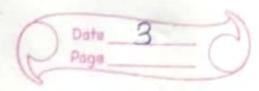
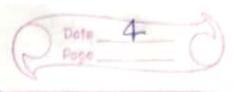


int binary (intal), int b, inta) SELECTION OF THE SECURITY WAS A inti=0, j=n-1, mid; while (ix=j) { mid=(i+j)/2; if (a [mid]==b) return mid +1; elses if (bca[mid]) J=mid-1; else if (12j) = { returnio; int main () { int n, i, 9[20], c, b, 5, 52; printf("Enter the number of elements of any); Scanf ("10(.d", for); printf("6nter the elements of army");



for (i=0; i<n; i++) Scanf(110/0d), falil); Sort(a,n); for (i=0; i<n; i++) printf ("106", a8i]); printf ("Enter the element to find in away"); Scanf (" (6)", 2b); C=binary(a,b,n)3 if (c!=0) } printf ("element is found at position % d2", c); else? printf("Elentent not found in"); printf("Enter the position of array to find sum and product in"); Scanf ("%d'6d", + 8, 452); 5,--3 printf ("The sum is % d', a [s,]+a[s,]); printf ("The Product is % d", a [s,] * a [s,]);



2. Sort the array using Merge sort where elements are taken from user and find the product of kth elements from the first and last where k is taken from the user.

```
#include <stalib+h>
#include<stdio-n2
Void merge (int arri], int I, int I, int c, inte)
int i, j, K;
int m1= C-1+1;
int 772= e-C3
int L(n1], REn2];
for (i=0; i<n; i++)
22:] = apr [6+1];
for (j=0; j<n2; j++)
 R[j] = 900 [C+1+j];
(=03
7=03
```

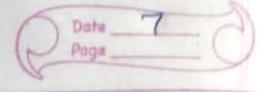
while (icn 28 j cn) }

if (LTi] <= RTj]) ?

```
mat ()
                                                                                                                    Data 5
  and [K] = 2 Bi];
   c++ 9 ?
   3 the set of the sale and shot when
  elses
  an [k]=R[j];
   gitt;
                    The second secon
    K++;
                                                                                         28 - 2 - 3 +0
   while (i< ny) {
    arrelej=LEij;
                                                                                     A THIN THE THE
    C++ :
                                                                                            ADERSON FO
                                                                         A PRODUCT LOS USA
    1c++;
                                                                                     のもとっているいちい
   while (j <n2) {
                                                                                        BELLIA TAND EN LIST
    aro [k] = R[j];
                                                                          (HESON TREES NO
                                                                               DESTINATION FEEDS
   7++3
Woid mergesort (int aro [], int1, inte) {
   int m= 1+(e-1)/2;
```

Scanned with CamScanner

```
merge Sout (000, 1, c);
mergesort (am,c+1,e);
merge (arr, 1, c, e);
      to ( also we have public as the second of the second
               MENT WASHINGTON
Void print Array (int AR7, intsite) {
inti:
for (i=0; i < size; i++)
Printf ("%d", ASID;
mntf ("In");
int main ()
int arr [T];
int ans_size = Size of (ans)/size of (ans col);
for (i = 0; icam size; i++) {
printf ("Enter the elements:");
 Scanf ("101.d", 4000[i]);
Printf ("Given among is ln");
printhogy (and, and_size);
mergesort (arr, D, arr_size-I);
```



printf ("I') n Sosted amony is In");

printf ("am, am-size);

int k;

printf ("anterthe value of k:");

Scanf ("all, 4k);

int fromfirst = amore-1];

int fromlast = amore-1];

printf ("al", fromlast * fromfirst);

return 0;

