一、

程序源码:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

程序名:

版权:

作者: 陶万启

日期: 2023-05-05 21:46

说明:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

int main() {

printf("陶万启\n");

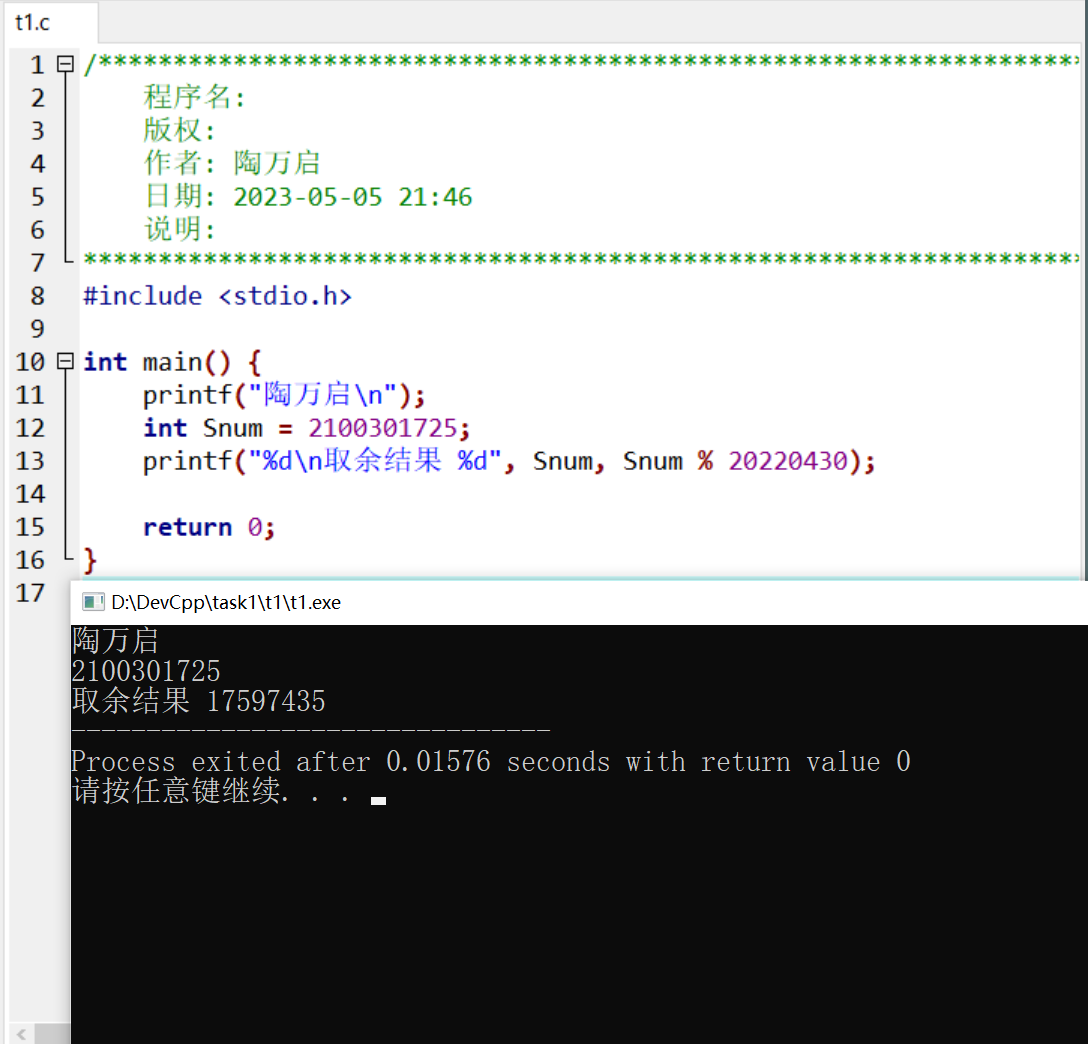
int Snum = 2100301725;

printf("%d\n取余结果 %d", Snum, Snum % 20220430);

return 0;

}

执行结果截图:



二、Select K

程序源码:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

程序名:

版权:

作者: 陶万启

日期: 2023-05-05 16:56

说明:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <stdbool.h>

void copyArray(int arr[], int arr2[], int length);

void generate\_random\_array(int arr[], int n);

void sort(int arr[], int l, int r, int seq);

int partition(int arr[], int l, int r);

void swap(int arr[], int i, int j);

bool isSeqArray(int arr[], int r);

#define N 50 //数组长度

int main() {

int Snum;

printf("请输入你的学号: ");

scanf("%d", &Snum);

int arr[N];

generate\_random\_array(arr, N);

printf("\nRandom array:\n");

for (int i = 0; i < N; i++) {

printf("%d ", arr[i]);

