# IS12 - Introduction to Programming
# Lecture 20: Nested Loops

Peter Brusilovsky
http://www2.sis.pitt.edu/~peterb/0012-051/

## Outline

- Nested Loops
- Example: Star chart
- Example: String to int and getline
- Example: Longest line
- Example: Simple pattern search

# Example: Star Chart (1)

```
#include <stdio.h>
void main () {
   int c, i, j;
   int ndigs[10];

   /* initialize */
   for(i = 0; i < 10; ++i) ndigs[i] = 0;

   /* count */
   while ((c = getchar()) != EOF)
       if (c >= '0' && c <= '9')
             ++ndigs[c - '0'];
```

# Example: Star Chart (2)

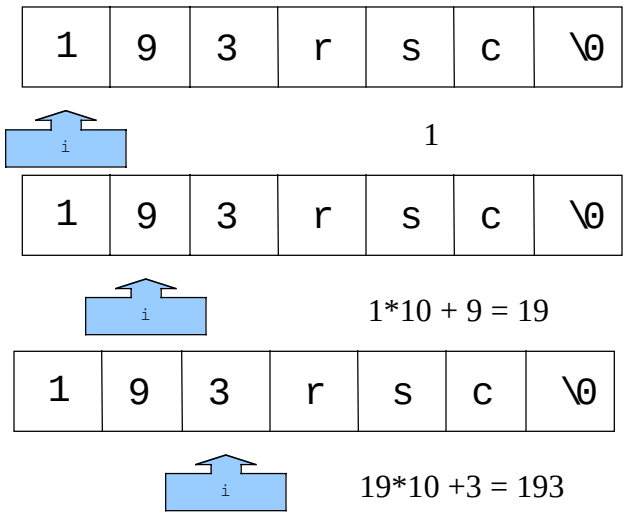
```
   /* reporting */
   printf("\n   ^\n");
   for (i = 0; i < 10; ++i) {
       printf(" %d |", i);
       for(j=1; j <= ndigs[i]; ++j)
             printf("*");
       printf("\n");
   }
   printf("   +----------------->\n");
}
```

nested loop

# Example: String to int (driver)

```
/*  Example: Convert String to Integer
   Source: K&R2, p43 (see also better version on p.61)
   Driver: Peter Brusilovsky  */

#include <stdio.h>
#define BUFSIZE 20 /* maximun input line size */

int atoi(char []); /* converts string to int */
int getline(char [], int); /* reads string from input */

void main() {
   char buf[BUFSIZE];

   if (getline(buf, BUFSIZE) > 0)
        printf("You have entered string %s and it was
   converted to %d\n", buf, atoi(buf));
}
```

# Example: String to int



## Example: String to int (atoi)

```
/* atoi: converts s to integer */

int atoi(char s[])
{
  int i, n;

  n = 0;
  for (i = 0; s[i] >= '0' && s[i] <= '9'; ++i)
      n = 10 * n + (s[i] - '0');
  return n;
}
```

## Example: getline

```
/* getline: read a line into s,  return length */
int getline(char s[], int lim) {
  int c, i;
  for (i = 0; i < lim - 1 && (c = getchar()) != EOF
            && c != '\n'; ++i)
      s[i] = c;
  if (c == '\n') {
      s[i] = c;
      ++i;
  }
  s[i] = '\0';
  return i;
}
```

# Example: Longest Line

```
/* print longest input line */
main() {
   int len; /* current line length */
   int max; /* maximum length seen so far */
   char line[MAXLINE]; /* current input line */
   char longest [MAXLINE]; /*longest line saved here */
   max = 0;
   while ((len = getline(line, MAXLINE)) > 0)
        if (len > max) {
               max = len;
               copy(longest, line);
        }
   if (max > 0) /* there was a real line */
        printf("%s", longest);
   return 0;
}
```

