Hash Table and Generic ADT

Project 3

Generic ADT (2nd week)

- Change directory to “generic”
- Create table.c to implement the set operations with generic data type

```
struct set {
        int count;
        int length;
        void **data;
        char *flags;
        int (*compare)();
        unsigned (*hash)();
        };
```

```
SET *createSet(int maxElts, int (*compare)(), unsigned (*hash)());

static int search(SET *sp, void *elt, bool *found)

void addElement(SET *sp, void *elt); (does not allocate new memory)

void removeElement(SET *sp, void *elt);

void *findElement(SET *sp, void *elt);

void *getElements(SET *sp);

(*sp->compare)(elt1, elt2)
(*sp->hash)(elt)
```

Assert

```
sp = malloc(sizeof(SET));
assert(sp != NULL);
```

If malloc failed and sp is NULL, nothing we can do about it. Program terminated and error message prompt.

unique.c

```
->addElement(sp,word)                                    table.c
    ->assert(sp != NULL && word != NULL)
```

If “sp” or “word” is NULL, “table.c” can do nothing about it.

Assert vs. if...else...

- If...else…: when we need to handle different conditions.
- Assert: when we DON’T need to handle different conditions.

Submission

- Submission deadline:
  - Sunday, May 2nd at 11:59 pm
  - 10% off every 24 hrs after deadline
  - No submission will be accepted after Wed, May 5th at 11:59 pm
- Demo deadline:
  - Lab section next week
  - No demo will be accepted in TA's office hours after your lab section next week
- File:
  - Both tar file or zip file will be accepted
    - tar -czvf project3.tar folder_path
    - folder_path is the directory of the folder that contains both "strings" folder and "generic" folder