}

int seq = Snum % 100;

printf("\n\n第 %d 小的元素： ", seq);

//为什么是 seq-1？

//数组下标从0开始

sort(arr, 0, N - 1, seq - 1);

printf("\nAfter sort:\n");

for (int i = 0; i < N; i++) {

printf("%d ", arr[i]);

}

return 0;

}

bool isSeqArray(int arr[], int r) {

for (int i = 0; i < r; i++) {

if (arr[i + 1] < arr[i])

return false;

}

return true;

}

void copyArray(int a[], int b[], int length) {

for (int i = 0; i < length; i++)

b[i] = a[i];

}

void generate\_random\_array(int a[], int n) {

srand((unsigned)time(NULL)); // 使用当前时间作为随机数种子

for (int i = 0; i < n; i++) {

a[i] = rand() % 50 + 1; // 生成随机数并赋值给数组元素

}

}

void sort(int arr[], int l, int r, int seq) {

if (l >= r)

return ;

int p = partition(arr, l, r);

if (isSeqArray(arr, seq) && p > seq) {

printf(" %d \n", arr[seq]);

return ;

}

sort(arr, l, p - 1, seq);

if (isSeqArray(arr, seq) && p > seq)

return;

sort(arr, p + 1, r, seq);

}

int partition(int arr[], int l, int r) {

// arr[l+1...j] < v ; arr[j+1...i] >= v

int j = l;

for (int i = l + 1; i <= r; i++)

if ((arr[i] - arr[l]) < 0) {

j++;

swap(arr, i, j);

}

swap(arr, l, j);

return j;

}

void swap(int arr[], int i, int j) {

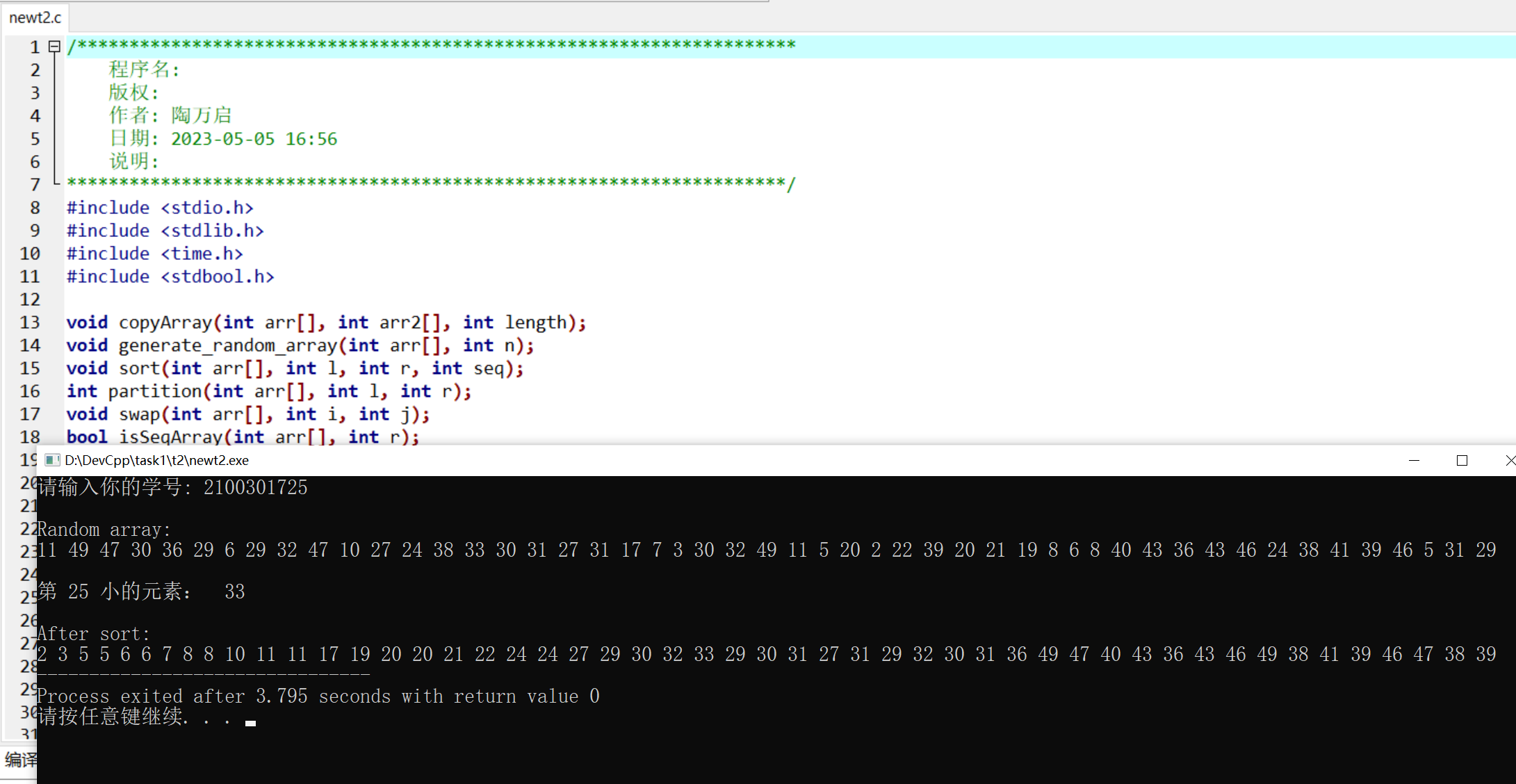
int temp = arr[i];

