

计算机程序设计基础大作业（2020 秋）

1. 作业要求

- 以下三道题目（矩阵运算、计算器和连连看游戏）可任选一题。
- 编程语言要求用 C 语言，代码量不作硬性要求，要求有适当的注释。
- 每个人单独一组，自主完成，如果发现抄袭或他人代写，记 0 分。
- 请注意程序的友好性，实现的功能要符合思维习惯。

2. 提交说明

大作业必须包括以下部分：

- 1) 实验报告：描述程序的整体框架、各个功能模块的设计和实现思路，以及最终的实现效果。
- 2) readme.txt 陈述简单的操作说明及注意事项。
- 3) 可执行的.exe 文件（即工程中 debug 或 release 文件夹下的 exe 文件），放置在根目录下。使用苹果电脑的同学，需在 Windows 下重新编译生成可执行文件。
- 4) 整个工程，包含所有的源程序及相关文件，可以重新编译。

所有文件统一压缩为 zip 格式的压缩包，并将其命名为“学号_姓名.zip”，例如：

2020010000_张三.zip

3. 分数与权重

每道题目的总分均为 100 分。

考虑到题目有难易差别，为公平起见，各题给出了难度系数，作为成绩计算的权重值。例如，如果选择二颗星的题目，若程序完全正确，则最终成绩为 $100 \times 0.75 = 75$ 分。

难度系数	权重值
★★★★★	1.0
★★★	0.85
★★	0.75

4. 选题之一：矩阵运算计算器（难度系数：★★★）

4.1 问题背景

在数学中，矩阵（Matrix）是一个按照长方阵列排列的复数或实数集合。矩阵是高等代数学中的常见工具，也常见于统计分析等应用数学学科中。由 $m \times n$ 个数 a_{ij} 排成的 m 行 n 列的数表称为 m 行 n 列的矩阵，简称 $m \times n$ 矩阵。记作：

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ a_{31} & a_{32} & \cdots & a_{3n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

这 $m \times n$ 个数称为矩阵 A 的元素，简称为元，数 a_{ij} 位于矩阵 A 的第 i 行第 j 列，称为矩阵 A 的 (i, j) 元，以数 a_{ij} 为 (i, j) 元的矩阵可记为 (a_{ij}) 或 $(a_{ij})_{m \times n}$ ， $m \times n$ 矩阵 A 也记作 A_{mn} 。

元素是实数的矩阵称为实矩阵。矩阵运算在科学计算中非常重要，而矩阵的基本运算包括矩阵的加法，减法，转置和乘法等。

另外，在编程时，我们还可以把矩阵整体右移一列，即把它的每一列元素都向右移动一列，而最右边那列元素，就被移动到了最左边。例如，对于矩阵

$$A = \begin{pmatrix} 0 & -2 & 1 \\ 3 & 0 & -2 \\ -2 & 3 & 0 \end{pmatrix}$$

如果把 A 整体右移一列，将会得到：

$$A\# = \begin{pmatrix} 1 & 0 & -2 \\ -2 & 3 & 0 \\ 0 & -2 & 3 \end{pmatrix}$$

4.2 基本要求

设计并实现一个程序，要求具有求矩阵的加法、减法、转置、乘法以及右移运算等功能。用户可以根据提示输入相应的矩阵，然后再进行相应的操作。本题目要求的矩阵是整数矩阵，**行数和列数最大均不超过 10**。

输入输出要求如下：

- 1) 矩阵输入。首先输入行数 m ，然后输入列数 n ，接着依次输入每行的数据，每个元素都是整数。如下所示：

2 3
1 2 3
4 5 6

- 2) 矩阵输出。按行依次输出矩阵的元素，各元素以空格隔开。

要求实现的功能如下：

- 1) 加法。输入矩阵 A_{mn} 、 B_{pq} ，输出和矩阵 C 。矩阵 A_{mn} 和 B_{pq} 能够相加的条件是： $m = p, n = q$ ；矩阵 A_{mn} 和 B_{pq} 如果不能相加，请给出提示信息；若能够相加，则求和矩阵 C 并输出 C ；
- 2) 减法。输入矩阵 A_{mn} 、 B_{pq} ，输出差矩阵 C 。矩阵 A_{mn} 和 B_{pq} 能够相减的条件是： $m = p, n = q$ ；矩阵 A_{mn} 和 B_{pq} 如果不能相减，请给出提示信息；若能够相减，则求差矩阵 C 并输出 C ；
- 3) 转置。将矩阵的行列互换得到的新矩阵称为转置矩阵，矩阵的转置 $A' = (a_{ji})_{n \times m}$ 。输入矩阵 A_{mn} ，输出转置矩阵 A' ；
- 4) 矩阵乘法。输入矩阵 A_{mn} 、 B_{nq} ，输出积矩阵 C 。如果矩阵 A 和 B 不能相乘，请给出提示信息；若能够相乘，则求积矩阵 C 并输出 C ；其中，

$$C_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj} = \sum_{r=1}^n a_{ir}b_{rj}$$

