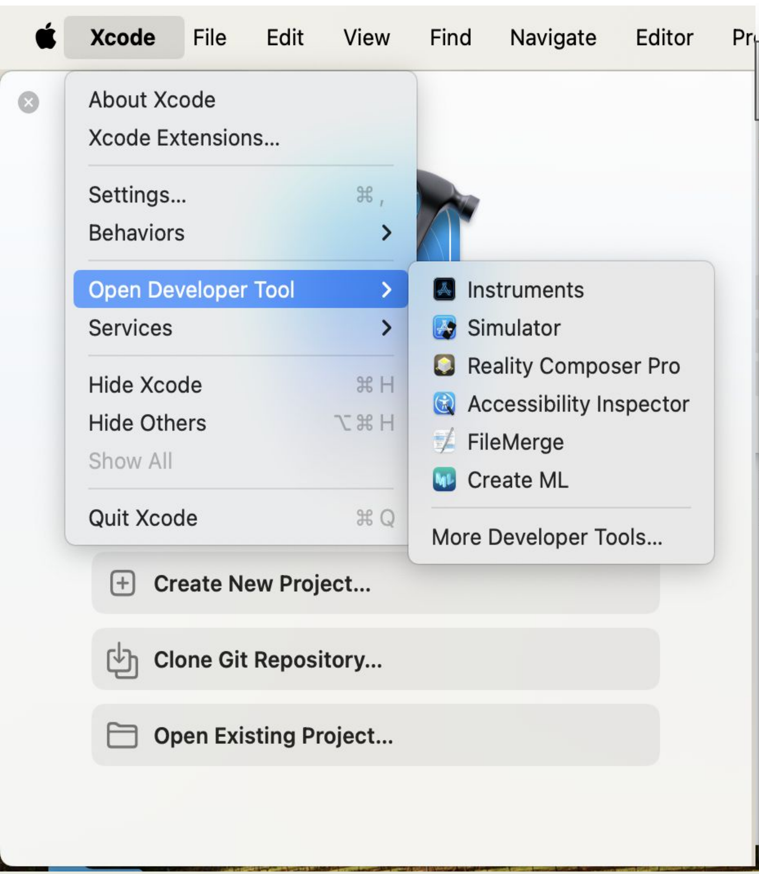
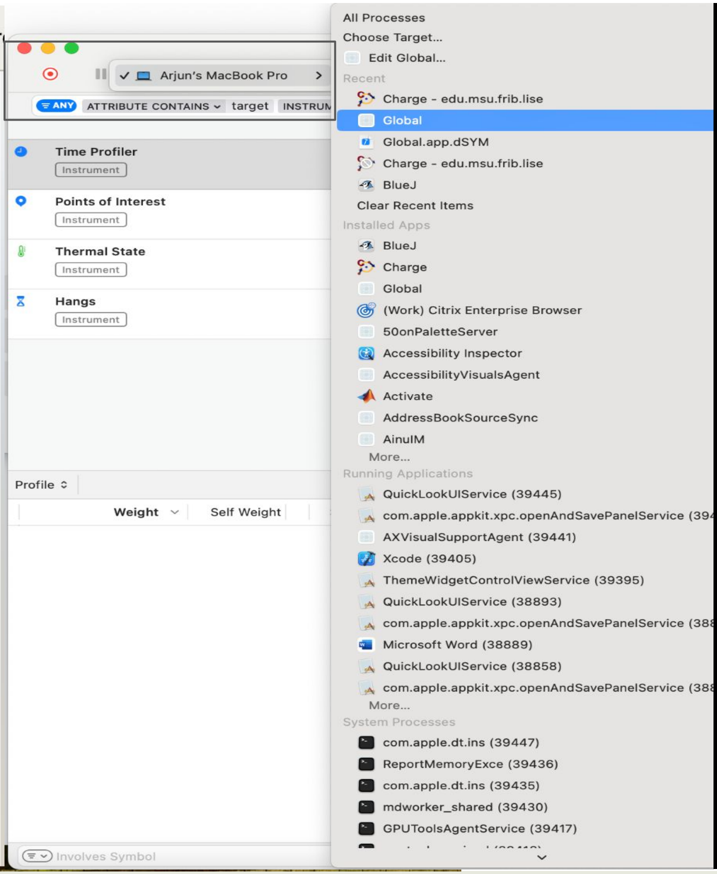
GLOBAL : PROFILING AND PARALLELIZATION

