

**实 验 报 告**

**（ 2019 / 2020 学年 第 一 学期）**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程名称 | 高级语言程序设计A（B0300311S） | | | | | |
| 实验名称 | 实验一： 流程控制实验 | | | | | |
| 实验时间 | 2019 | 年 | 10 | 月 | 24 | 日 |
| 指导单位 | 计算机学院、软件学院、网络空间安全学院 | | | | | |
| 指导教师 | 朱立华 | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 学生姓名 | 蒋怀萱 | 班级学号 | B19011607 |
| 学院(系) | 通信与信息工程学院 | 专 业 | 通信工程 |

| **实 验 报 告** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **实验名称** | **实验一 流程控制实验** | | | **指导教师** | 朱立华 |
| **实验类型** | **验证** | **实验学时** | **2** | **实验时间** | 2019.10.24 |
| 1. **实验目的和要求**   （1）掌握Visual Studio 2010集成开发环境的基本使用方法，体会开发程序的完整过程：编辑、编译、链接、运行。  （2）理解C语言程序的流程控制方式，较好地运用if语句、循环语句解决分支、循环结构的问题。  （3）使用调试器进行单步跟踪，观察每一步变量的变化情况，并学会设置断点，或者跟踪到光标处等方式，观察分支及循环结构程序执行的流程走向，以及相关变量的变化情况。 | | | | | |
| 二、**实验环境(实验设备)**  硬件: 微型计算机  软件: Windows 操作系统、Microsoft Visual Studio 2010 | | | | | |
| **三、实验原理及内容**  **实验题目（1）【见实验教材实验一的题目１】：**在D盘上建立以自己学号命名的文件夹（例如：B07020518）（以后实验的所有文件存于此文件夹下）；正确进入VS2010集成开发环境，建立一个单文件项目，项目名为：exp1\_1，然后向该项目中以新建方式添加一个源文件，名为：exp1\_1.c，完成：从键盘输入两个整数a和b，计算并输出他们的和sum、差diference、积product、商quotient和平均值ave（注意ave和商的定义类型）。  具体要求：  ① 编辑、编译、连接并运行程序，观察运行结果。  ② 观察exp1\_1.obj、exp1\_1.exe、exp1\_1.c所在的文件夹。  ③ 对变量quotient分别定义为int和double型，在printf函数中正确使用格式串，当读入的两个整数为4和5时，观察运行结果并分析原因，思考怎么使变量quotient更准确。  ④ 对变量ave分别定义为int和double型，在printf函数中正确使用格式串，当读入的两个整数为4和5时，观察不同的运行结果并分析原因。  ⑤ 单步跟踪程序（用F10键），观察每一步程序运行时各个变量的变化情况。  **实验解答：**   1. 源程序代码如下： | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| #include<stdio.h>  #include<math.h>  int main( )  {  int a,b;  int sum,difference,product;  double quotient,ave;  printf("Please input a and b:");  scanf("%d%d",&a,&b);  sum=a+b;  printf("sum=%d\n",sum);  difference=a-b;  printf("difference=%d\n",difference);  product=a\*b;  printf("product=%d\n",product);  quotient=(double)a/b;  printf("quotient=%lf\n",quotient);  ave=(double)(a+b)/2;  printf("ave=%lf\n",ave);  return 0;  }  ②   |  |  | | --- | --- | | **文件名** | **文件所在的完整路径(例如：d:\ttt\t1.c)** | | exp1\_1.c | D:\B19011607\exp1\_1\exp1\_1.c\exp1\_1.cpp | | exp1\_1.obj | D:\B19011607\exp1\_1\exp1\_1.c\Debug\exp1\_1.obj | | exp1\_1.exe | D:\B19011607\exp1\_1\exp1\_1.c\Debug\exp1\_1.c.exe |   ③   |  |  |  |  | | --- | --- | --- | --- | | quotient的类型 | 求解quotient的语句 | 输出quotient的格式串 | 屏幕输出的quotient值 | | int | Quotient=a/b | Printf(“%d”,quotient) | 0 | | double | Quotient=(double)a/b | Printf(“%lf”,quotient) | 0.8 |   **运行结果的差异、原因以及保证结果正确的措施：**  **差异：一个为0，一个为正确答案。**  **原因：int为整型变量，除法整数除结果仍为整数，保留结果的整数部分，double实数除，（double）试被除数是double类型，结果浮点类型，所以更加精确。**  **措施：把quotie定义为double变量，且由于一开始的a、b是int类型，写表达式时要在一个数前加上（double）。**  ④   |  |  |  |  | | --- | --- | --- | --- | | ave的类型 | 求解ave的语句 | 输出ave的格式串 | 屏幕输出的ave值 | | int | Ave=(a+b)/2 | Printf(“%d”,ave) | 4 | | double | ave=(double)(a+b)/2 | printf("%lf\n",ave) | 4.5 | |