- 5) 右移运算。输入矩阵 A ，把它整体向右移动一列，然后输出新得到的矩阵。

4.3 评分标准

1、功能性要求（70 分）

- 评分方法：对于每一个基本要求，有若干个测试点，设该功能共 a 分，共 N 个测试点，通过一个测试点可以得到 $\frac{a}{N}$ 分。通过一个测试点的含义是对于这个测试点的表达式，在 10 秒内输出正确的计算结果。
- 各个功能的分数分配如下：（70 分）
 - a) 加法计算（10 分）
 - b) 减法计算（10 分）
 - c) 转置计算（10 分）
 - d) 乘法计算（20 分）
 - e) 右移运算（20 分）

2、非功能性要求（10 分）

- 程序健壮性：你的程序需要处理异常的输入，并对异常的输入输出提示信息，尽量不要使程序崩溃。助教会使用一些常见的异常输入测试你的程序（譬如，用户输入的可能不是数字，用户输入的行数或列数可能大于 10，等等），如果程序崩溃或没

有给出相应提示信息，则每个未处理的异常输入扣 2 分，扣完为止。（7 分）；

- 提交正确性：提交的内容要完整。提交的二进制可以直接在 Windows 系统上运行，或者其对应源文件可以在 Visual Studio，32 位 Debug 模式下正确编译。（3 分）

3、文字材料（20 分）

- 实验报告，包括但不限于以下几点。（15 分）
 - a) 功能的代码位置(文件名或起始行号、函数名等)。
 - b) 描述各个功能模块的设计和实现思路。
 - c) 其他任何你想让助教知道的内容。
 - d) 编写用户使用手册，给出详细的功能说明，如各步骤运行结果截图。
- 源代码（5 分）
 - a) 数据存取和处理方式合理，代码结构清晰，模块独立。（2 分）
 - b) 代码具有良好的编码规范，文件与目录结构清晰，排版风格统一，可读性强，注重命名规范、编码格式和注释等。（3 分）

5. 选题之二：计算器（难度系数：★★★）

5.1 基本要求

设计并实现一个计算器，用户可以根据提示输入相应的表达式，然后程序输出表达式的计算结果。

输入输出要求：输入一行数学表达式，表达式包含数字和运算符。输出一行数字，表示数学表达式的结果。

【必选部分】

- 1) 数字只包含正整数，运算符只包含+，-，运算符和数字之间直接连接，没有空格，表达式中不会有超过 10000 个运算符，表达式的数字和计算结果保证在 int 范围内，输出要求必须是整数，并且和正确结果完全一致。

例：

输入：1+2-3+3 输出：3

【限选部分】

- 1) 数字只包含整数，运算符只包含+，-，*，%（求余数）和括号。运算符与数字之间可能有空格，但数字内部保证没有空格，表达式中不会有超过 10000 个运算符，表达式的

数字保证在 int 范围内，计算结果保证在 long long 范围内。注意取模运算的符号和 C99 标准一致。输出要求必须是整数，并且要和正确结果完全一致。

例：

输入： $(1 + (-2)) * 5 \% 2 - 4$ 输出： -5

- 2) 数字包含正整数和小数，运算符包含+, -, *, /, ^ (乘方) 和绝对值(| · |)，运算符与数字之间可能有空格，但数字内部保证没有空格，表达式中不会有超过 10000 个运算符。输入保证在正确的使用下，double 类型可以保证结果的精度在要求范围内。输出一个数字，要求和正确结果的绝对误差在 0.0001 以内。

例：

输入

$|-1.0 * 3| - 4 / 3 + 0.2^{0.2}$

输出

2.39144633034436219813728066754824 ± 0.0001 内都是正确的

【任选部分】

注意这一部分对程序运行时间有要求，对于符合题目描述的输入，程序要在 4 秒内得到结果。

- 1) 数字包含整数或小数，运算符只包含+, -, *, 和括号。运算符与数字之间可能有空格，但数字内部保证没有空格，表达式中不会有超过 30 个运算符，表达式中的数字不会超过 500 位，输出要求必须和正确结果完全一致，不允许有前导 0，也不允许小数点后有后置 0。

