

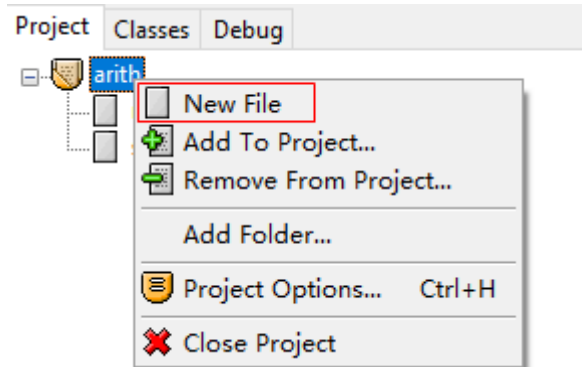
命令行参数

辽师张大为@<https://daweizh.github.io/csp/>

很多时候希望在程序启动时输入变化参数，我们把用这种方式输入的参数叫做命令行参数。

params.h

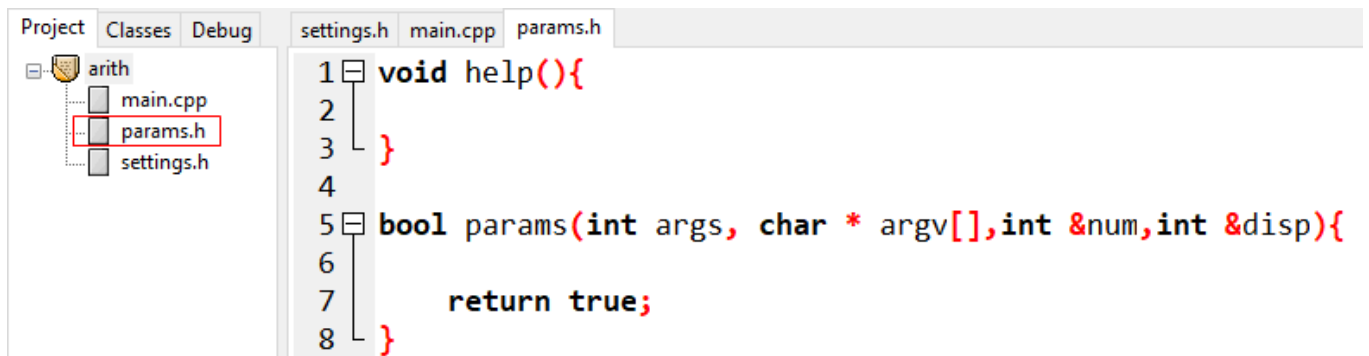
在项目arith中创建新文件



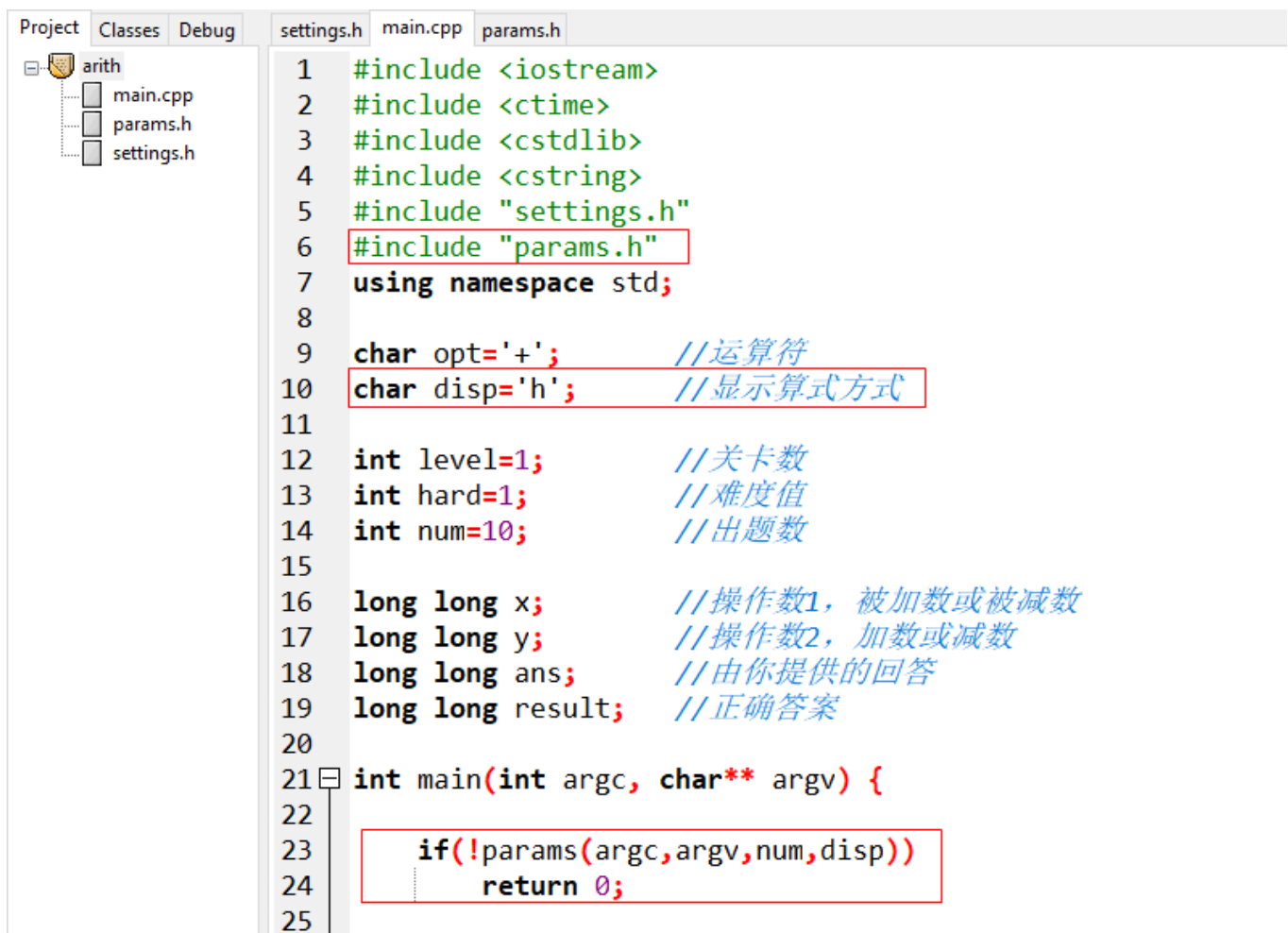
在文件中输入如下内容

```
void help(){  
  
}  
  
bool params(int args, char * argv[],int &num,char &disp){  
  
    return true;  
}
```

按Ctrl+s存盘，选择“保存类型”为“.h”，文件名为params.h



在main.cpp中引入如下语句



--help

考虑到在不同场合都可能出现提示软件使用方法的帮助信息，把显示help信息打包成一个独立的函数。

```
void help(){
    printf("Usage: game.exe [--help]|[--disp h|v]|[--numb 20]\n\n");
    printf("--help :display help\n");
    printf("--disp :show model of question. You can select h(horizontal) or v(vertical)\n");
