

作业 2

练习部分（有批改）

Q1. 写一个函数`tri_area`，接受输入参数`b`和`h`作为三角形的底和高，返回三角形面积，输出结果为`area`

```
function area = tri_area(b,h)
```

Q2. 写一个函数`corners`，接受输入参数`M`作为一个二维矩阵，返回矩阵四个角的值，分别保存为`a,b,c,d`（左上、右上、左下、右下）。

```
function [a,b,c,d] = corners(M)
```

Q3. 写一个函数`taxi_fare`，计算出租车车费。已知起步价1km为10元。每超出1公里（不足1公里按1公里计算）加收3元，如果出现堵车等待时间每分钟（不足1分钟按1分钟计算）0.5元。函数有两个输入参数`km`，`wait`分别表示里程数和等待时长。返回值为车费`fare`。

```
function fare = taxi_fare(km,wait)
```

Q4. 写一个函数`minimax`接受一个输入参数`A`，`A`是一个二维矩阵。返回`A`中每一行最大值和最小值的差，记作`x`，以及`A`中最大值和最小值的差，记作`y`。

```
function [x,y] = minimax(A)
```

Q5. 写一个函数`trio`，接受输入参数`a`和`b`，返回`M`矩阵。`M`矩阵为`3a*b`大小，上三分之一全为1，中三分之一全为2，下三分之一全为3。

```
function M = trio(a,b)
```

Q6. 写一个函数`picker`，接受输入参数`cond`，`a`和`b`。`cond`表示判决条件，如果`cond`为真，输出参数`out`的值等于`a`，否则等于`b`。

```
function out = picker(cond,a,b)
```

Q7. 写一个函数`eligible`，接受输入参数`v`和`q`，分别代表2门课程的期末成绩，如果两门课程均分91以上，且每门成绩不低于89分则通过，返回值`admit`为真，否则为假。

```
function admit = eligible(v,q)
```

Q8. 写一个函数`under_age`，接受一个或两个输入参数。如果是两个输入参数，第一个表示某人的年龄，第二个表示某酒吧限定最低进入年龄，返回值`ty`为真如果某人年龄超过最低限。如果是一个输入参数，则默认最低进入年龄限制是21岁。（提示是用`varargin`）

```
function ty = under_age(varargin)
```

Q9. 写一个函数`valid_date`，输入参数`y,m,d`分别表示年，月，日。使用`if`判断给定的`ymd`可以组成一个合法的日期。返回值`v`表示真或假。例如2022,11,31为假，2022, 2, 29为假。

```
function v = valid_date(y,m,d)
```

Q10. 写一个函数`halfsum`，接受输入参数`A`是一个2d矩阵，返回矩阵右上角三角形的所有元素之和`summa`。例如以下矩阵返回值为29。

`A =`

```
8     1     6
3     5     7
4     9     2
```

```
function summa = halfsum(A)
```

Q11. 写一个函数`next_prime`，接受输入参数`n`为整数，返回第一个比`n`大的素数。（使用`isprime()`判断是否为素数）

```
function k = next_prime(n)
```

Q12. 写一个函数`freezing`，接受输入参数`A`表示某地区一段时间的气温，返回`A`中所有温度小于32度的天数总数`n`。

```
function n = freezing(A)
```

Q13. 写一个函数`max_sum`，接受输入参数向量`v`和正整数`n`。返回`v`中连续`n`个元素和为最大值的情况，返回值`s`和`idx`分别表示最大可能的和以及第一个元素开始的索引。如果`v`的长度小于`n`，`s`和`idx`分别为0和-1。

```
function [s,idx] = max_sum(A,n)
```

例如`v = [8 3 4 1 5 9 6 7 2]`, `n = 3`

```
[s,idx] = max_sum(v,n)
```

% 得到结果 $s = 22$, $idx = 6$

作业部分（无批改）

- q1. 写一个自定义函数名为circle，接受一个数值型输入参数r，输出/返回数值型参数2个，分别为area和cf。把r理解为圆的半径，area=以r为半径的圆的面积，cf=周长。
- q2. 写一个自定义函数名为 flip_it，接受一个行向量 v，输出参数是行向量 w，使 w 成为 v 的逆序输出。如输入 v=[1 2 3]，输出 w=[3 2 1]。请不要使用 MATLAB 自带函数 flip,flipr,或者 flipud 等。
- q3. 写一个自定义函数名为zero_stat，接受一个数值型矩阵m，m的大小未知，但知道m中所有元素都是0或者1，输出参数p为m中所有等于0的元素所占总元素的比例。例如在一个5x5大小的矩阵中有10个0和15个1，则返回p值等于40因为零元素所占比例为40%。
- q4. 写一个自定义函数名为sum3and5，接受一个整型参数n。要求把1至n中所有是3或者5的倍数的数值单独列出并求和，例如n=15，那么是3或者5的倍数有3,5,6,9,10,12,15。将它们全部加起来得到60。将这个和定义为输出参数s作为函数的返回值。
- q5. 写一个自定义函数move_me，接受两个参数分别为一个行向量v，和一个数值a。要求将v中所有等于a的元素移到v的末尾，并将结果保存为新的行向量x作为输出参数。例如v=[1 2 3 4], a=2,则运行完x=move_me(v,a)后，x=[1 3 4 2]，如果函数没有提供a的值，则默认需要移动的元素值为0。