例：

输入

$(123123124132452351345123413521341324-$

$3413241340324123784571932874)*21332984732842331243241324234.234$

输出：

2626583654562831808544433473472064917404583073752617482123538877.3

注意，形如：

2626583654562831808544433473472064917404583073752617482123538877.300

或

002626583654562831808544433473472064917404583073752617482123538877.3

的答案是不合法的。

- 2) 数字只包含整数, 运算符只包含*, 运算符与数字之间可能有空格, 但数字内部保证没有空格, 表达式中只有两个数字和一个*号, 其中数字的位数不会超过 100000 位。

输出要求必须是整数, 并且要和正确结果完全一致。

例:

输入:

```
123123124132452351345123413521341324*3413241340324123784571932874
```

输出:

```
420248937238745134153510177991568300037542274470401219770285176
```

注意:

- 1) 你的程序需要处理异常的输入, 并对异常的输入输出提示信息, 尽量不要使程序崩溃。一些典型的异常输入包括:
3/ (3 + 2 - 5) (除 0)。
(12+3)-23)*3(括号不匹配)
(23+34-)(运算符缺少操作数)
- 2) 本次作业不允许使用标准库之外的库。
- 3) 对于你选择的每一项要求, 你可以每个要求单独编译一份可执行程序, 也可以整合到一个可执行程序中。

5.2 评分标准

1、功能性要求 (70 分)

- 评分方法: 对于每一个基本要求, 有若干个测试点, 设该功能共a分, 共N个测试点, 通过一个测试点可以得到 $\frac{a}{N}$ 分。通过一个测试点的含义是对于这个测试点的表达式, 在规定的时间内 (必选和限选为 10 秒, 任选为 4 秒) 输出正确的计算结果。程序在 Visual Studio, 32 位 debug 模式下运行。
- 各个功能的分数分配如下: (70 分)
 - a) 必选功能 (30 分)
 - b) 限选功能: 实现一个功能 30 分, 如果多做, 则按照得分高的给分。
 - c) 任选功能: 实现一个功能 10 分, 如果多做, 则取分数相加, 10 分封顶。形式化地, 假设你五个功能点的得分分别为a,b,c,d,e, 则最终的得分为 $a + \max\{b, c\} + \min\{10, d + e\}$

2、非功能性要求（10 分）

- 程序健壮性：你的程序需要处理异常的输入，并对异常的输入输出提示信息，尽量不要使程序崩溃。助教会使用一些常见的异常输入测试你的程序（譬如，用户输入的可能不是数字，用户输入的行数或列数可能大于 10，等等），如果程序崩溃或没有给出相应提示信息，则每个未处理的异常输入扣 2 分，扣完为止。（7 分）；
- 提交正确性：提交的内容要完整。提交的二进制可以直接在 Windows 系统上运行，或者其对应源文件可以在 Visual Studio，32 位 Debug 模式下正确编译。（3 分）

3、文字材料（20 分）

- 实验报告，包括但不限于以下几点。（15 分）
 - a) 所实现功能在源文件中的位置（文件名或起始行号、函数名等）。如果实现了某个功能，必须在文档中声明，否则将无法得到这部分分数。如果声明实现了某项功能，但没有给出对应的代码位置和实现思路，或没有描述使用的算法，缺少一项将会被扣除 2 分，封顶 10 分，扣完为止。
 - b) 描述各个功能模块的设计和实现思路。
 - c) 其他任何你想让助教知道的内容。
 - d) 编写用户使用手册，给出详细的功能说明，如各步骤运行结果截图。
- 源代码（5 分）
 - a) 数据存取和处理方式合理，代码结构清晰，模块独立。（2 分）
 - b) 代码具有良好的编码规范，文件与目录结构清晰，排版风格统一，可读性强，注重命名规范、编码格式和注释等。（3 分）

6. 选题之三：连连看（难度系数：★★★★★）

6.1 规则简介

连连看是一款经典的 PC 端益智小游戏，游戏玩法非常简单，选择一对相同的图案连线，如果连线路径可避开其他图案且不超过三条直线，即可消除此对图案，获得积分。

下面对本题的游戏规则进行介绍：

- 1) 启动游戏后进入菜单界面，包括“开始游戏”、“游戏设置”、“查看排行榜”、“退出游戏”等功能；
- 2) 点击“开始游戏”后进入游戏界面，生成 10 行×14 列随机排列的图案，图案共有 13 种，图案自定；每种图案的数量须是偶数，每局游戏限时 6 分钟以内，进入游戏