Arjun Ray
Undergraduate Research Assistant I
raya@frib.msu.edu

PROFILER SETUP AND USE : MAC OS





-OPEN XCODE

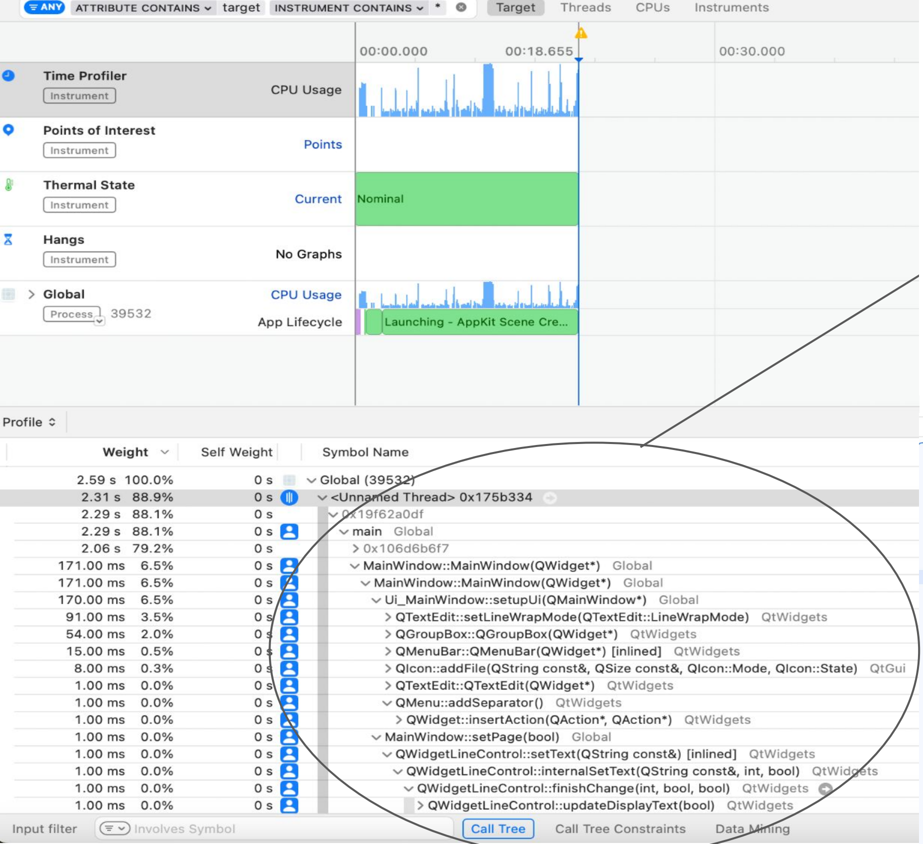
-NAVIGATE TO INSTRUMENTS AND THEN TO TIME PROFILER

-ATTACH THE EXECUTABLE FILE TO THE PROFILER AND RUN

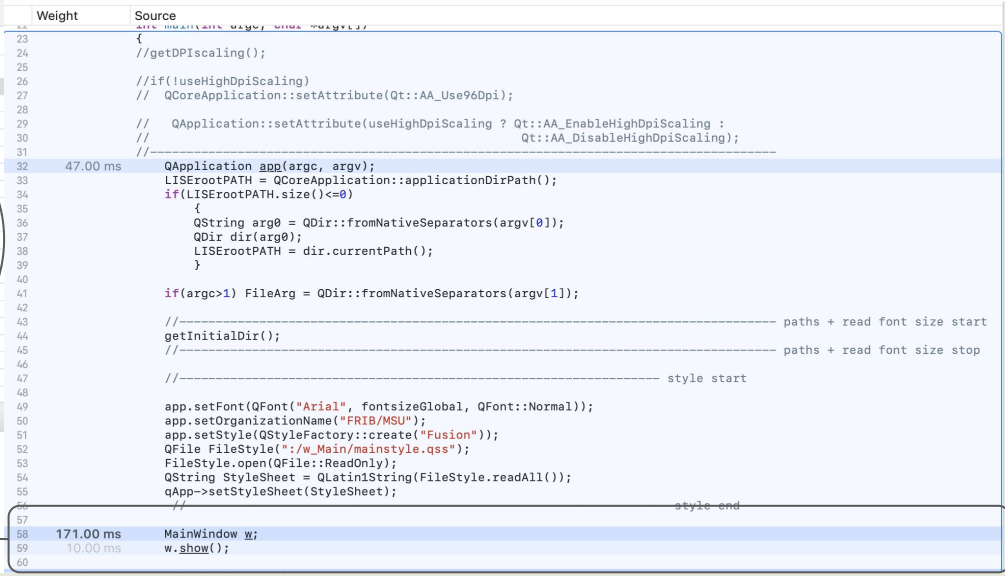
-RUN THE PROGRAM THOROUGHLY

-ONCE DONE, EXIT THE PROGRAM AND OPEN THE MAIN THREAD

-THE TOP SUBTHREAD IS THE PART OF THE PROGRAM WITH THE HIGHEST RUNTIME



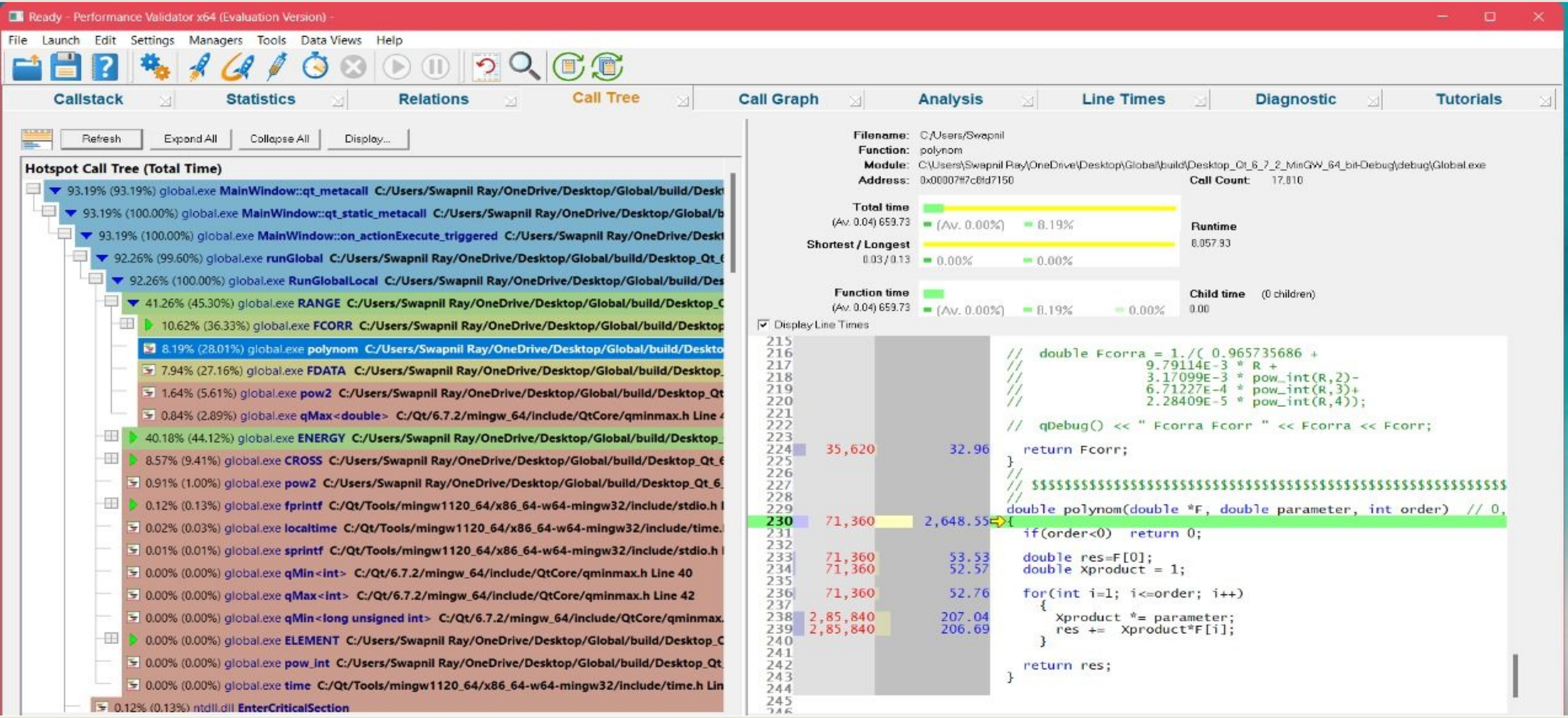
```
    Weight         Self Weight   Symbol Name
  2.59 s  100.0%      0 s        Global (39532)
  2.31 s   88.9%      0 s          <Unnamed Thread> 0x175b334
  2.29 s   88.1%      0 s            0x19f62a0df
  2.29 s   88.1%      0 s              main  Global
  2.06 s   79.2%      0 s                > 0x106d6b6f7
171.00 ms   6.5%      0 s                MainWindow::MainWindow(QWidget*)  Global
171.00 ms   6.5%      0 s                  MainWindow::MainWindow(QWidget*)  Global
170.00 ms   6.5%      0 s                    Ui_MainWindow::setupUi(QMainWindow*)  Global
 91.00 ms   3.5%      0 s                      > QTextEdit::setLineWrapMode(QTextEdit::LineWrapMode)  QtWidgets
 54.00 ms   2.0%      0 s                      > QGroupBox::QGroupBox(QWidget*)  QtWidgets
 15.00 ms   0.5%      0 s                      > QMenuBar::QMenuBar(QWidget*) [inlined]  QtWidgets
  8.00 ms   0.3%      0 s                      > QIcon::addFile(QString const&, QSize const&, QIcon::Mode, QIcon::State)  QtGui
  1.00 ms   0.0%      0 s                      > QTextEdit::QTextEdit(QWidget*)  QtWidgets
  1.00 ms   0.0%      0 s                      QMenu::addSeparator()  QtWidgets
  1.00 ms   0.0%      0 s                        > QWidget::insertAction(QAction*, QAction*)  QtWidgets
  1.00 ms   0.0%      0 s                    MainWindow::setPage(bool)  Global
  1.00 ms   0.0%      0 s                      QWidgetLineControl::setText(QString const&) [inlined]  QtWidgets
  1.00 ms   0.0%      0 s                        QWidgetLineControl::internalSetText(QString const&, int, bool)  QtWidgets