    printf("--numb :question number\n\n");

    return ;
}
```

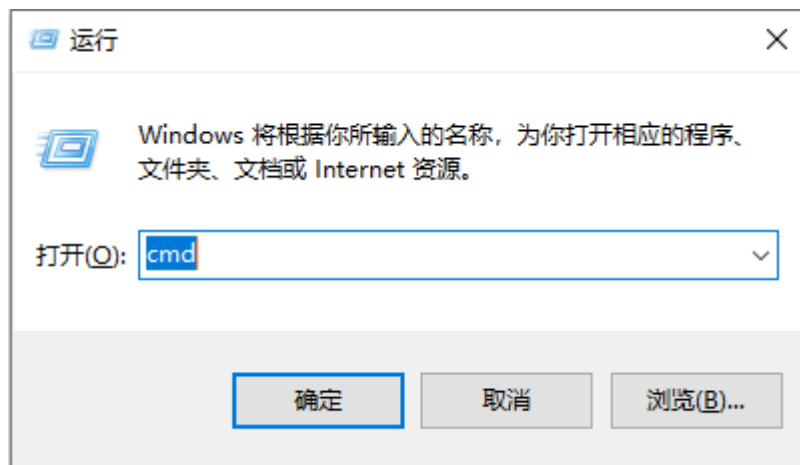
在params函数中以--help参数或出错的形式显示帮助信息

```
bool params(int args, char * argv[],int &num,char &disp){
    if(args>1){        //如果命令行有参数输入
        for(int i=1;i<args;i++){    //分析每个参数
            if(strcmp(argv[i],"--help")==0){    //如果参数是--help
                help();                        //显示帮助信息
                return false;                //结束游戏程序
            }else {                            //有命令行参数输入，但不是我们要的
                help();                        //显示帮助信息
                return false;                //结束游戏程序
            }
        }
    }
}
```

