```
13/6/24
    Lab-7
            Johnson torther
                                                      ( ++ p-all
  # include cstdio. W
                         3 de 1} ((d(anci)-1]==1) &6 1/= nun-1
  # include <stallib. h>
  (as (i) > av (i+1) & c con (i) = alidous < (i) mos > (i+1) vo < (i) mos )
   ind flag = 0;
           int t = *a;
                                    : Didour = 9 - Widger
                                                 Jelor 5
           *b = t;
 3
 int search (int arr(), int num, int mobile) {
              int 9;
          for (g=0; g<num; g+t) {
       seturn g + 1;
                                          Clas altern mobile;
            else &
                  flog tt;
                           voil formulations (in auc), intol (), intrum ) ?
          suturn -1 )
                           led weble - lind- Mobile (and, ours);
                            it pos = search (an num probili):
it flut - Mobil (is arr(7, is d(7, int num) {= (1-(1-a)) was b) !
           ind mobile=0;
           int mobil -1=0;
                            and (dan (100-1), dan (10-2));
                            and (com (you -1), & an (you);
        for (1=0) i < num; 1++) {
                    if ((d (ara G)-1] ==0) ee 1:1=0) { as G7
                 if (are (1) ) are (1-1) fe are (1)> mobile P) {
                       mobile-P=mobil;
```

```
13/6/26
                                                                  d Care (i) -1)=1;
       clas 5
                                                                   else
         flag tt;
                                                                    d Car (1)-1=0;
       3
    3 classif (Cd(ara (1) -17==1) &6 11= num-1) {
            if (are (i) > are (i+1) se are (1) > mobile - P) ? toi) quous bide
                                                                   for (111 =0; icr
                 mobile = arr (i);
                 mobile-P=mobil;
                                                                       ("1.d"
               3 clse {
                                                                        ("\n") | third
                  flag ++ )
                                                                    3
                            int search (int are(), int rum, int nobile) ?
             3 else {
                                                                     int factorial (int
                  flag + t;
                                                                             14/=
                              for (9:0) 8 com : 9+1) {
                                                                            for City
                                il (an (3) == mobile)
          (mobil -P==0 le mobil ==0) seturo;
          else altum mobile;
                                                                                 9re
                                             ( ++ post
voil permutations (int are CT, int dCT, intrum ) {
                                                                            int mo
          ind mobile = (und - Mobile Cand, rum);
          int pos = search (arr, num mobile);
        if (d (au [pes-1)-1]=20) munting (3/2 to 1) (3/20 to) (3/20)
              swap (dan (por-1), Lan (por-2));
        else
         swap (lar (pos-1), & an (pos));
     if (an (1) > mobile) ?
                  if (d(anti)-1) = 20)
```

```
d Care (i) -1)=1; (3 "n/b) = returned lest ") (b)
             prid ("All possible permitation an: 14"); es asser
   else
        d (an (1)-1=0;
                                      for (it i=0; icrum; i+1) [
        , ato
                                                  803=0;
   for (11 i=0; 1 c num; 1++) {
                                      (C(1) no "61" ) ) bessel
       print (" 1.d", an (3);
      wint ("In");
 3
 int factorial (int k) {
        1nt (=1)
                                                       notion or
     for (intiel; ick+ol; 1++) {
      setun f;
                                                Enter them number:
                                                 Total psychottons :
int main () {
                                       all pessible permutation an:
      int rum = 0;
      porint ( " Tohuson Teather algorithm to find all permitations of
                ghen number (n");
     point (" " but the newber: ");
     Scanf ("/d" & nem);
   1st an (new ] d (new);
      int 2= factorial (num);
```

```
forit ("Total permutation = %d \n", Z);
 wind ("All passible permutation are: \n");
                                           0=1-(1) NO) k
                                                        Obi,
  for (inti=0; i < num; i+1) {
         d (i) = 0;
                               3 ( TT 1 ( man > 1 ( 0= 1 td) 10)
        auci) = i+1;
                                    prod (170 , an 03)
        with (" "/d", an (i));
                                                fred ("In")
   print ("\n");
   for (it i = 1; s < z : s++) {
                                                int lautotral (int 10) 5
        permutations (an, d, mm);
                                                   (1= ) to 1
                                  for (int)=1; i=k+01; i+t) {
      retuno)
 Owhat :-
 Ender then number: 3
Total pentations : 6
                                                       ing marker ()?
 All possible permutation an:
                                                it num = 0;
      1 23
    print (" Toluna Teath algorithm to had all forms sions of
                                 given number ( ");
                                point (" late the number: ");
                                        Dan [ ( " ) of if prom);
                                       Is on (now ) of (now):
                                      int 2 - factoral com);
```