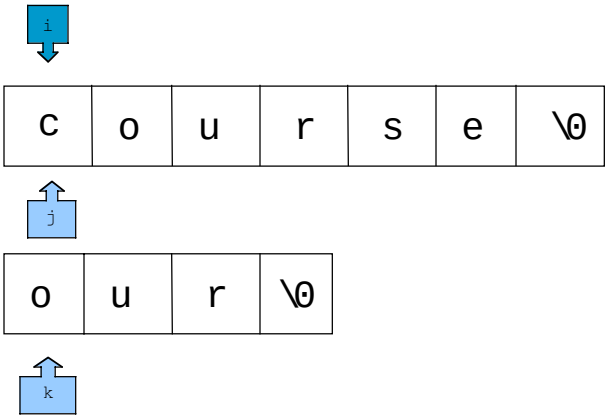
# Example Longest Line: copy

```
/* copy: copy 'from' into 'to'; assume to is big enough
   */
void copy(char to[], char from[]) {
   int i = 0;
   while((to[i] = from [i]) != '\0')
        ++i;
}
```

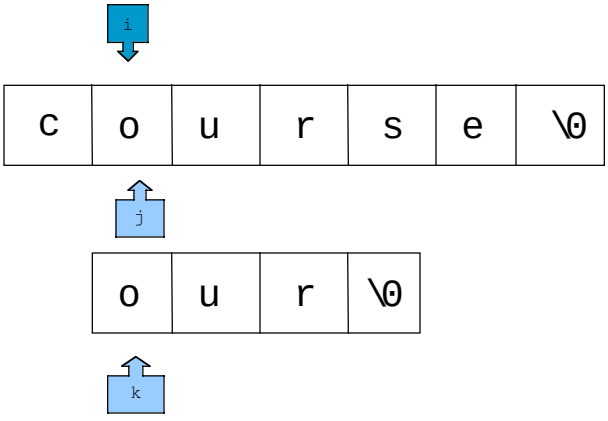
---

```
void copy(char to[], char from[]) {
   int i;
   for(i=0; (to[i] = from [i]) != '\0'; ++i)
        ;
}
```

