```
// ==================== Settings Class  ==================
// Developed by Bryan V. Egner, Darren E. Holland, and Julie V. Logan
// Modified by Darren Holland 2020-11-02
// ========================================================================
//
// This file contains settings needed for the Geant geometry creation and
// subseqent analysis
// ==================== IMPORT ClASSES ====================================
//
#include <string>
#include <vector>

using namespace std;
namespace Settings
{
      // Node element lists:
      extern const int RSMTet(0);
                        // If 1 assumes elements are Tets (original RSM design)
                                    // otherwise uses TesselatedSolids (newer
designs)
      extern std::string fname_nodes("RSM_nodes.inp");
      extern std::string fname_ele("RSM_elements.inp");

      // Output filename:
      extern std::string fname_out("Ofile");

      // Source info: (These are overwritten)
      extern const int nParts2Run(50000);     // Number of source particles
    extern std::string SourceEnergyType("AmBe.mac");// Source Energy Spectrum
      extern std::string PartType("gamma");     // Source particle
      extern const int SourceDiv(2);           // Source position sub-divisions
      extern const double SourceDist(86.36);  // Source distance (cm)
      extern const double coneangle(17.5);      // Cone angle for variance
reduction
      extern const int numEnergies(1);        // Number of source energies
(currently commented out)
      extern const double energiesMeV(0.662); // Source particle energy

      // Mask Angles:
      extern const double deltatheta(10);     // Theta increment
      extern const double deltaphi(10);       // Phi increment

      // Geometry (in cm):
      extern const double DetRad(3.81);       // Detector radius
      extern const double DetHeight(3.81);      // Detector half height
      extern const double SleeveOuterRad(4.1275);  // Sleeve radius
      extern const double SleeveHeight(5);      // Top of sleeve (extends past
detector)
      extern const double SleeveBottom(55);     // Total length of sleeve
    // Vectors for running batch (do not change):
      extern const double StartPhi(0);        // Initial phi measurement
position
      extern const double EndPhi(170);        // Final phi measurement position
}
```