arr[i] = arr[j];

arr[j] = temp;

}

执行结果截图:



三、

程序源码:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

程序名:

版权:

作者: 陶万启

日期: 2023-05-04 17:43

说明:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

void get(int n);

void print(int arr[]);

void loop(int arr[], int v, int n);

int main() {

int Snum;

printf("请输入你的学号：\n");

scanf("%d", &Snum);

int n = Snum % 100 %9+ 9;

printf("总共有 %d 只队伍\n", n);

get(n);

return 0;

}

void get(int n) {

int i = 0;

int j = 1;

if (n % 2 == 1) { //如果队伍是奇数

n++;

i = 1;

}

int arr[n] ;

//队伍赋值，若队伍为奇数，首位赋值就跳过，且冗余值为0

if (i != 0) {

arr[0] = 0;

}

for (i; i < n; i++) {

arr[i] = j++;

}

for (i = 0; i < n; i++) { //队伍赋值，若队伍为奇数，首位赋值就跳过，且冗余值为0

printf("%d ", arr[i]);

}

printf("\n");

loop(arr, n / 2, n); //循环编排开始

}

void loop(int arr[], int v, int n) {

int length = n;

int fre, end;

int arr2[length];//arr2是下一轮循环所用的新圆圈

arr2[0] = arr[0];//首位不动

for (int i = 0; i < length - 1; i++) {

if (arr[0] == 0) {

fre = 1, end = length - 1;

} else {

fre = 0, end = v;

}

printf("\n第 %d 天的比赛：\n", i + 1);

while (fre < v) {

printf("%dvs%d ", arr[fre], arr[end]);

if (fre == 0)

end = length;

fre++;

end--;

}

printf("\n");

//重新逆时针调整数组元素

fre = 2;

int temp = arr[1]; //保存位置为 1 的数组元素

for (; fre < length; fre++) {

arr2[fre - 1] = arr[fre];

