

单线程累加求和程序编写



单线程累加求和程序编写



运动程序完成累加求和任务。定义了全局变量sum用于保存累加和，局部变量begin用于保存累加起点，局部变量end用于保存累加终点。累加完成以后程序结束。

1. 新建控制台项目

在Visual Studio中新建项目工程。

2. 调用库及配置文件

将工程中需要使用的动态链接库、头文件以及控制器配置文件拷贝到项目的源文件目录下。

3. 添加库文件

在项目-属性-链接器-输入-附加依赖项中添加gts.lib库文件；添加库文件的另一种方法是，在程序中使用#pragma comment(lib, "gts.lib")。

4. 添加头文件

将代码中需要使用到的指令的头文件包含到程序中。

5. 程序实现



01

单线程累加求和程序编写

```
// -----  
// 累加求和  
// begin 累加起点  
// end 累加终点  
// -----  
int sum;  
int add(int begin, int end)  
{  
    int i;  
    int cc;  
    i = begin;  
lbl_loop:  
    cc = i>end;  
    if(cc) goto lbl_end;  
    sum = sum + i;  
    i = i + 1;  
    goto lbl_loop;  
lbl_end:  
    return sum;  
}
```



01

单线程累加求和程序编写

//应用程序负责编译、下载、初始化、启动运动程序。

```
#include <iostream>
#include "conio.h"
#include "gts.h"
int main(int argc, char* argv[])
{
    short rtn;
    short funId;
    TVarInfo sum, begin, end;
    double value;
    TThreadSts thread;
    // 打开运动控制器
    rtn = GT_Open();
    printf("GT_Open()=%d\n", rtn);
    // 复位运动控制器
    rtn = GT_Reset();
    printf("GT_Reset()=%d\n", rtn);
```



01

单线程累加求和程序编写

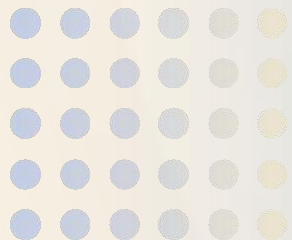
```
// 下载运动程序sum.bin,  
// 必须保证sum.bin文件位于工程文件夹中  
rtn = GT_Download("sum.bin");  
printf("GT_Download()=%d\n", rtn);  
// 获取函数ID  
rtn = GT_GetFunId("add", &funId);  
printf("GT_GetFunId()=%d\n", rtn);  
// 获取全局变量sum的ID  
rtn = GT_GetVarId(NULL, "sum", &sum);  
printf("GT_GetVarId()=%d\n", rtn);  
// 获取局部变量begin的ID  
rtn = GT_GetVarId("add", "begin", &begin);  
printf("GT_GetVarId()=%d\n", rtn);  
// 获取局部变量end的ID  
rtn = GT_GetVarId("add", "end", &end);  
printf("GT_GetVarId()=%d\n", rtn);  
// 绑定线程，函数，数据页  
rtn = GT_Bind(0, funId, 0);  
printf("GT_Bind()=%d\n", rtn);  
value = 0;
```



01

单线程累加求和程序编写

```
// 初始化运动程序的全局变量sum  
rtn = GT_SetVarValue(-1, &sum, &value);  
printf("GT_SetVarValue()=%d\n", rtn);  
value = 1;  
// 初始化运动程序的局部变量begin  
rtn = GT_SetVarValue(0, &begin, &value);  
printf("GT_SetVarValue()=%d\n", rtn);  
value = 100;  
// 初始化运动程序的局部变量end  
rtn = GT_SetVarValue(0, &end, &value);  
printf("GT_SetVarValue()=%d\n", rtn);  
// 启动线程  
rtn = GT_RunThread(0);  
printf("GT_RunThread()=%d\n", rtn);
```



01

单线程累加求和程序编写

```
do
{
// 查询线程状态
rtn = GT_GetThreadSts(0, &thread);
// 查询全局变量sum的值
rtn = GT_GetVarValue(-1, &sum, &value);
printf("run=%d sum=%-10.0lf\n", thread.run, value);
}while( 1 == thread.run ); // 等待线程运行结束
return 0;
}
```



谢谢观看

