

Announcements

Thursday Extras: CS Commons on Thursdays @ 4:00 pm
this week: ????

Office hours: normal this week

Monday: 1:00-3:00 pm

Tuesday, Thursday: 2:30-4:00 pm

Wednesday, Friday: 1:30-3:00 pm

Notes

- When handing work in at my office:
 - if door closed, put work under the door
 - if door open and I'm there, hand it to me
 - If I'm talking to someone in the office, stay in the hall, and wave at me with the work—I'll take in and continue with the person in the office
 - If door open and I'm not there, put work on my chair or on the floor by my office (not on a pile elsewhere)
- If issues with an assignment, talk to me early and often
 - If many issues, deadlines might be adjusted
 - If no prior discussion or only at last minute, deadlines are firm

A user is to type words on each of 2 lines.

The program is to read the first word of each line, and ignore the rest of the input.

Assume each word is no longer than 9 characters, and assume the following declarations:

```
char str1 [10], str2 [10], bigstr[100], ch;
```

Two approaches are proposed:

1. `scanf ("%9s", str1);`
 `fgets (bigstr, 100, stdin);`
 `scanf ("%9s", str2);`
2. `scanf ("%9s", str1);`
 `while ((ch=getchar()) != '\n') ;`
 `scanf ("%9s", str2);`

Which, if any, of these approaches works?

A. Both 1 and 2

B. 1, not 2

C. 2, not 1

D. neither 1 nor 2

As in the previous slide, a user is to type a word on each of 2 lines. Two more approaches are proposed:

```
3.  int i = 0;
    do { str1[i] = getchar();
        i++;
    }
    while (!isspace(str1[i]))
    str1[i] = 0;
    while ((ch=getchar()) != '\n') ;
    int i = 0;
    do { str2[i] = getchar();
        i++;
    }
    while (!isspace(str2[i]))
    str2[i] = 0;
```

```
4.  int i = 0;
    do { str1[i] = getchar();
        i++;
    }
    while (!isspace(str1[i-1]))
    while ((ch=getchar()) != '\n') ;
    int i = 0;
    do { str2[i] = getchar();
        i++;
    }
    while (!isspace(str2[i-1]))
```

- A. Both 3 and 4
- B. 3, not 4

- C. 4, not 3
- D. neither 3 nor 4

Suppose input will contain a name in the first 40 characters of a line, followed by a number.

```
char line [101]; //leaving room for null character
char name [41];
int number;
```

Two approaches are proposed reading a line

1. `scanf ("%41s %d", name, &number);`
2. `fgets (line, 101, stdin);`
 `strncpy (name, line, 40);`
 `name[40] = 0;`
 `number = atoi (line+40);`

Which of these approaches works reliably?

- A. Approaches 1 and 2
- B. Approach 1, not 2
- C. Approach 2, not 1
- D. Neither 1 nor 2

Suppose you want to read an integer at the start of one line, skip anything else on that line, and then read a double at the start of the next line.

```
int one;  
double two;  
#define max 100  
char line [max+1];
```

Two approaches are proposed:

1. `scanf ("%d", &one);`
 `while (getchar () != '\n') ;`
 `scanf ("%lf", &two);`
2. `fgets (line, max+1, stdin);`
 `one = atoi (line);`
 `while (line[strlen(line)-1] != '\n')`
 `fgets (line, max+1, stdin);`
 `fgets (line, max+1, stdin);`
 `two = atof (line);`

Which of these approaches works reliably?

- A. Approaches 1 and 2
- B. Approach 1, not 2
- C. Approach 2, not 1
- D. Neither 1 nor 2