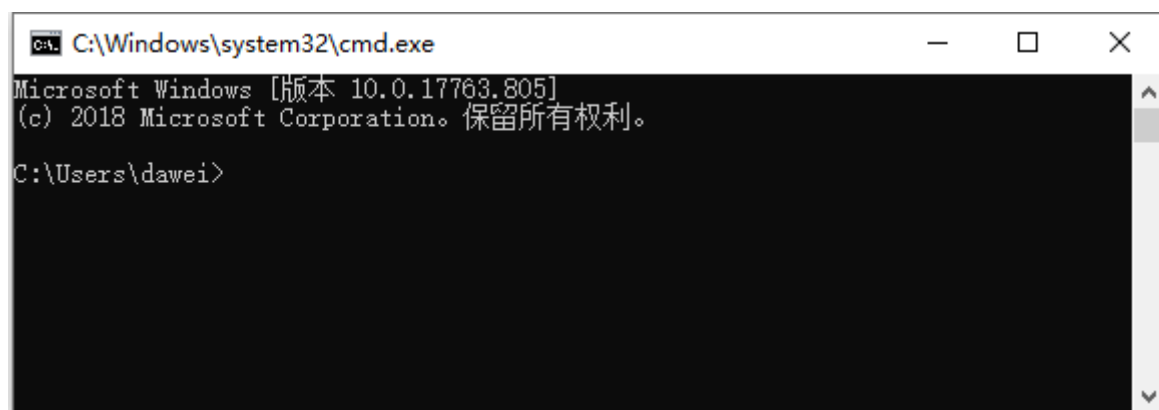
```
    }  
}  
  
return true;  
}
```

回到main.cpp获得焦点，按F12重新构造全部。

按Ctrl+r并输入cmd命令，出现运行窗口



点击确定，出现命令终端



通过DOS命令（根据自己的具体情况），进入代码空间

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.17763.805]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\dawei>f:

F:\>cd a.steam

F:\a.steam>cd dreamx

F:\a.steam\dreamx>cd wedo

F:\a.steam\dreamx\wedo>cd game

F:\a.steam\dreamx\wedo\game>cd arith

F:\a.steam\dreamx\wedo\game\arith>dir
驱动器 F 中的卷是 Mine
卷的序列号是 5439-F4B4

F:\a.steam\dreamx\wedo\game\arith 的目录
2019/11/05  15:39    <DIR>          .
2019/11/05  15:39    <DIR>          ..
2019/11/05  15:21             1,018 arith.dev
2019/11/05  15:39       1,925,874 arith.exe
2019/11/05  15:21             158 arith.layout
2019/11/05  15:38             1,067 main.cpp
2019/11/05  15:39             5,058 main.o
2019/11/05  15:39             1,119 Makefile.win
2019/11/05  15:49             714 params.h
2019/11/05  15:16             1,140 settings.h
                8 个文件      1,936,148 字节
                2 个目录  47,891,394,560 可用字节

F:\a.steam\dreamx\wedo\game\arith>_
```

测试命令行参数的执行情况：

1. 输入正确参数的情况

```
F:\a.steam\dreamx\wedo\game\arith>arith --help
Usage: game.exe [--help][[--disp h|v]][--numb 20]

--help :display help
--disp :show model of question. You can select h(horizontal) or v(vertical)
--numb :question number
```

2. 输入错误参数的情况

```
F:\a.steam\dreamx\wedo\game\arith>arith 3
Usage: game.exe [--help][[--disp h|v]][--numb 20]

--help :display help
--disp :show model of question. You can select h(horizontal) or v(vertical)
--numb :question number
```

--disp

在params函数中增加对--disp的判断和处理