后显示倒计时；

- 3) 鼠标左键单击可选择图案，再次单击可取消选中；选择一张图案后，单击其他图案，若连接线路径符合要求且图案相同则消除该图案对，否则选中图案切换到第二个；
- 4) 单击“洗牌”按钮重新洗牌，每局游戏仅允许玩家主动洗牌 1 次；
- 5) 单机“提示”按钮游戏自动显示一组可以消除的图案对，每局游戏仅可提示 2 次；
- 6) 分数统计：游戏界面实时显示游戏分数，每消除一对图案+2 分，每次 combo（消除时间间隔<2 秒）得分翻倍，分高者进入排行榜，每次启动游戏时可查看排行榜；
- 7) 单击“退出”按钮放弃本局游戏，回到菜单界面；
- 8) 在规定时间内消除全部图案，判定为赢，游戏结束，返回菜单界面；规定时间结束仍有图案未消除或单击退出，判定为输。
- 9) 当出现残局（场上无可连接的图案）时，自动洗牌；

6.2 功能要求

1、基本功能

- 1) 图形用户界面，限定使用 **easyx** 图形库来实现 (<https://easyx.cn/>)，实现效果可参考下图，按美观度打分：



- 2) 图案随机初始化，初始化图案中至少有 10 对相同图案相邻；
- 3) 采取合适的数据结构，高效检测图案连线是否满足条件；
- 4) 统计分数、时间等实时显示；

- 5) 正确实现基本操作，对于非法操作有相应处理；
- 6) 正确存储/读取排行榜数据；
- 7) 正确判定输赢；
- 8) 良好的界面和交互方式。

2、扩展功能

- 1) 用户可自行设置游戏图案的行列个数；
- 2) 游戏背景音乐和音效；
- 3) 难度递增的闯关功能；
- 4) 每次消除后图案位置变化（向上、向下、左右集中等）；

6.3 评分标准

1、功能性要求（60 分）

- 图形用户界面（20 分）：完整实现作业基本要求中的所有功能和相应的图形用户界面，每缺少一个界面扣 3 分，扣完为止；
- 游戏功能（40 分）：
 - a) 基本功能（30 分）：完整实现作业基本要求的功能，程序能运行正常，每缺少一个功能扣 5 分，扣完为止；
 - b) 扩展功能（10 分）：对于 4 个扩展功能，可以任选其中的一个完成，给 10 分。
 - c) 自定义功能（10 分）：如果实现了一些其他的有特色的功能，可以酌情加分，最多不超过 10 分。此部分分数为额外的附加分，不包含在总分 100 以内。

2、非功能性要求（23 分）

- 可用性：用户交互界面清楚、美观，软件在使用过程中有友好的用户体验，即站在用户的角度，当他在使用某个功能时，应该有足够的帮助信息。例如，在某些情形下，某些暂时不能使用的功能按钮应该被禁止，或者在做出某些动作之前有明确的用户提示信息等（5 分）；
- 性能：游戏过程中操作流畅，有良好的响应速度（4 分）；
- 可靠性：程序具有良好的健壮性和可靠性。例如，当用户操作错误时有良好的提示，而不是发生异常而导致程序崩溃。再如，如果连续运行多个功能，程序也不会出现异常（4 分）；
- 测试：总分 10 分，在程序运行时，每发现一处 bug 扣 2 分，扣完为止（10 分）。

3、文字材料（17 分）

- 实验报告，包括但不限于以下几点。（12 分）
 - a) 描述程序的整体架构，整个工程的实现思路，不同模块之间相互调用的关系，最好能画一个模块之间的关系调用图。（3 分）
 - b) 描述各个功能模块的设计和实现思路（3 分）。
 - c) 程序中有哪些特别的设计思路，由于这一设计给整个程序带来了哪些好处？例如，某种设计优化了代码结构、某算法加快了程序的执行速度等。换言之，你的程序中有哪一些突出的亮点。（4 分）
 - d) 编写用户使用手册，给出详细的功能说明，如各步骤运行结果截图。（2 分）
- 在 `readme` 给出简单的按键说明或操作方式。（1 分）
- 源代码（4 分）
 - a) 数据存取和处理方式合理，代码结构清晰，模块独立。（1 分）
 - b) 代码具有良好的编码规范，文件与目录结构清晰，排版风格统一，可读性强，注重命名规范、编码格式和注释等。（3 分）