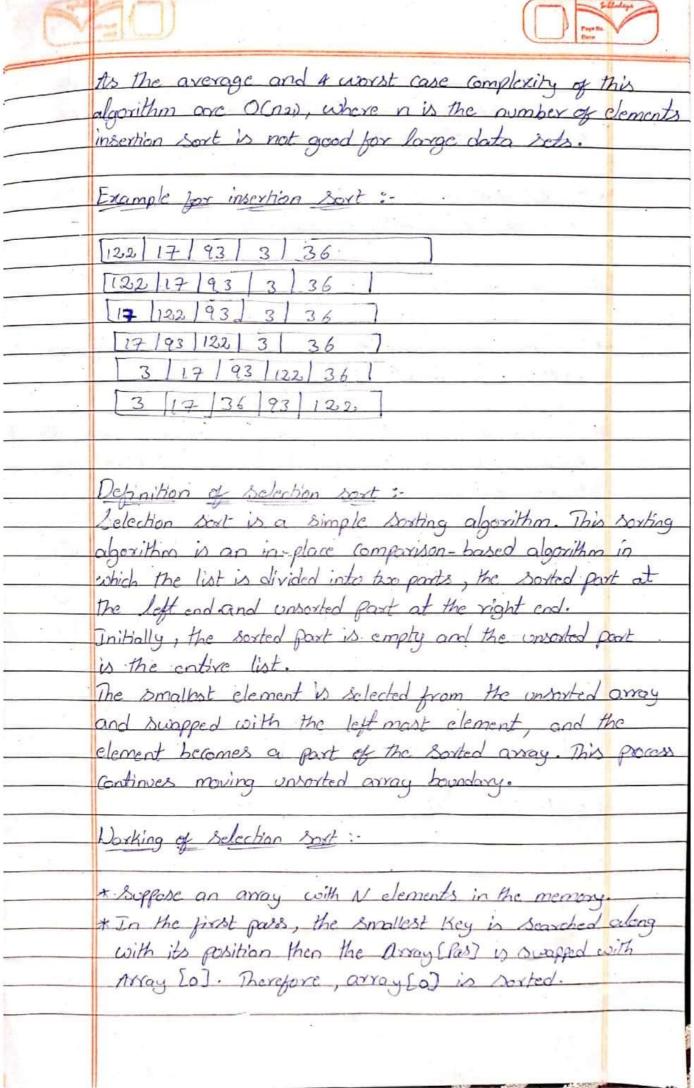


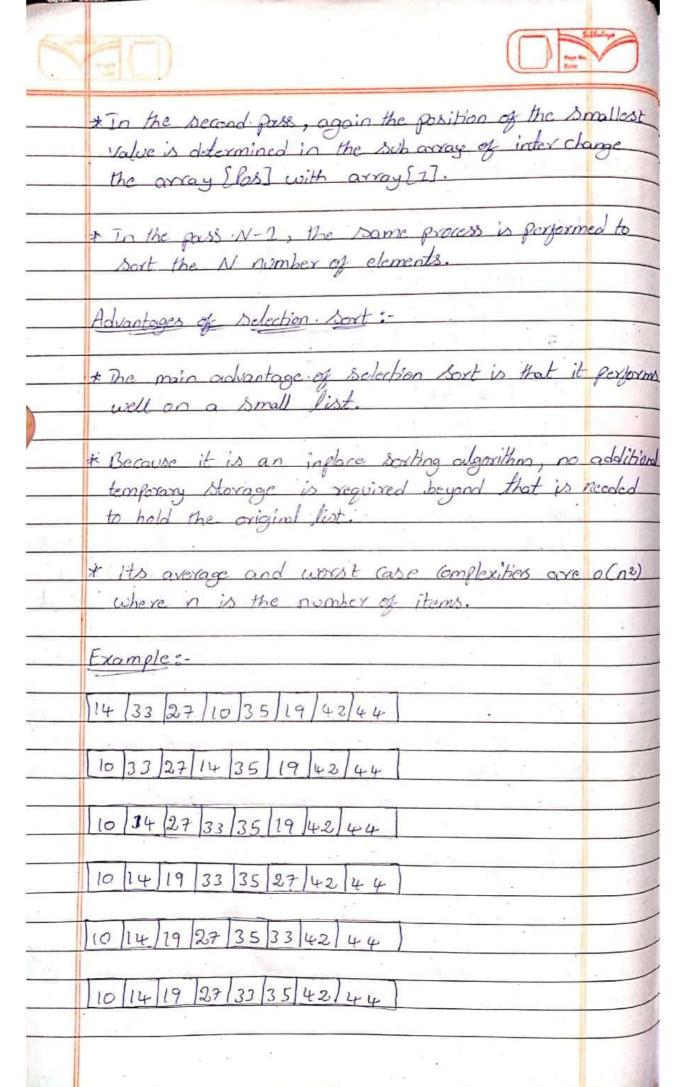
CHU
2) but the array rains major but where changes one
Cake John Se was the pine the property of the character
poor pirat and last when it is the your
<u>figuran</u>
Include of Afolians >
It define Most - See 5
Used Marse Sout Clot lat)
Word more - associat int int int);
int on- sout Imax - size?:
int main ()
int 1, K, (m = 1)
Printly C Simple merge best example general and any it is
tox (i to x in max size : i+)
Stank (" 7.2" karr - Sout [1]):
mints (" your obto : ");
jox (i = 0 : 12 mx = she : ivx)
print ("red", one Nort ED:
merge_Sout (o, mox_Size=D)
Bigth (Souted data : 2)
tor (i=0; ic max-size; izy)
1
Bridge ("Vad", ar Sort Liddi
And the second s
Prints (Find the product of R" elements from past and livering
fre - arr - Aret [K] arr - part I max - 2 - 2 - 2 - 2 -
Grinty C product = 1100 ", Gods

9	getch ();
A 4	}
	void merge-sort (int; inti);
	1
0	int m;
	if (iei)
	m= (i+i)/2;
	merge_Sort (i, m);
	merge-soxt (m+1, j);
	merge_array(i, m, m+1);
	ł
	3
	Void merge array (int a, int b, int c, int d)
i a	{
N 2	int [50];
	int i=a, i=c, K=0;
	while (iz=b dd iz=d);
	\
	if (arr-Sort [i] carr-sort [i])
	¿[K++]= arr-sort[i++];
	else
	t [K++] = arr-sort[i++7;
3	3
	while (i c=b)
-	FSK++]= arr-sort [i++];
	While (jz=d)
	E[K++] = arr-Sort[i++];
	for (i=a; j=0; i <=d; i++; j++)
	arr_sort[i]=t[j];
	The state of the s

O	ar and the second secon
	Output:
	Sample merge sort example punctions and array
	Enter 5 elements for sorting: 1
	3
	5
	Jour data: 5432, 1 Sorted data: 12345
	find the product of K" elements from first and last where
, 1	Product = 20