y

Output

Briter the elements: 65

Enter the elements: 98

Enter the elements: 32

Enter the elements: 25

Criter the elements: 15

Criter the elements: 46

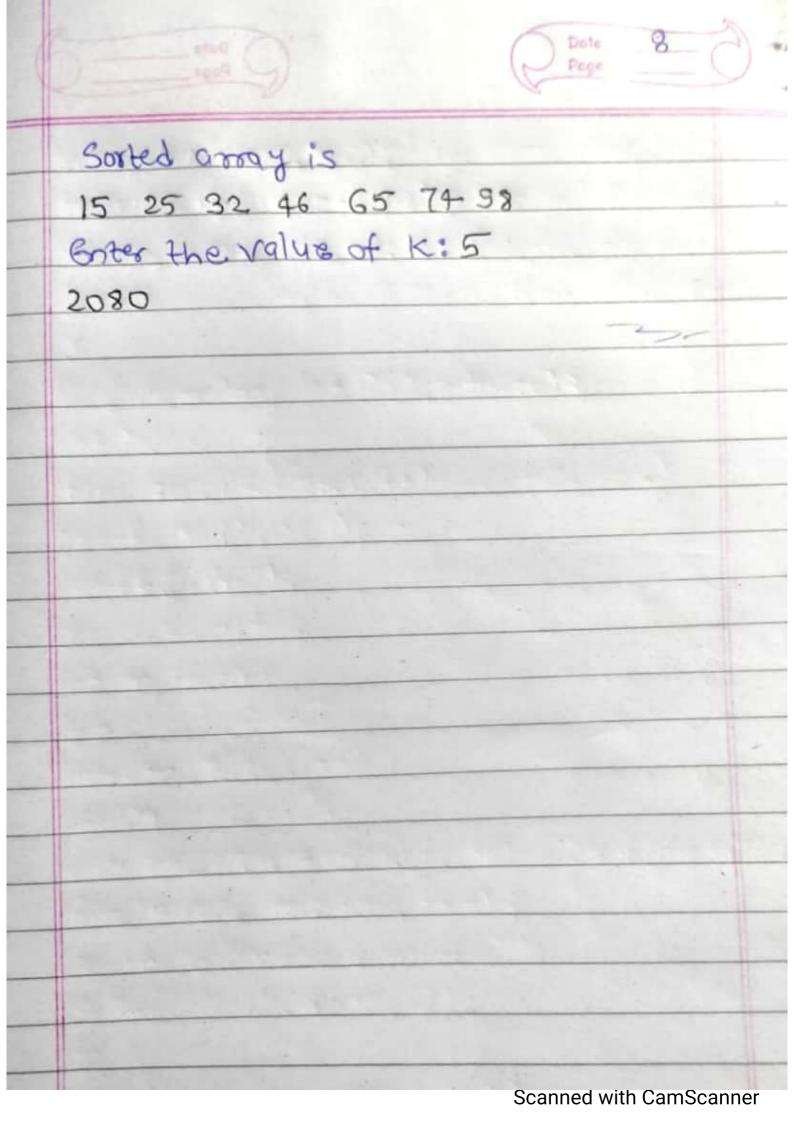
Enter the elements: 74

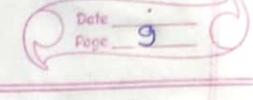
Criven array is

65 98 32 25 15 46 74

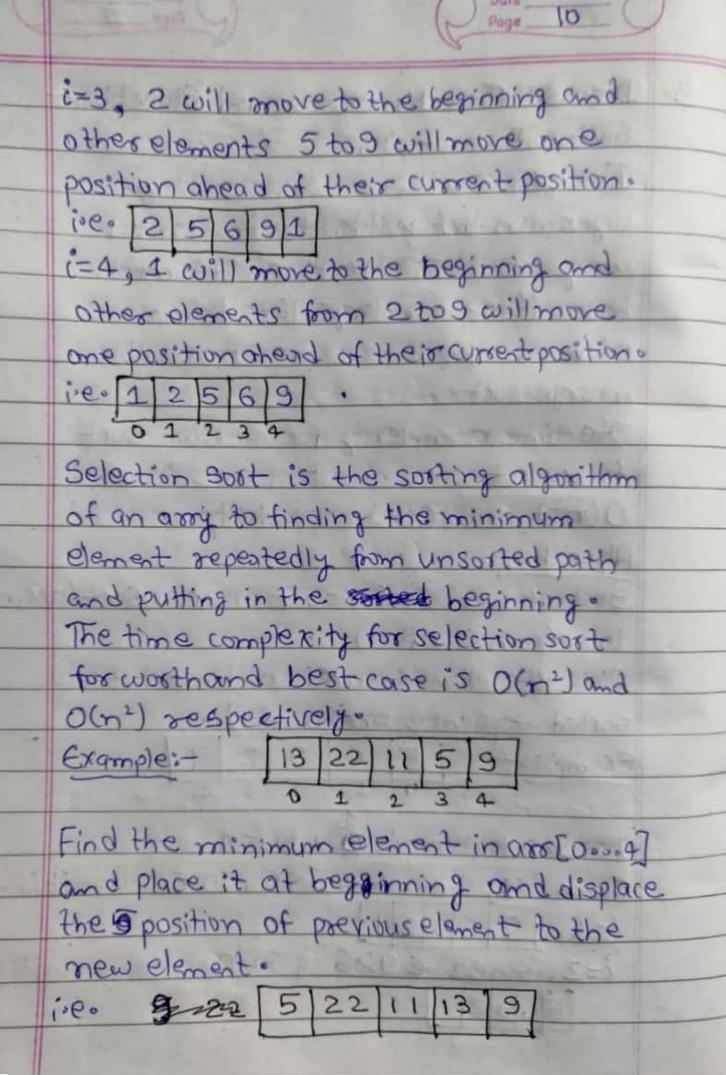
Stateda

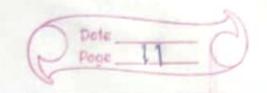
Scanned with CamScanner





3. Discuss Insertion sort and Selection Soft with examples. Insertion sort is a simple sorting algorithm that builds the final sorted array one item at a time. It is much less efficient on large lists than more advanced algorithms such