Tutorial Sheet (August 10, 2018)

ESC101 – Fundamentals of Computing

Announcements

1. **Two extra sessions on Saturday, August 11, 2018**
    a. **5PM-6PM**: Extra session for students getting introduced to computer usage. No lecture material will be covered.
    b. **6PM-7PM:** Extra session for students facing difficulty with English lectures. Only for students who are fluent with Hindi. Revise lecture slides, practice questions before coming.

    Venue for both sessions, NCL CC-02.
2. Students are advised not to submit minor quiz questions more than once. Doing so results in a straight zero score. The form allows you to go back to change your answers. In case of difficulty with form, contact TA/tutor/inst. **DO NOT RESUBMIT**.
3. Students are advised to use relproxy when attending lab. This ensures a hassle free experience with the minor quiz.
4. Students are instructed to bring their IITK ID cards when attending labs as lab attendance will be taken using IITK ID cards. Some students don't do so and this causes delays.
5. Piazza accounts have been created for all students. Students are requested to log on to Piazza and start discussing. Students without a Piazza account may mail the instructor.

Demonstrations

1. Please demonstrate the use of simple but timesaving and very useful keyboard shortcuts on Prutor.

a. CTRL+S: save
b. CTRL+C: copy
c. CTRL+V: paste
d. CTRL+Z: undo
e. CTRL+Y: redo
f. CTRL+SHIFT+C: compile
g. CTRL+SHIFT+X: execute
h. CTRL+SHIFT+V: evaluate

2. Please demonstrate the proxy setting procedure once more: `relproxy.iitk.ac.in: 3128` with `noproxy` for `*.iitk.ac.in`

Revision (ask for doubts)

1. Distinction between declaration and initialization of variables.
2. Commenting code is useful and important too. Comment styles.
3. scanf for int, long, float. Multiple inputs via scanf and Mr C's habit of skipping multiple whitespaces while reading ints/floats/longs.
4. Legal identifier names and best practices while giving names.
5. Mixed type operations (int, long, float). int a; long b; float c; Mr C automatically typecasts upwards when operands mixed.
   I. b = a + a will cause integer addition as both operands int
   II. b = long(a) + a will cause long addition as one is long
   III. b = a + 50000000000 will cause long addition as 5e10 is long
   IV. c = 2 / 3 will cause integer division and result will be c = 0.0
   V. c = 2 / 3.0 or c = 2 / (float) 3 will cause float division since at least one operand is float

---

Sample Question to discuss

Given: two integers between -1,000,000 and +1,000,000. Read them into two integer variables and print their product. Be careful about exceeding the range of integers. The integers will be presented in the input as

\\integer1||integer2//

```
#include<stdio.h>
int main(){
    int a ,b;
    long prod;
    scanf("\\\\%d||%d//", &a, &b);
    prod = (long)a * (long)b;
    printf("%ld", prod);
    return 0;
}
```

Some Pitfalls and recognizing compiler error messages

1. Multiple declarations, simultaneous declarations allowed but mixed type declarations forbidden e.g. int a, float b; Mixed type initializations allowed though e.g. int a; float b; a = 1, b = 2.2;
2. Printf can handle without any need for escape commands,
    a. Latin letters: upper and lower case
    b. Arabic numerals: 0 – 9
    c. Arithmetic operators: + - / * = < >
    d. Parentheses: ()[]{}
    e. Punctuation marks: space ~`!@#$^&_|?/,.

However, " \ % newline, tab require escape commands. Reason: tab, newline are whitespace characters requiring special handling, whereas " \ % are themselves have special roles inside the format string so they themselves have to be escaped.

3. /* */ comments do not nest i.e. /* /* */ */ will cause error
4. count = count + 1 will take count, add one to its value, and put the result back into count itself, overwriting the old value.
5. Importance of bracketing (a+b)/2 vs a+b/2