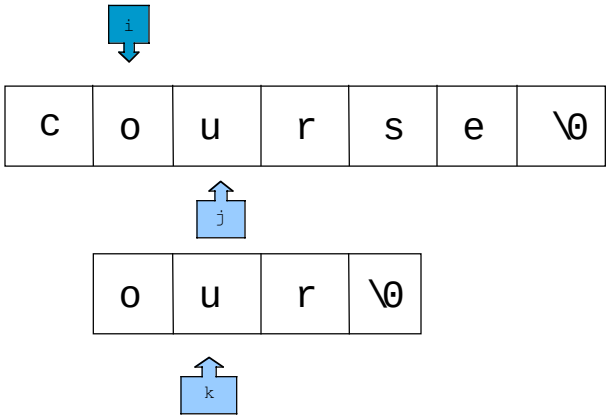
## Example: String Search



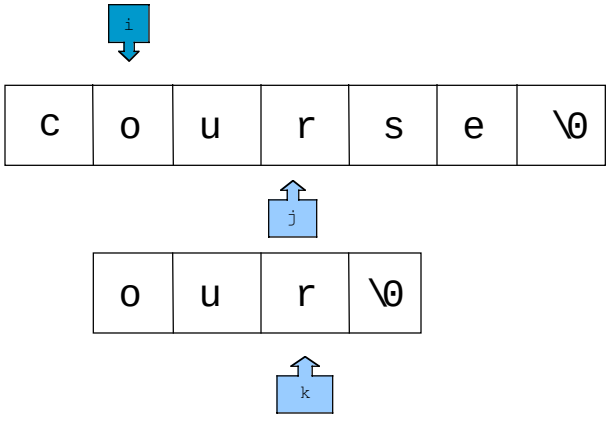

## Example: String Search



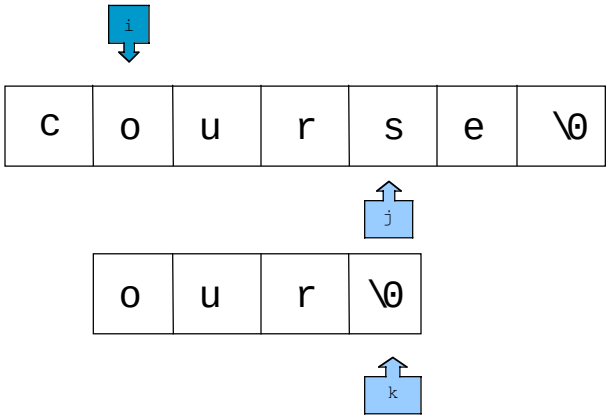
# Example: String Search



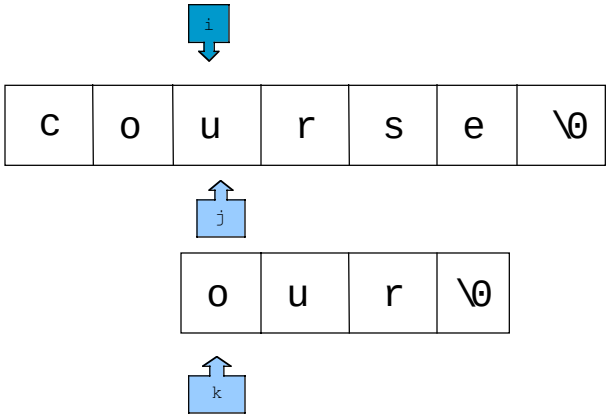

# Example: String Search



# Example: String Search




# Example: String Search

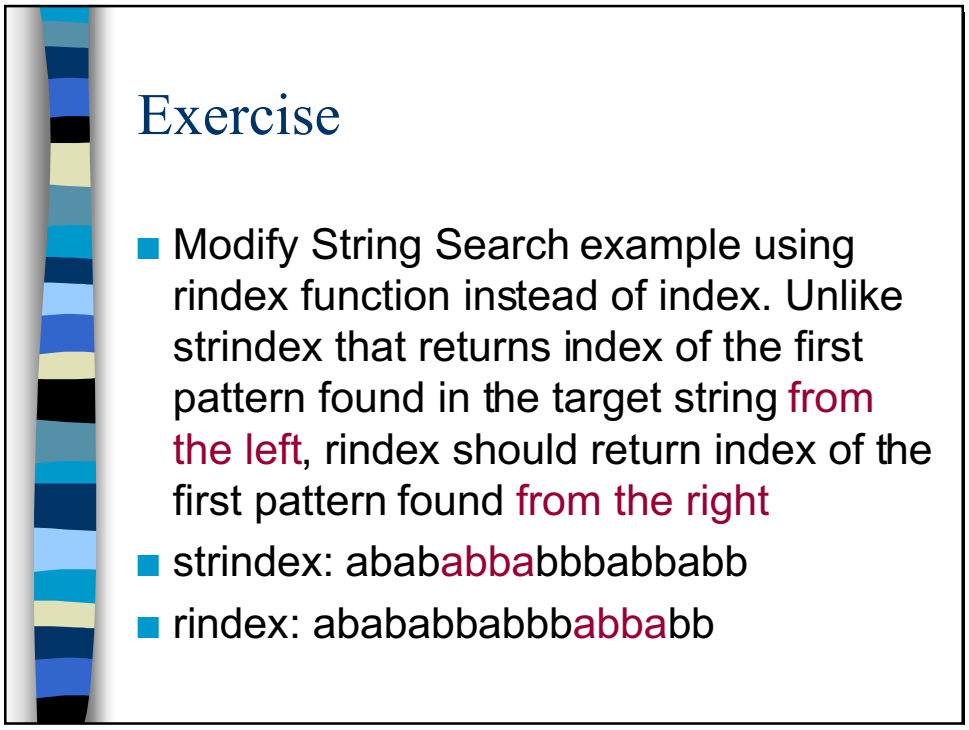


# Example: String Search (1)

```
int getline(char line[], int maxline);
int strindex(char source[], char searchfor[]);
char pattern[] = "ould";
/* find all lines matching pattern */
void main()
{
   char line[MAXLINE]; /* current input line */
   int found = 0;

   while (getline(line, MAXLINE) > 0)
        if (strindex(line, pattern) >= 0) {
               printf("%s", line);
               found++;
        }
}
```

# Example: String Search (2)

```
/* strindex: return index of t in s, -1 if none */
int strindex(char s[], char t[])
{
   int i, j, k;

   for(i = 0; s[i] != '\0'; i++) {
        for(j = i, k = 0; t[k]!='\0' && s[j]==t[k];
                              j++, k++)
               ;
        if (k > 0 && t[k] == '\0')
               return i;
   }
   return -1;
}
```


Exercise
Modify String Search example using rindex function instead of index. Unlike strindex that returns index of the first pattern found in the target string from the left, rindex should return index of the first pattern found from the right
strindex: ababbabbbbabbabb
rindex: abababbabbbabbabb