**用途与运行环境：**

本程序平台采用C#实现了基于Petri网的Anytime A\*搜索算法，详细算法描述请参见我的期刊论文。

本平台基于Visual Studio 2008开发，安装VS2008后直接双击后缀名为.sln或.csproj文件即可打开运行。如果采用的不同版本的VS则可能无法直接打开，此时需要新建空白C# Console工程，将Main.cs, AStar.cs和Heap.cs三个文件，以及bin文件夹和帮助文档拷贝至新建工程的所在目录，然后在工程中Project->add existing item将三个cs文件加入即可。（将老的程序备份，并创建新的程序也是此流程）

运行：打开main.cs，点击Debug->Start Debugging(F5)

**平台组成：**

Main.cs: 主程序，输入一些用户参数

AStar.cs: 搜索程序，将各搜索算法定义成不同的函数或重载函数

Heap.cs: 堆栈操作函数，用于处理OPEN, CLOSED之类的堆栈操作，通常不用改动，直接调用即可

Bin/Debug文件夹：存放输入和输出文件。输入文件，对于一个Petri网，需要给出两个输入文件，其中xxx\_matrix.txt放置Petri网的关联矩阵，xxx\_init.txt放置起始标识，库所操作时间信息和目的标识。输出文件为result.txt。

**注意：**

* 新增命名要规范，名字要有意义，且变量名首字母小写，函数名首字母大写等；
* 多写注释，方便自己和他人以后理解；新增句子和函数要注释作用、姓名和日期，如//xxxxxxxxxx-HB-20221023；
* 程序中的Mr即论文中的R，即托肯的剩余操作时间；
* 程序出现bug时，采用单步执行查看变量、设置断点、设置输出变量、打印中间结果等方法进行Debug；

**版本更新：**

1. 可以输入任意Petri网的initial和matrix文件；
2. 实现了TASE 2021的h\_H函数；
3. 实现了h\_H+h+e函数；

202104：

1. 将WRT, ResNum, Upi, M0r变成AStar的全局变量，避免循环重复定义与赋值；
2. 将大部分double变成decimal，避免计算出错，但WRT等少数还是double，因为赋值时需要数字后加m，所以还是用的double，在具体计算时，使用(decimal)进行强制类型转换；
3. 增加了向文件输出的每轮全局信息；

20221024：

1. 新增了平台使用word文档；
2. 修正了部分注释；
3. 修正了（1+e）f中挑选最深节点的循环比较bug；