计算机程序设计实验报告 实验九

学号：PB22020514 姓名：郭东昊

一、实验题目

**1.** **p180 4.3.2-指针用于内存操作程序填空**

源代码：

|  |
| --- |
|  |

**2. p186 4.3.3.1取字符串的子串**

源代码：

|  |
| --- |
|  |

执行结果：

|  |
| --- |
| 测试一：  **The substring is: bcde** |

**3．p192 4.3.3.7-比较两个文本文件是否相同**

源代码：

|  |
| --- |
|  |

执行结果：

|  |
| --- |
| 测试一：  **电脑有些问题 程序闪退（所有exe文件都这样）** |

**4. p192 4.3.3.8-统计出现次数最多的单词**

源代码：

|  |
| --- |
| //p192 4.3.3.8-统计出现次数最多的单词  #include <stdio.h>  #include <string.h>  #include <ctype.h>  #include <stdlib.h>  #define MAX\_WORDS 100 // 定义单词最大数量  #define MAX\_WORD\_LENGTH 100 // 定义单词最大长度  // 结构体，用于存储单词及其出现次数  struct WordCounter {  char word[MAX\_WORD\_LENGTH]; // 单词  int count; // 单词出现次数  };  // 定义比较函数，用于按照出现次数从多到少排序  int compare\_by\_count(const void \*a, const void \*b) {  struct WordCounter \*wa = (struct WordCounter \*)a;  struct WordCounter \*wb = (struct WordCounter \*)b;  return wb->count - wa->count;  }  int main() {  char line[MAX\_WORDS \* MAX\_WORD\_LENGTH] = {0}; // 用于存储输入的文本  struct WordCounter words[MAX\_WORDS] = {0}; // 用于存储单词及其出现次数  int num\_words = 0; // 用于存储单词数量  // 打开文件  FILE \*fp = fopen("text.txt", "r");  if (fp == NULL) {  printf("文件打开失败！\n");  return 1;  }  // 读取文本  while (fgets(line, sizeof(line), fp) != NULL) {  // 处理文本，提取单词  char \*p = strtok(line, " ,.!?\n"); // 使用分隔符提取单词  while (p != NULL) {  int found = 0;  // 忽略单词的大小写  for (int i = 0; i < strlen(p); i++) {  p[i] = tolower(p[i]);  }  // 统计单词出现次数  for (int i = 0; i < num\_words; i++) {  if (strcmp(words[i].word, p) == 0) {  // 如果单词已经存在，则将其出现次数加 1  words[i].count++;  found = 1;  break;  }  }  if (!found) {  // 如果单词不存在，则将其添加到单词数组中  strcpy(words[num\_words].word, p);  words[num\_words].count = 1;  num\_words++;  }  // 提取下一个单词  p = strtok(NULL, " ,.!?");  }  }  // 对单词数组进行排序  qsort(words, num\_words, sizeof(struct WordCounter), compare\_by\_count);  // 输出出现次数最多的前十个单词及其出现次数  printf("出现次数最多的单词：\n");  for (int i = 0; i < 10 && i < num\_words; i++) {  printf("%s %d\n", words[i].word, words[i].count);  }  // 关闭文件  fclose(fp);  return 0;  } |

执行结果：

|  |
| --- |
| 测试一：  **China’s outlook on human rights has been continuously enriched and improved in practice, with its own cognitive perspective and ideological connotation based on the actual conditions of the country. China is committed to protecting and promoting human rights in development, and follows the “development-based approach to human rights.” Hunger that once threatened the lives of hundreds of millions of people has made China keenly aware that poverty is the biggest obstacle to the realization of human rights, and that the rights to subsistence and development are the primary basic human rights. Colonial aggression against China following the Opium War has led the nation to fully understand that it is impossible to talk about human rights without sovereignty, and difficult to safeguard certain individuals’ human rights without protecting collective rights.**  **Human rights have historical, specific and practical contexts, and there are no fixed standards or identical models for its development and protection. The path of human rights development in each country should be respected, and the human rights conditions of that country should be judged by its own people. Based on the human rights philosophy that centers on the people, China has proposed that “living a life of contentment is the ultimate human right.”**  **出现次数最多的单词：**  **the 14**  **rights 13**  **human 11**  **and 11**  **that 7**  **of 7**  **to 7**  **development 4**  **has 4**  **on 4** |

**5. p192 4.3.3.9-用函数指针实现生成三角函数表，并存入文件以便查阅**

源代码：

|  |
| --- |
| #include <stdio.h>  #include <math.h>  // Function prototype for the table() function  void table(double (\*f)(double), double first, double last, double incr);  int main() {  // Declare function pointers for the trigonometric functions  double (\*sin\_ptr)(double) = sin;  double (\*cos\_ptr)(double) = cos;  double (\*tan\_ptr)(double) = tan;  // Generate tables for the sin, cos, and tan functions  table(sin, 0, 360, 10);  table(cos, 0, 360, 10);  table(tan, 0, 360, 10);  return 0;  }  // Definition of the table() function  void table(double (\*f)(double), double first, double last, double incr) {  // Open a file for writing the table  FILE \*fp = fopen("trig\_table.txt", "a");  // Print a header for the table  fprintf(fp, "Trigonometric Function Table\n\n");  // Loop over the values of the argument  for (double x = first; x <= last; x += incr) {  // Print the value of the argument and the corresponding value of the function  if (f(x) >= 0)  fprintf(fp, "x = %-3.3f\tf(x) = %-8.3f\n", x, (\*f)(x));  else  fprintf(fp, "x = %-3.3f\tf(x) = %-8.3f\n", x, (\*f)(x));  }  fprintf(fp, "\n");  // Close the file  fclose(fp);  } |

执行结果：

|  |
| --- |
| 测试一： |

**二、实验的时候遇到的问题（bug、助教验收时问的问题）**