

COS 217 Midterm Exam

Princeton University, Spring 2007

March 14, 2007

This exam is **closed book, closed notes**. You have **50 minutes**.

When writing program code, don't bother with comments.

Your name:

Princeton login ID:

Precept (circle one):

Mon/Weds

Tues/Thurs

Write and sign the honor pledge:

1) Write an “acronym server” program. This program reads from the file phrases data like this,

```
Computer Science
Association for Computing Machinery
United Parcel Service
Miles Per Gallon
Uninterruptible Power Supply
Advanced Cruise Missile
```

Then it reads from the standard input queries like this

```
ACM
MPG
UPS
```

For each query, it prints out a list of the phrases that match that acronym; in this example the output would be,

```
Association for Computing Machinery } these 2 might have come out in either order
Advanced Cruise Missile             }
Miles Per Gallon
United Parcel Service               } these 2 might have come out in either order
Uninterruptible Power Supply       }
```

The acronym is made from the capital letters of the phrase; doesn't matter whether they're at the beginnings of words or not.

Guidelines and hints:

- Write your program on the next 3 pages. I have provided a skeleton solution; you fill it in.
- For fast lookup of acronyms, make the acronym be the key of your symbol table, and let the binding be a list of phrases.
- You might start by writing down the data-structure declaration for a linked list of phrases.
- Remember that `fgets` leaves the trailing newline character at the end of the text that it reads in.
- It might be useful to consider what happens when you apply your `extract_acronym` to the string "UPS\n".

```
/* Includes and data structure definitions */

#include "syntable.h"

char *extract_acronym(char *phrase) {

    /* count how many capital letters are in the phrase */

    /* make the acronym out of just the capital letters */

}
```

```
void add_a_phrase(Symtable_T table, char *phrase) {
```

```
}
```

```
void do_a_query(Symtable_T table, char *acronym) {
```

```
}
```

```
enum {MAXLINE = 1000};
```

```
int main(int argc, char **argv) {  
    Symtable_T table = Symtable_new();  
    char line[MAXLINE];  
    FILE *phrases = fopen("phrases","r");  
    assert (phrases != NULL);  
    while (fgets(line, MAXLINE, phrases))  
        add_a_phrase(table, line);  
    while (fgets(line, MAXLINE, stdin))  
        do_a_query(table, line);  
    /* we will not worry about freeing things */  
    return 0;  
}
```