```c
// Last Change: 2013-06-25 22:22:55
#include <errno.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void printv(int v[], int n);
void swap(int *x, int *y);
int next(int v[], int n);

int main(void) {
  int *v; /* This is the array */
  int i, size_of_arr; /*loop terminator*/
  printf("Know the next permutation.\nhow long your array is (no. of elements):\n");
  scanf("%d",&size_of_arr);

  v = (int *)malloc((size_t)size_of_arr * sizeof(int));
  if(v==NULL) {
    fprintf(stderr, "\ndynamic memory allocation failed\n");
    exit(EXIT_FAILURE);
  }

  printf("input the elements with spaces:\n");
  for(i=0; i<=size_of_arr-1 ; i+=1) {
    scanf("%d",(v+i));
  }
  printf("\n");
  /* P1 */

  next(v,size_of_arr);
  printf("returned to main(). Next permutation is:\n");
  printv(v, size_of_arr);

  free(v);
  v = NULL;
  system("PAUSE");
  return 0;
}

void printv(int *v, int n) {
  /*just to show off the last filled array, from main*/
  int i;
  for(i = 0; i < n; i++) {
    printf("%d ", *(v+i));
  }
  printf("\n");
}

void swap(int *x, int *y) {
  int tmp;
  tmp=*x;
  *x=*y;
  *y=tmp;
}

////////////////////////////////////////////////////////////////////////
```

```
    ////////////
58  /*!
59    Generates the next permutation of the vector v of length n.
60    @return 1, if there are no more permutations to be generated
61    @return 0, otherwise
62  */
63  int next(int *v, int n) {
64    /* P2 */
65    /* Find the largest i */
66    int i = n - 2; int k; int j;
67
68    /*[> NOTE: stage A <]*/
69    while((i >= 0) && (*(v+i) > *(v+i+1))) {
70      --i;
71    }
72
73    /* If i is smaller than 0, then there are no more permutations. */
74    /*[> NOTE: stage B <]*/
75    if(i < 0) {
76      return 1;
77    }
78
79    /* Find the largest element after vi but not larger than vi */
80    /*[> NOTE: stage C <]*/
81    k = n - 1;
82    while(*(v+i) > *(v+k)) {
83      --k;
84    }
85    /*[> NOTE: stage D <]*/
86    swap((v+i), (v+k));
87
88    /* Swap the last n - i elements. */
89    k = 0;
90
91    for(j = i + 1; j < (n + i) / 2 + 1; ++j, ++k) {
92      swap((v+j), (v+(n-k-1)));
93
94    }
95    return 0;
96  }
97  ////////////////////////////////////////////////////////////////////////
    ////////////
98
99
```