  1.00 ms   0.0%      0 s                          QWidgetLineControl::finishChange(int, bool, bool)  QtWidgets
  1.00 ms   0.0%      0 s                            > QWidgetLineControl::updateDisplayText(bool)  QtWidgets
```



```
Weight      Source
            {
            //getDPIscaling();

            //if(!useHighDpiScaling)
            //  QCoreApplication::setAttribute(Qt::AA_Use96Dpi);

            //   QApplication::setAttribute(useHighDpiScaling ? Qt::AA_EnableHighDpiScaling :
            //                                                   Qt::AA_DisableHighDpiScaling);
            //-----------------------------------------------------------------------------
 47.00 ms       QApplication app(argc, argv);
                LISErootPATH = QCoreApplication::applicationDirPath();
                if(LISErootPATH.size()<=0)
                    {
                    QString arg0 = QDir::fromNativeSeparators(argv[0]);
                    QDir dir(arg0);
                    LISErootPATH = dir.currentPath();
                    }

                if(argc>1) FileArg = QDir::fromNativeSeparators(argv[1]);

                //------------------------------------------------------------------- paths + read font size start
                getInitialDir();
                //------------------------------------------------------------------- paths + read font size stop

                //--------------------------------------------------------- style start

                app.setFont(QFont("Arial", fontsizeGlobal, QFont::Normal));
                app.setOrganizationName("FRIB/MSU");
                app.setStyle(QStyleFactory::create("Fusion"));
                QFile FileStyle(":/w_Main/mainstyle.qss");
                FileStyle.open(QFile::ReadOnly);
                QString StyleSheet = QLatin1String(FileStyle.readAll());
                qApp->setStyleSheet(StyleSheet);
                //--------------------------------------------------------- style end

171.00 ms       MainWindow w;
 10.00 ms       w.show();
```