	Discuss the insextion sort and selection sort
Ans:-	Definition of insertion sort:
	Insertion sort is a simple sorting algorithm that builds the final sorted array (or list) one item at a time.
•	advanced algorithms such as quick sort, heap sort or merge
•	Insertion sort works by inserting the set of values in
	The primary concept behind insertion sort is to insert elements
	The insertion method saves the effective amount of memory.
	Scanned with CamScanner

		C love
**	Advantages of insertion sort	
3	* Simple implementation: Jon Bentley should evertion a version, and a five-line of	himized version.
	* Effective por (quite) small data sets, suc	h like other
	* More efficient in practice than most other algorithms such as selection sort or bu	simple guadratic
	* Adaptive, i.e., efficient for obta sets that sets substantially sorted: The time comp when each element in the input is no mo away from its sorted position.	ove already dexity is O(Kn) we Han K places
	* Stable i.e., does not change the relative or with equal Keys.	der of elements
in a	Working and Complexity of insertion sort	
	In each iteration, it compares the correct of values in the sorted array. If the correct greater than the element in the array, the	nt element is
	element and iterates to the next away ele if the current element is smaller than the then it moves the vest of the element in the position and makes space for the current	array element
1 1 1 1 1	This is how insertion sort takes one input time, iterates through the sorted sub on	elements at a
. 74	iteration it inserts one element at its of this is why the algorithm is known as	1





