

Announcements

Announcements (Extra credit for any of these)

Rosenfield Symposium: Tyranny of Data, this Wed-Thurs

Thursday Extra: Thursdays@4:00 in Science 3821

This week: Nathaniel Williams '20: Internship at Vocdoni

Class slides posted weekly (link on syllabus)

Quiz results: variations in versions a and b!

Web server: for now: www-temp.cs.grinnell.edu/~walker

MyroC documentation on Home Page link

New lab partners for this week (stay tuned)

Today's outline

- Working with partners
- Procedures
- Using procedures to structure solutions
- Incremental software development

Working with Partners

Driver, Navigator

Listening, anticipating

Change person at keyboard at least every class!

Constructive tone!!!!

Write on a card (and give to me)

- what worked well for you in CSC 151
- what could have gone better (and what do you suggest as an approach to help resolve the troubles)

I'll put together your responses (anonymously) for next class

Consider the following program: What is printed?

```
#include <stdio.h>

void procA ()
{
    printf ("A\n");
}

void procB ()
{
    printf ("B\n");
    printf ("    B1\n");
    printf ("    B2\n");
}

void procC ()
{
    printf ("C\n");
}

int main ()
{
    procA ();
    procB ();
    procA ();
    return 0;
}
```

- A. A
B
B1
B2
C
- B. A
B
B1
B2
A
- C. A B B1 B2 C
- D. A B B1 B2 A
- E. none of the above

3 Widely observed problem-solving tips:

1. Hours of coding can save minutes of planning
2. Write comments first (to clarify your thinking)
3. You don't have to develop a large, complex program all at once!
 - Start with a small piece that has steps identified (perhaps procedures as stubs)
 - Add elements, a few small pieces at a time

Song Project (Due Friday at start of class)

- Choose a song, at least 30 seconds in duration
 - May be from another source or of your own making
 - Creativity encouraged (4 min, 33 sec silence already done by J. Cage)
- Note: rBeep/Scribbler 2 hangs if duration > 3.0014 seconds
- Introduction
 - give song, composer in comments and eSpeak
 - if song found elsewhere, full citation essential
- Structure (indicated by formatting, comments, etc.)
 - All complete programs for course require Honesty Certification (handwritten signatures)
 - divide song into at least 5 functions (in addition to main)
 - each function
 - introduced by eSpeak
 - header with comments on function doing (e.g., line 2 of song)
 - MyroC header file provides examples of header format
- Submission (due before class—don't print after class begins!)
 - on submission per group
 - separate document about correctness (e.g., length, what heard)
 - email to csc161-01-grader@grinnell.edu
 - email subject 161.01, Project 000, Partner names
 - printed copy with both code and correctness statement