-CLICK ON THE THREAD TO SEE SOURCE CODE AND LINE TIMES

CALLTREE(GLOBAL):PERCENTAGE RUNTIME


Ready - Performance Validator x64 (Evaluation Version) -
File Launch Edit Settings Managers Tools Data Views Help
Callstack Statistics Relations Call Tree Call Graph Analysis Line Times Diagnostic Tutorials
Refresh Expand All Collapse All Display...

Hotspot Call Tree (Total Time)

```
93.19% (93.19%) global.exe MainWindow::qt_metacall C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Deskt
93.19% (100.00%) global.exe MainWindow::qt_static_metacall C:/Users/Swapnil Ray/OneDrive/Desktop/Global/b
93.19% (100.00%) global.exe MainWindow::on_actionExecute_triggered C:/Users/Swapnil Ray/OneDrive/Deskt
92.26% (99.60%) global.exe runGlobal C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_Qt_6
92.26% (100.00%) global.exe RunGlobalLocal C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Des
41.26% (45.30%) global.exe RANGE C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_C
10.62% (36.33%) global.exe FCORR C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop
8.19% (28.01%) global.exe polynom C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Deskto
7.94% (27.16%) global.exe FDATA C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_
1.64% (5.61%) global.exe pow2 C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_Qt
0.84% (2.89%) global.exe qMax<double> C:/Qt/6.7.2/mingw_64/include/QtCore/qminmax.h Line 4
40.18% (44.12%) global.exe ENERGY C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_
8.57% (9.41%) global.exe CROSS C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_Qt_6
0.91% (1.00%) global.exe pow2 C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_Qt_6_
0.12% (0.13%) global.exe fprintf C:/Qt/Tools/mingw1120_64/x86_64-w64-mingw32/include/stdio.h l
0.02% (0.03%) global.exe localtime C:/Qt/Tools/mingw1120_64/x86_64-w64-mingw32/include/time.
0.01% (0.01%) global.exe sprintf C:/Qt/Tools/mingw1120_64/x86_64-w64-mingw32/include/stdio.h l
0.00% (0.00%) global.exe qMin<int> C:/Qt/6.7.2/mingw_64/include/QtCore/qminmax.h Line 40
0.00% (0.00%) global.exe qMax<int> C:/Qt/6.7.2/mingw_64/include/QtCore/qminmax.h Line 42
0.00% (0.00%) global.exe qMin<long unsigned int> C:/Qt/6.7.2/mingw_64/include/QtCore/qminmax.
0.00% (0.00%) global.exe ELEMENT C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_C
0.00% (0.00%) global.exe pow_int C:/Users/Swapnil Ray/OneDrive/Desktop/Global/build/Desktop_Qt
0.00% (0.00%) global.exe time C:/Qt/Tools/mingw1120_64/x86_64-w64-mingw32/include/time.h Lin
0.12% (0.13%) ntdll.dll EnterCriticalSection
```