| **运行结果的差异及原因：**  **差异：一个为4 答案错误，一个为4.5，答案正确。**  **原因：int定义为整型变量，除法保留整数部分，故保留4，而定义ave为double变量且（double）（a+b）让a+b也成为了double型，因此结果正确。**  **实验题目（2）【见实验教材实验二的题目6】：**编程序exp2\_6.c，读入3个double型变量a,b,c的值，用if语句判断它们能否构成三角形，如果能构成三角形又是构成的何种三角形（等边三角形、等腰三角形、直角三角形、等腰直角三角形、一般三角形）？输出各种情况下判断的结果。（建议参考实验指导教材的流程图4.20，先对a,b,c进行由小到大的排序可以简化代码）  **实验解答：**   1. 源程序exp2\_6.c的代码是：   #include<stdio.h>  #include<math.h>  int main()  {  double a,b,c;  scanf("%lf%lf%lf",&a,&b,&c);  if(a<=0||b<=0||c<=0)  {  printf("error input!!!\n");  }  else  {  if(a+b>c&&a+c>b&&b+c>a)  {  if((a==b&&fabs(a\*a+b\*b-c\*c)<1E-3)||(b==c&&fabs(b\*b+c\*c-a\*a)<1E-3)||(a==c&&fabs(a\*a+c\*c-b\*b)<1E-3))  {  printf("等腰直角三角形\n");  }  else if((a==b&&b!=c)||(b==c&&c!=a)||(a==c&&c!=b))  {  printf("等腰三角形\n");  }  else if(a==b&&b==c)  {  printf("等边三角形");  }  else if(fabs(a\*a+b\*b-c\*c)<1E-3||fabs(a\*a+c\*c-b\*b)<1E-3||fabs(b\*b+c\*c-a\*a)<1E-3)  {  printf("直角三角形\n");  }  else  {  printf("一般三角形\n");  }  }  else  {  printf("不能构成三角形\n");  }  }  return 0;  } |
| ②多次运行程序，输入不同的数据，得到不同结果，请填写下表   |  |  |  | | --- | --- | --- | | **三个原始数据代表的情况** | **你所输入的三条边长** | **你程序的输出结果** | | 能构成等边三角形 | 3 3 3 | 等边三角形 | | 能构成直角三角形 | 3 4 5 | 直角三角形 | | 能构成等腰直角三角形 | 1 1 1.414 | 等腰直角三角形 | | 能构成等腰三角形 | 8 8 9 | 等腰三角形 | | 能构成一般三角形 | 2.7 3.3 4.5 | 一般三角形 | | 不能构成三角形 | 9 9 100 | 不能构成三角形 |   **实验题目（3）【见实验教材实验三的题目1】：** 利用while、for、do…while循环语句计算求和问题：    **实验解答：**   1. 求前1000项之和的源程序代码如下：   #include<stdio.h>  #include<math.h>  int main()  {  double i;  double x;  scanf\_s("%lf",&x);  double item,sum;  sum=0;    double m,n;  m=2,n=1;  i=1;    do  {  item=pow(-1,i+1)\*(double)m/n;  sum+=item;  m+=n;  n=m-n;  i+=1;    }  while(i<=x);  printf("%lf",sum);  return 0;  } |
| 1. 如果需要计算的项数并不是常量1000而是变量n（通过输入获得值），请在程序exp3\_1.c的基础上做修改，修改后的源程序名为：exp3\_1A.c，运行修改后的程序。请在下面表格中写出你所改动（包括增加、删除、修改）的语句（表第一行为示例）  |  |  |  | | --- | --- | --- | | **原来在exp3\_1.c中** | **操作说明** | **对应修改后在exp3\_1A.c中** | | int i,sign; | 定义两个整型变量i和sign | int i,sign,x; | | 无 | 输入项数x | scanf("%d",&x); | | while(i<=1000); | 有1000项时限定条件 | while(i<=x); | |  |  |  |  1. 运行以上程序，填写下表，观察运行结果  |  |  | | --- | --- | | **键盘输入的n值** | **屏幕输出的求和结果** | | 2 | 0.500000 | | 8 | 0.577683 | | 13 | 2.195954 | | 14 | 0.577921 | | 15 | 2.195956 | | 16 | 0.577922 | | 19 | 2.195956 | | 20 | 0.577922 | | 199 | 2.195956 | | 200 | 0.577922 |   **请自行任意输入超过16的任意奇数、偶数，观察求和结果，结合以上表格中的结果，你发现了什么规律？请观察该求和每项的特点，查阅资料，解释你所发现的规律**。  奇数为2.195956，偶数为0.577922  原因：费博纳茨数列特性。  **实验题目（4）【见实验教材实验三的题目4】：**编写程序exp3\_4.c，先用一层循环判断从键盘读入的正整数x是否为质数，并输出判断结果。在此基础上改写，再加上外层循环，找出规定范围（a,b）内的所有质数，并按每行5个的格式输出。（这里a，b变量从键盘输入并且范围为：10≤a≤b≤1000）  **实验解答：**   1. 写程序exp3\_4.c实现从键盘上读入一个正整数x，判断是否为质数并输出判断结果，代码如下： |
