```
/****
 *
 * Class Edit is the model class for the rolodex edit handling.  This is an
 * initial skeletal design with no preconditions, postconditions, or data
 * represenation yet defined.
 *
 */

#ifndef editIncluded
#define editIncluded

#include "basic-types.h"
#include "model.h"
#include "str.h"


class Edit : public Model {
  public:
    Edit(View* v = NULL);
    /*
     * Construct this.
     */

    ~Edit();
    /*
     * Destruct this.
     */

    virtual void Undo();
    /*
     * Undo the last edit operation.  This function can be specialized by
     * subclasses to undo the last operation in a specific tool context.
     */

    virtual void Redo();
    /*
     * Redo the last undone operation.
     */

    virtual void Repeat(int n);
    /*
     * Repeat the last edit operation, n times.  This function can be
     * specialized by subclasses to repeat the last operation in a specific
     * tool context.
     */

    virtual void Cut();
    /*
     * Cut the currently selected text.  This function can be specialized by
     * subclasses to cut a tool-specific object
     */

    virtual void Copy();
    /*
     * Copy the currently selected text.  This function can be specialized by
     * subclasses to copy a tool-specific object
     */

    virtual void Paste();
    /*
     * Paste the most recxently cut or copied text.  This function can be
     * specialized by subclasses to past a tool-specific object
     */

    virtual void Delete();
    /*
     * Delete the currently selected text.  This function can be specialized by
     * subclasses to delete a tool-specific object.
     */

    virtual void SelectAll();
    /*
     * Select all text objects in the current canvas.  This function can be
     * specialized by subclasses to select tool-specific objects.
     */

    virtual void Find(String* s);
    /*
     * Find the given string as text in the current canavas.  This function can
     * be specialized by subclasses to perform tool-specfic forms of search,
     * such as by object name instead of raw text string.
     */

    virtual void Command(String* s);
    /*
     * Execute the given command string in the context of the current canvas.
     */


  protected:
                                  // Data represenation forthcoming
};

#endif
```