as heapsout, merge sout. The time complexity for insertion sort for worst case and best case are O(n2) and O(n) respectively. Algorithm Sout of arril I of size n insertions ort (arr, n) loop from i= 1 to n-1. Pick the element aroli] and insert it into sorted sequence aro [0...i-1]. Example: - 5 6 9 2 let us loop for i= a to i= 4. Bines i=1, Since 500 5<6,50 5 will as it is. 6 5 6 9 2 1 i=2, since €9>6, so 6 will remain in its position





Find the minimum element in arr [1.4] and place it at beginning of arr [1.4].

[5 9 11 13 22

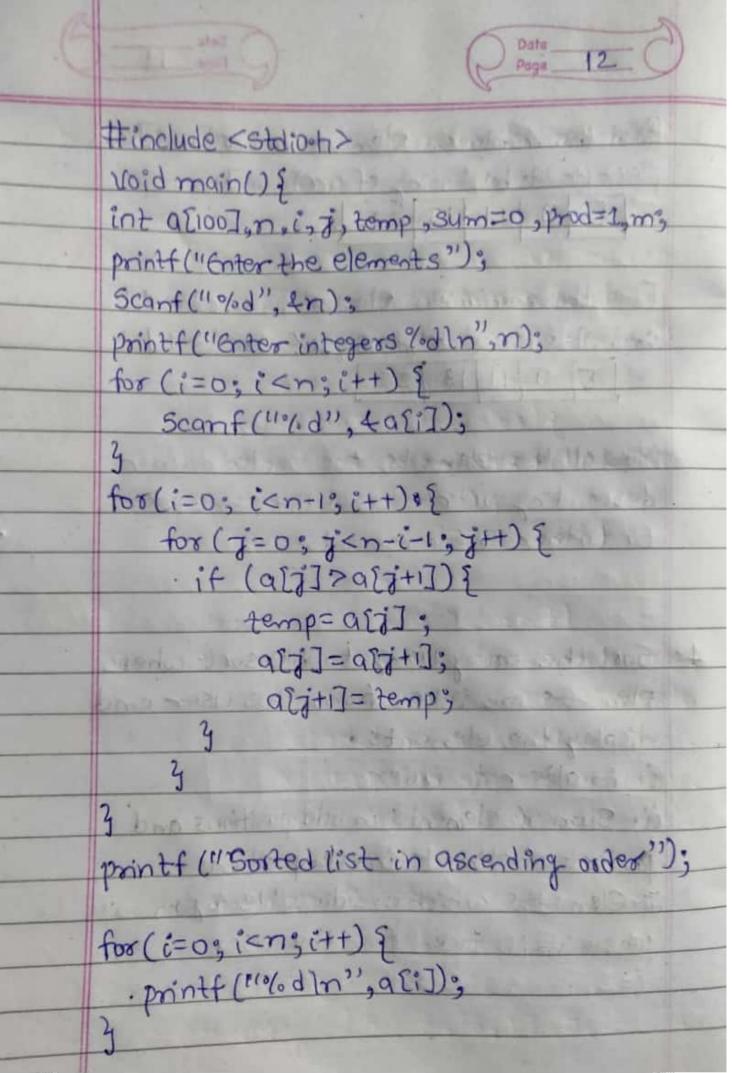
Find the minimum element in arr [2.4] and place it at beginning of arr [2.4].