| #include<stdio.h>  #include<math.h>  int main()  {  int n,i,k;  do  {  scanf("%d",&n);  }  while(n<=0);  if(n==1)  {  printf("%d is not a prime");  }  else  {  k=(int)sqrt(n\*1.0);  for(i=2;i<=k;i++)  {  if(n%i==0)  {  break;  }  }  if(i>k)  {  printf("%d is a prime.\n",n);  }  else  {  printf("%d is not a prime.\n",n);  }  }  return 0;  }    ② 在程序exp3\_4.c的基础上改写，加上外层循环，找出规定范围（a,b）内的所有质数，并按每行5个的格式输出，改写后的程序exp3\_4A.c代码如下：  （**提示：**用于判断是否质数的变量x不再是读入的，而是作为循环控制变量，依次取规定范围(a,b)内的所有数，变量a,b的值从键盘输入，保证读入的a,b在[10，1000]范围内，并且a是下限b是上限，也就是说，应该符合的条件是：10<=a<=b<=1000，否则输入数据无效，提示用户重新输入a,b直到符合要求为止） |
| #include <stdio.h>  #include <math.h>  int main()  {  int i,j,m,n,k,s,a,count=0;  do  {  printf("请输入在10到1000之间的两个整数\n");  printf("第一个数要小于第二个数：");  scanf("%d",&n);  scanf("%d",&m);  }while(n<10||n>1000||m<10||m>1000||n>=m);  for(j=n;j<m;j++)  {  k=sqrt(j\*1.0);  a=1;  for(i=2;i<=k;i++)    if(j%i==0)  {  a=0;  break;  }  if(a==1)  {  printf("%7d",j);  count++;  if(count%5==0)  {  printf("\n");  }  }  }  }  运行程序，按表格第一列分别输入a,b的值，填入对应的输出结果   |  |  | | --- | --- | | **键盘输入的a,b值** | **屏幕输出的结果** | | 12 14 | 13 | | 24 28 | 无 | | 34 56 | 37,41,43,47,57 | | 111 189 | 113 127 131 137 139  149 151 157 163 167  173 179 181 | |  |  | |
| **四、实验小结**（包括问题和解决方法、心得体会、意见与建议、实验出错信息及解决方案等）  (一)实验中遇到的主要问题及解决方法  主要问题：   1. 在判断三角形时，if语句的先后关系会影响输出结果的正确性。 2. 调用函数如sqrt时，里面的数据必须是浮点型，而初始定义为整型。 3. 最后一题语句的嵌套，按行输出5个，以及是否输出怎么输出。 4. 在费博纳茨第二小问卡住了。   解决方法：   1. 进行多次调试，调换if语句的位置。 2. 查阅书本资料进行了解学习。 3. 查阅书本及网络资料进行学习思考。 4. Debug没有发现问题，最后把所有变量定义为double类型成功。   （二）实验心得  C语言对逻辑的需求非常高，且有时候拼命想不出来，这时候如果静心思考，虚心请教，再把代码敲出来，非常有成就感。  C语言的应用广泛，以后要重视C语言的学习。  （三）意见与建议（没有可省略） |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. **支撑毕业要求指标点** | | | | | | |
| **六、指导教师评语** | | | | | | |
| **评 分 细 则** | **评分项** | **优秀** | **良好** | **中等** | **合格** | **不合格** |
| **遵守实验室规章制度** |  |  |  |  |  |
| **学习态度** |  |  |  |  |  |
| **算法思想准备情况** |  |  |  |  |  |
| **程序设计能力** |  |  |  |  |  |
| **解决问题能力** |  |  |  |  |  |
| **课题功能实现情况** |  |  |  |  |  |
| **算法设计合理性** |  |  |  |  |  |
| **算法效能评价** |  |  |  |  |  |
| **回答问题准确度** |  |  |  |  |  |
| **报告书写认真程度** |  |  |  |  |  |
| **内容详实程度** |  |  |  |  |  |
| **文字表达熟练程度** |  |  |  |  |  |
| **其它评价意见** |  | | | | |
| **本次实验能力达成评价（总成绩）** |  | **批阅人** |  | **日期** |  |