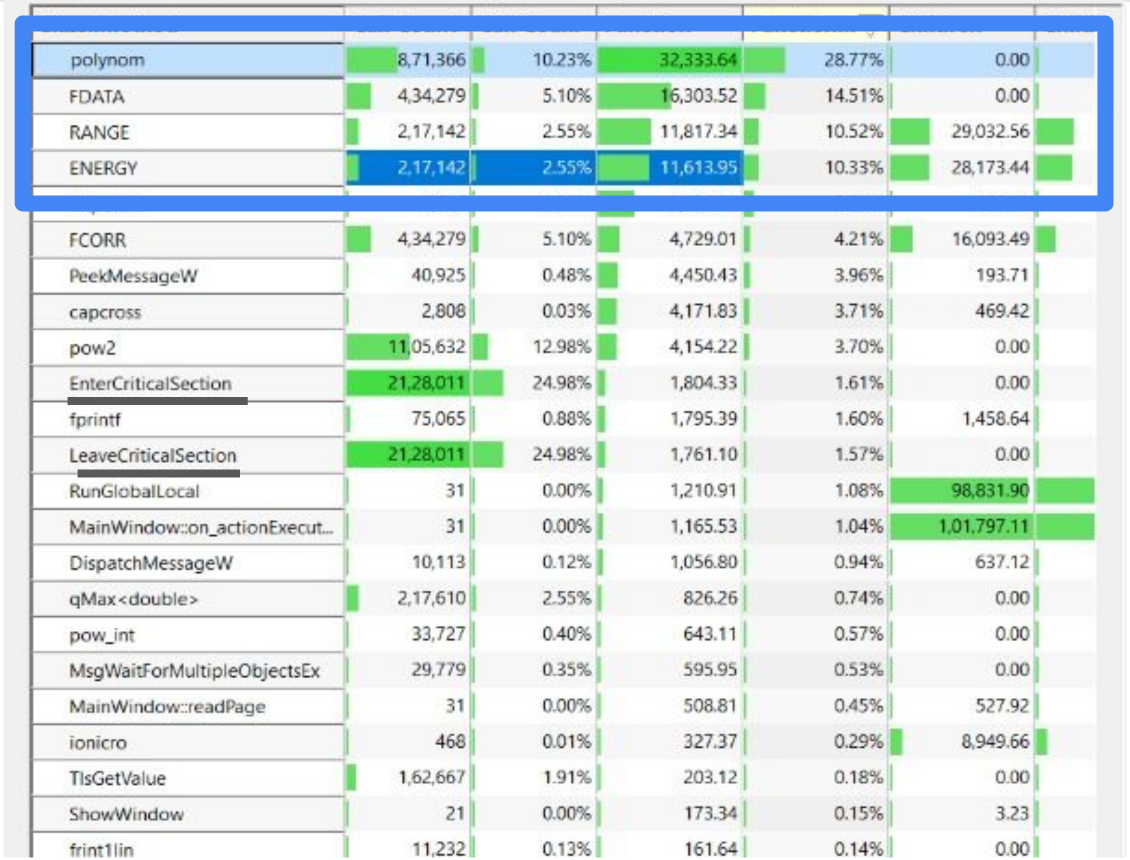
Filename: C:/Users/Swapnil
Function: polynom
Module: C:\Users\Swapnil Ray\OneDrive\Desktop\Global\build\Desktop_Qt_6_7_2_MinGW_64_bit-Debug\debug\Global.exe
Address: 0x00007ff7c8fd7150
Call Count: 17,810

Total time (Av. 0.04) 659.73 — (Av. 0.00%) 8.19%
Shortest / Longest 0.03 / 0.13 — 0.00% 0.00%
Runtime 8,057.93
Function time (Av. 0.04) 659.73 — (Av. 0.00%) 8.19% 0.00%
Child time (0 children) 0.00