Now all the elements has settled down in descending order. Repeat the same process for further reasonngements.

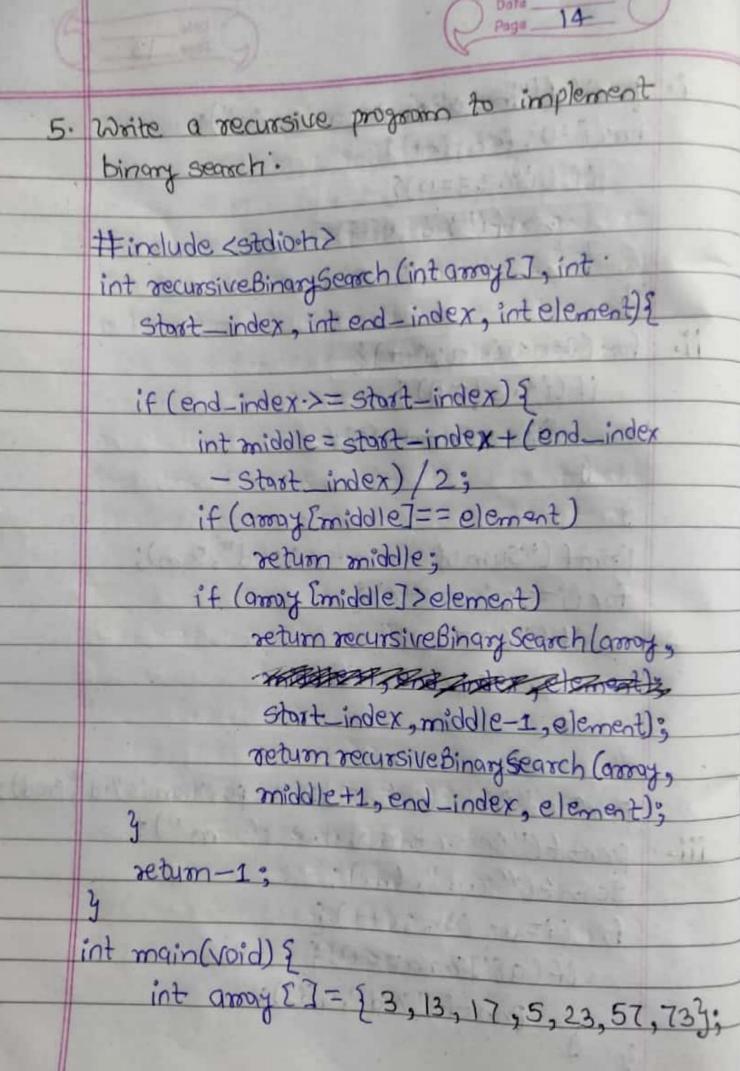
4. Sort the array using bubble sort where elements are taken from the user and display the elements.

i. in alternate order.

ii. Sum of elements in odd positions and product of elements in even positions iii. Elements which are divisible by m where mis taken from the user.



```
i. printf ("The alternate order is");
          for(i=0; i<n; i+t)?
                        if (i%2==0)9
                              printf ("10/0d", a(i));
                                    day mad the house who have to
                         white the company of the second secon
           for (i=0; icn; i++) {
                         if (i°62 (=0) {
                                    Sum= Sum+ali];
                                                    richting white
                4 - I tapas to Establish to be
                  Printf ("Sum of odd index is "/od", Sum);
                   for (i=0; i<n; i+t){
                               i+ (ib/.2==0) }
                               prod= prod*a li];
                    printf ("Product of ofd even position is %d", prod):
                     printf ("Enter the value of m");
iii.
                     Scanf (" (.d", 4m);
                    for (i=0; icn; i++);
                                 if (ali]%m==0) {
                                3 Printf ("%d", a [:]);
                                                                                                                            Scanned with CamScanner
```



Scanned with CamScanner