	l Drus
4)	Bost the array using hibble sort where elements one
	taken from the user and display the elements
	(i) An alternate order
	(ii) som of elements in odd positions and product of element
	in even position
	(iii) Elements which are divisible by where m is taken
	from the user.
	Grogram:
	# include < stdie.h>
4	# include < conjo.h >
	int main ()
	<u>{</u>
	int and sol, i, i, n, temp, sum=0, Bodut = 1;
	Brinty C'enter total no-of elements:");
	scanf (" 1,d", 4n);
	bioty ("coter yed elements:");
	por (i=0; icn; i++)
	scanf ("y.d", Karr [i]);
	printf (" soiting away using bubble soit technique: ");
	for(i=0; i=(n-1); i++)
	<u> </u>
	for (i=0; i=(n-1-1); i++)
1	
	if (arr[i] > arr[i+1])
	<u>{</u>
-	temp = arr[i];
	arr[i] = arr[i+i];
Ti .	arrsitil = temp;
2	5
	3
	1

YEU	Tree to	<u></u>
Brinty ("all array elements	sorted successfully ");	1476
Brinty ("array elements in		
bor(i=0; i<0; i++)		S
S	180	1 2
Brinty (" yed In", arr	:7);	1
3		
prints ("array elements i	n alternate order (");	
tor Ci=o; i c=n; i=i+	2,)	
Į.		1 1
Printf(" y.d \n", arrsi	7):	
}		
20x (i=1; i = 0; i=1	(1)	
\$	T-20,)	
8 1 1 5:-	7 -	only fill
Som = som + arr [i]	15	-0.00
0:4:00	La company	
frinty ("The sum of oda	psitions element are 2401	n' num);
por (1=0; ic=n; i=i	+2,)	- na
2	*	P-1 G P-2
Brochet = arx [i]:		
0	9	2 13
exintly (" The froduct of a	ven position elements = yed	n" Brodu
ouch ())		,,,,,,,
return ();		180
5		. 3
Output:-		
enter total noon eleme	ents = E	-
enter 5 elements: 1	442 0 0	
2		
3		
4		
5		- ,9
Section and		
Sorting array using bubb	de sort technique:	

C	
	All array elements sorted successfully
	Array elements in ascending order
	2
	3
	4
	5
	array elements in alternate order
	1
	3
	5
4	The sum of odd position elements are = 6
*	The product of even parition elements = 15
5)	Write a recursion fragram to implement binary search?
	2
	togram:
	# Include Carlo.h>
	#include Catalib.b>1
	void binary search (intark), int num, int first, int last).
7.	}
	int mid;
	if (first > last)
	{
	Printf ("Number is not found");
	3
J	else
	\{
	mid = (birst + last)/2;
/	if (arr [mid] == num) {
/	prints ("clements is foundat/d", mid);
1	

1		<u> </u>
	exit (e);	
	£ (0.2)	
	elseif (arr[mid] > num) {	
	Birary search Carr num, first, mid-D;	
	3	
1	clses	
	Binary search (avr, num, mid+1, last);	
	3	
	3	
	5.	
	Void main ()	
	2	
-	int arr [50], beg, mid, end, i, n, num;	
	Brinty (" enter the size of an array :");	-
	scanf (" y.d", an);	
	frints ("enter the values in socied sequence");	
-	pr (i=o; ian; i+t)	10.1
	Scanf ("1.d", Varisi);	
-	3	
	beg =0;	146
	end = n-1	
	Printf ("enter a value to be search:");	
	Scanf ("-led", 4 num);	
8	Binary Search Carr, num, beg, end);	
-	9	_
		_
	Output:	
-	enter the Size of an away: 5	
-	ender the values in sorted sequence	
-	2,	1
-	3	1
	4	
	5	