```

10 bool params(int args, char * argv[], int &num, char &disp){
11     if(args>1){ //如果命令行有参数输入
12         for(int i=1; i<args; i++){ //分析每个参数
13             if(strcmp(argv[i], "--help")==0){ //如果参数是--help
14                 help(); //显示帮助信息
15                 return false; //结束游戏程序
16             }
17             else if(strcmp(argv[i], "--disp")==0){ //如果输入的参数中有--disp
18                 if(++i<args){ //而且--disp后面还有参数
19                     if(strcmp(argv[i], "h")==0){ //取--disp后面的参数看看是不是h
20                         disp = 'h'; //是h, disp='h'
21                     } else if(strcmp(argv[i], "v")==0){ //不是h, 看看是不是v
22                         disp = 'v'; //如果是v, disp='v'
23                     } else{ //--disp后面跟的既不是h, 也不是v
24                         help(); //--disp后面的参数是错的, 显示帮助
25                         return false; //结束游戏程序
26                     }
27                 } else{ //如果参数中有--help, 但后面没有更多参数
28                     help(); //显示帮助
29                     return false; //结束游戏程序
30                 }
31             }
32         } else { //有命令行参数输入, 但不是我们要的
33             help(); //显示帮助信息
34             return false; //结束游戏程序
35         }
36     }
37 }
38
39 return true;
40 }

```

回到main.cpp获得焦点, 按F12重新构造全部。

回到命令终端, 测试--disp正确和错误情况

1. 正确输入参数情况

```

F:\a.steam\dreamx\wedo\game\arith>arith --disp v
Please select level([1]-18):
Please select operator([+],-):
Please select Hard([1]-60):
level=1,opt=+,hard=1
... ..

```

2. --disp后没有参数情况

```

F:\a.steam\dreamx\wedo\game\arith>arith --disp
Usage: game.exe [--help][--disp h|v][--numb 20]

--help :display help
--disp :show model of question. You can select h(horizontal) or v(vertical)
--numb :question number

```

3. --disp后输入错误参数情况

```

F:\a.steam\dreamx\wedo\game\arith>arith --disp 3
Usage: game.exe [--help][--disp h|v][--numb 20]

```

```
--help :display help
--disp :show model of question. You can select h(horizontal) or v(vertical)
--numb :question number
```

--numb

在params函数中增加对--numb的判断和处理

```
32 else if(strcmp(argv[i],"--numb")==0){ //如果输入的参数中有--numb
33     if(++i<args ){ //而且--numb后面还有参数
34         int qs = atoi(argv[i]); //将输入参数转化为数值
35         if(qs>0){ //如果数值>0
36             num = qs; //赋值给num
37         }else{ //如果数值为0
38             help(); //显示帮助
39             return false; //结束游戏程序
40         }
41     }else{ //如果--numb后面没有参数
42         help(); //显示帮助信息
43         return false; //结束游戏程序
44     }
45 }
```

回到main.cpp获得焦点，按F12重新构造全部。

回到命令终端，测试--numb正确和错误情况

1. 正确输入参数情况

```
F:\a.steam\dreamx\wedo\game\arith>arith --numb 5
Please select level([1]-18):
Please select operator([+,-]):
Please select Hard([1]-60):
level=1,opt=+,hard=1
... ..
```

2. --disp后没有参数情况

```
F:\a.steam\dreamx\wedo\game\arith>arith --numb
Usage: game.exe [--help][--disp h|v][--numb 20]

--help :display help
--disp :show model of question. You can select h(horizontal) or v(vertical)
--numb :question number
```

3. --disp后输入错误参数情况

```
F:\a.steam\dreamx\wedo\game\arith>arith --numb d
Usage: game.exe [--help][--disp h|v][--numb 20]

--help :display help
--disp :show model of question. You can select h(horizontal) or v(vertical)
--numb :question number
```

检查参数设置情况

修改main.cpp

```
21 int main(int argc, char** argv) {  
22  
23     if(!params(argc,argv,num,disp))  
24         return 0;  
25  
26     settings(level,opt,hard);  
27     printf("level=%d,opt=%c,hard=%d,disp=%c,numb=%d\n\n",level,opt,hard,disp,num);  
28
```

在main.cpp获得焦点，按F12重新构造全部。

回到命令终端，测试全部参数的正确性

```
F:\a.steam\dreamx\wedo\game\arith>arith --numb 5 --disp v  
Please select level([1]-18):  
Please select operator([+],-):  
Please select Hard([1]-60):  
level=1,opt=+,hard=1,disp=v,numb=5
```