[✓] Display Line Times

```
                         //  double Fcorra = 1./( 0.965735686 +
                         //                  9.79114E-3 * R +
                         //                  3.17099E-3 * pow_int(R,2)-
                         //                  6.71227E-4 * pow_int(R,3)+
                         //                  2.28409E-5 * pow_int(R,4));

                         //  qDebug() << " Fcorra Fcorr " << Fcorra << Fcorr;

   35,620        32.96     return Fcorr;
                         }
                         //
                         // $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                         //
                         double polynom(double *F, double parameter, int order)  // 0,
   71,360     2,648.55   {
                           if(order<0)   return 0;

   71,360        53.53     double res=F[0];
   71,360        52.57     double Xproduct = 1;

   71,360        52.76     for(int i=1; i<=order; i++)
                             {
 2,85,840       207.04         Xproduct *= parameter;
 2,85,840       206.69         res +=  Xproduct*F[i];
                             }

                           return res;
                         }
```

SLOWEST FUNCTION RUNTIMES



polynom	8,71,366	10.23%	32,333.64	28.77%	0.00
FDATA	4,34,279	5.10%	16,303.52	14.51%	0.00
RANGE	2,17,142	2.55%	11,817.34	10.52%	29,032.56
ENERGY	2,17,142	2.55%	11,613.95	10.33%	28,173.44
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
FCORR	4,34,279	5.10%	4,729.01	4.21%	16,093.49
PeekMessageW	40,925	0.48%	4,450.43	3.96%	193.71
capcross	2,808	0.03%	4,171.83	3.71%	469.42
pow2	11,05,632	12.98%	4,154.22	3.70%	0.00
EnterCriticalSection	21,28,011	24.98%	1,804.33	1.61%	0.00
fprintf	75,065	0.88%	1,795.39	1.60%	1,458.64
LeaveCriticalSection	21,28,011	24.98%	1,761.10	1.57%	0.00
RunGlobalLocal	31	0.00%	1,210.91	1.08%	98,831.90
MainWindow::on_actionExecut...	31	0.00%	1,165.53	1.04%	1,01,797.11
DispatchMessageW	10,113	0.12%	1,056.80	0.94%	637.12
qMax<double>	2,17,610	2.55%	826.26	0.74%	0.00
pow_int	33,727	0.40%	643.11	0.57%	0.00
MsgWaitForMultipleObjectsEx	29,779	0.35%	595.95	0.53%	0.00
MainWindow::readPage	31	0.00%	508.81	0.45%	527.92
ionicro	468	0.01%	327.37	0.29%	8,949.66
TlsGetValue	1,62,667	1.91%	203.12	0.18%	0.00
ShowWindow	21	0.00%	173.34	0.15%	3.23
frint1lin	11,232	0.13%	161.64	0.14%	0.00

—-FDATA is the function with the largest runtime

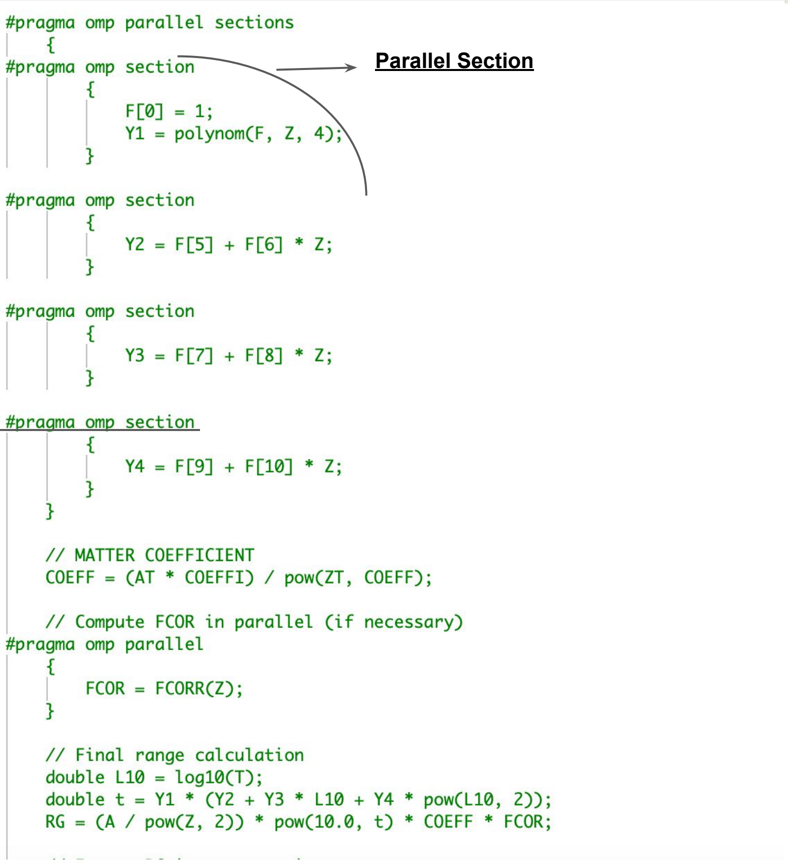
—Even though FDATA is the function with the largest runtime, polynom is called more, and thus takes the highest runtime.