}

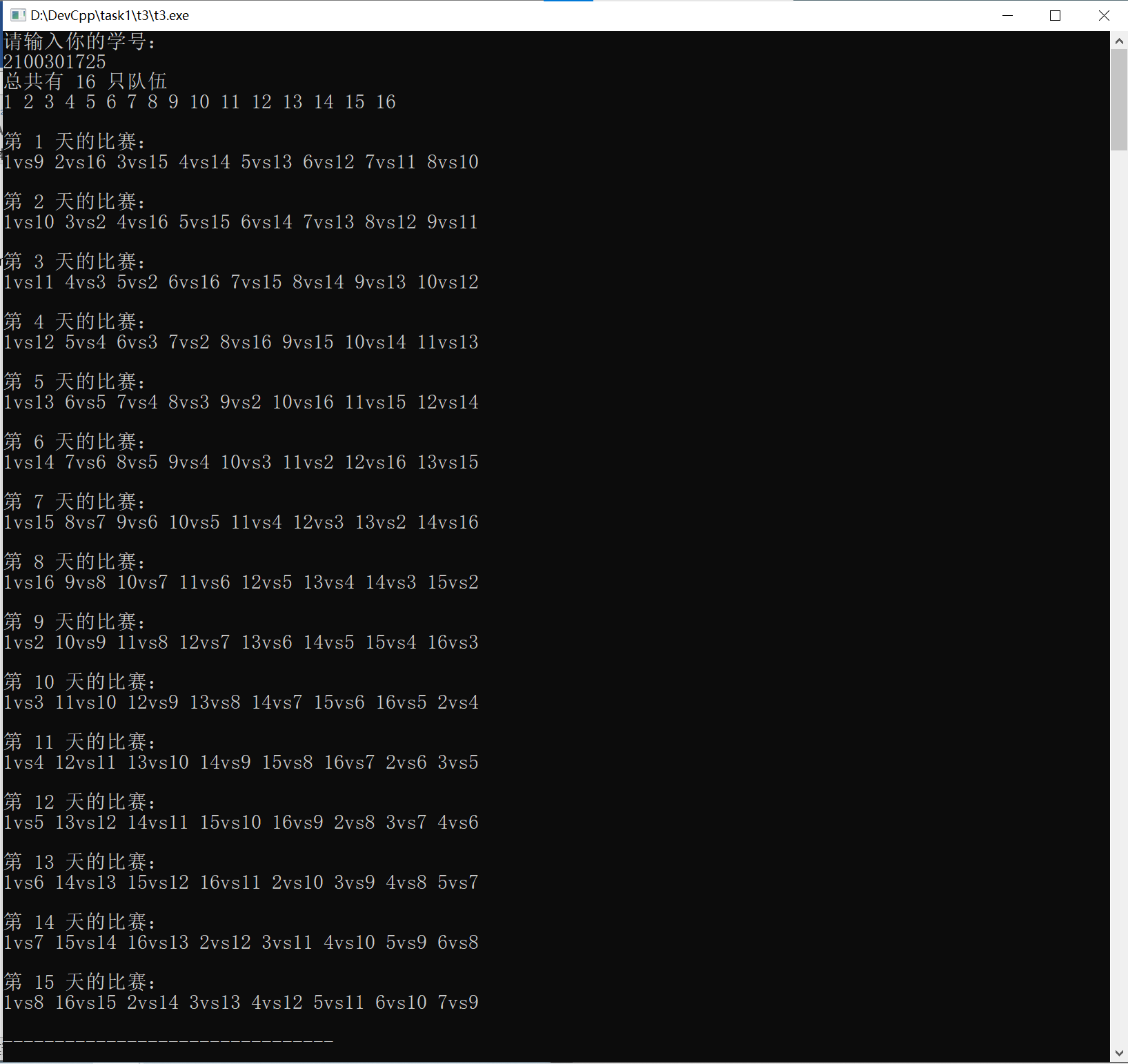
arr2[fre - 1] = temp;

arr = arr2;

}

}

执行结果截图：



四、

程序源码：

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

程序名:

版权:

作者: 陶万启

日期: 2023-05-05 21:05

说明:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <stdbool.h>

void InsertSort(char arr[], int len);

void swap(char \*a, char \*b);

void reverse(char \*str, int start, int end);

int nextPermutation(char \*str, int n);

void lexicographicPermute(char \*str, char \*str2);

bool flag = false;

int sum = 0;

// 交换两个字符

void swap(char \*a, char \*b) {

char temp = \*a;

\*a = \*b;

\*b = temp;

}

// 反转字符串

void reverse(char \*str, int start, int end) {

while (start < end) {

swap(&str[start], &str[end]);

start++;

end--;

}

}

void InsertSort(char arr[], int len) {

for (int i = 0; i < len; i++) {

char temp = arr[i];

int j;

for (j = i; j - 1 >= 0 && (temp - arr[j - 1]) < 0; j--) {

arr[j] = arr[j - 1];

}

arr[j] = temp;

}

}

// 找到下一个字典序排列

int nextPermutation(char \*str, int n) {

int i = n - 2;

while (i >= 0 && str[i] >= str[i + 1]) {

i--;

}

if (i < 0) {

return 0;

}

int j = n - 1;

while (j > i && str[j] <= str[i]) {

j--;

}

swap(&str[i], &str[j]);

reverse(str, i + 1, n - 1);

return 1;

}

// 输出全排列

void lexicographicPermute(char \*str, char \*str2) {

int n = strlen(str);

InsertSort(str, n); // 先排序

printf("\n排序后: \n");

printf("%s\n\n", str);

while (nextPermutation(str, n)) {

if (flag) {

printf("第 %d 号: %s\n", sum, str);

exit(0);

}

if (strcmp(str, str2) == 0) {

flag = true;

}

sum++;

}

}

int main() {

int Snum;

printf("请输入你的学号: ");

scanf("%d", &Snum);

int n = (Snum % 100) % 5 + 5;

printf("请输入 %d 个元素: \n", n);

char arr[n];

scanf("%s", &arr);

printf("\n你输入的元素是：\n");

printf("%s", arr);

printf("\n");

char arr2[n];

strcpy(arr2, arr);

lexicographicPermute(arr, arr2);

return 0;

}

执行结果截图：

