Help file Project 5

The following are the headers of the two classes: CreatureClass and MazeClass:

```
struct coordinate
{
        int row, col;
};

enum SquareType {Wall, Clear, Visited, Path};

class CreatureClass
{
  public:
        CreatureClass();
      ~CreatureClass();
      void MoveUp();
      void MoveDown();
      void MoveLeft();
      void MoveRight();
      void AssignLocation(coordinate);
      coordinate ReportLocation();
  private:
      coordinate position;
};

class MazeClass
{   public:
      MazeClass();
      ~MazeClass();
      void ReadMaze(ifstream&);
      void DisplayMaze();
      bool IsWall(coordinate);
      bool IsClear(coordinate);
      bool IsPath(coordinate);
      bool IsVisited(coordinate);
      bool IsExit(coordinate);
      bool InMaze(coordinate);
      void MarkPath(coordinate);
      void MarkVisited(coordinate);
      coordinate GetEntrance();
   private:
      SquareType **Maze;
      coordinate entrance, Exit;
      int       height, width;
};
```

The pseudoCode for the client program function GoNorth() is defined as following:

```
void  GoNorth(maze, oneCreature, success)
        create a creature
        put creature at the location of "oneCreature"
        if (the square to the north is inside the maze, clear, and unvisited)
        {       move to the north → move up
                mark the square as part of the path
                if (at exit)
                        success = true;
                else
                {  GoNorth(maze, creature, success);
                   if (!success)
                   {
                        GoWest(maze, creature, success);
                        if (!success)
                        {
                           GoEast(maze, creature, success);
                           if  (!success)
                           {  Mark square visited
                              GoSouth(maze, creature, success);
                           }
                        }
                   }
                }
          }
          else
              success = false;
```

Partial translation from the above pseudo code to C++ language:

```
void GoNorth(MazeClass &maze, CreatureClass &oneCreature, bool & success) {
        CreatureClass creature;
        coordinates   tmpPos = oneCreature.ReportLocation();
        creature.SetLocation(tmpPos);
        tmpPos.row --;
        if (maze.InMaze(tmpPos) && maze.IsEmpty(tmpPos) &&
!maze.IsVisited(tmpPos))
        {
                creature.MoveUp();
                tmpPos=creature.ReportLocation();
                maze.MarkPath(tmpPos);
                …..
```

Make sure MoveUp is consistent in changing row value: row—

MoveLeft is consistent in changing col value: col--