—Other functions with a high runtime are RANGE and ENERGY

—Since polynom is already optimized, optimization wa done for range AND FDATA

—-EnterCriticalSection and LeaveCriticalSection- These functions are used to protect shared resources from being simultaneously accessed by multiple threads.

PARALLELIZATION CODE IMPLEMENTATION AND EXPLANATION



```
#pragma omp parallel sections
    {
#pragma omp section
        {
            F[0] = 1;
            Y1 = polynom(F, Z, 4);
        }

#pragma omp section
        {
            Y2 = F[5] + F[6] * Z;
        }

#pragma omp section
        {
            Y3 = F[7] + F[8] * Z;
        }

#pragma omp section
        {
            Y4 = F[9] + F[10] * Z;
        }
    }

    // MATTER COEFFICIENT
    COEFF = (AT * COEFFI) / pow(ZT, COEFF);

    // Compute FCOR in parallel (if necessary)
#pragma omp parallel
    {
        FCOR = FCORR(Z);
    }

    // Final range calculation
    double L10 = log10(T);
    double t = Y1 * (Y2 + Y3 * L10 + Y4 * pow(L10, 2));
    RG = (A / pow(Z, 2)) * pow(10.0, t) * COEFF * FCOR;
```

Parallelization Using OpenMP:

- Utilized OpenMP for parallel processing to optimize calculations across different sections, improving execution speed for large datasets.

Parallel Sections for Polynomial Terms:

- Split polynomial calculations for terms `Y1`, `Y2`, `Y3`, and `Y4` into separate parallel sections using `#pragma omp parallel sections` to reduce computation time.

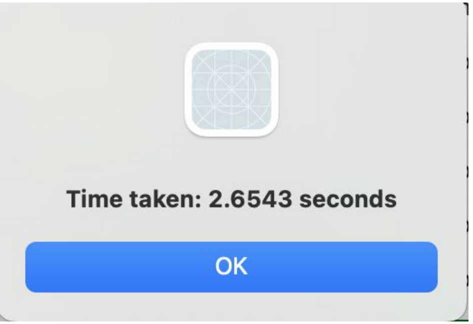
Parallelization of Correction Factor (`FCOR`):

- Computed `FCOR` (a correction factor) in a separate parallel block, ensuring faster adjustment for particle characteristics.

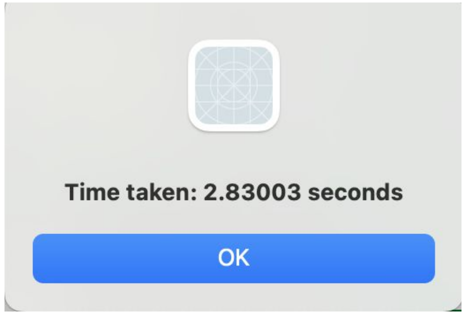
Optimization Outcome:

- Through parallel sections, computational efficiency improves significantly for large datasets, leveraging multi-threading for faster processing.

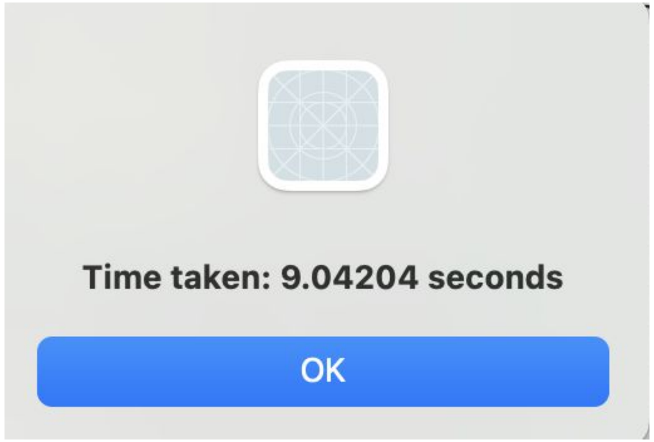
PARALLELIZATION RESULTS


Time taken: 2.6543 seconds

Using new RANGE Function


Time taken: 2.83003 seconds

With Old RANGE Function


Time taken: 9.04204 seconds

Using new FDATA Function

–The new function FDATA makes the program considerably slow.

–Implementing parallelization in both functions, FDATA and RANGE, program Global seems to be working faster only when the new RANGE function is implemented.

–This change is not very noticeable, even for heavy calculations

– Research would suggest there is no point in parallelization of FDATA

IMPLEMENTATION AND USE OF TIMER

```
auto start = std::chrono::high_resolution_clock::now();
result = runGlobal (fno, false, tGlobal_version,
                   gAF,  gZF,  gAT,  gZT,
                   gDTARGET,  gEN0,   gQIN,
                     I_WR,     iOption,    iCSoutput,
                     iLoop,  N_Steps,   Qshow,
                      DELZF,  DELE,
                      DELQ,   DELZT,  DELDT);
auto end = std::chrono::high_resolution_clock::now();
std::chrono::duration<double> duration = end - start;
if (ui->timerCheckBox->isChecked()) { // Check if the timer checkbox is checked
    QString timeMessage = QString("Time taken: %1 seconds").arg(duration.count());
    QMessageBox::information(this, "Execution Time", timeMessage);
}
```

-CODE TO FIND PROGRAM RUNTIME

-DISPLAYS SAID TIME IN A POPUP WINDOW

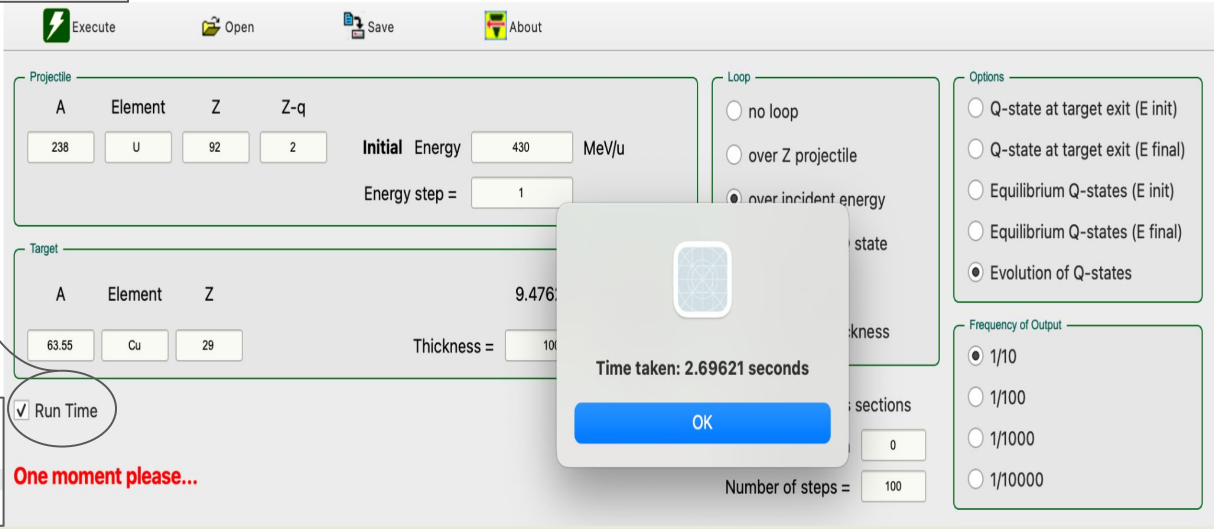
-USES INBUILT CHRONO(HIGH PRECISION TIMER) FUNCTION TO CALCULATE RUNTIME

-RUN TIME CALCULATOR NOW IMPLEMENTED IN GLOBAL

-CHECK TICK BOX TO DISPLAY RUNTIME

-SOURCE CODE FOR IMPLEMENTATION

```
if (ui->timerCheckBox->isChecked()) { // Check if the timer checkbox is checked
    QString timeMessage = QString("Time taken: %1 seconds").arg(duration.count());
    QMessageBox::information(this, "Execution Time", timeMessage);
}
```



Projectile

A	Element	Z	Z-q
238	U	92	2

Initial Energy 430 MeV/u

Energy step = 1

Target

A	Element	Z
63.55	Cu	29

9.476

Thickness = 10

[✓ Run Time]

One moment please...

Loop
- no loop
- over Z projectile
- [✓ over incident energy]
- state
- kness
- sections

0

Number of steps = 100

Options
- Q-state at target exit (E init)
- Q-state at target exit (E final)
- Equilibrium Q-states (E init)
- Equilibrium Q-states (E final)
- [✓ Evolution of Q-states]

Frequency of Output
- [✓ 1/10]
- 1/100
- 1/1000
- 1/10000

Time taken: 2.69621 seconds

OK