ECS 30 **Homework #2 (35 points)** **Winter 2013**

Due: Wednesday, January 23rd. Written: 4pm in 2131 Kemper. Programs: 11:55pm using handin to cs30, p2 directory. Filenames: graphics.c, clock.c, grades.c, loan.c.

Written (5 points): p. 167: 1, 2, 3, 4, 5.

1. Define top-down design and structure charts.
2. What is a function prototype?
3. When is a function executed, and where should a function prototype and function definition appear in a source program?
4. What are three advantages of using functions?
5. Is the use of functions a more efficient use of the programmer's time r the computer's time? Explain your answer.

Zyante: 2.3-2.7, 2.9, 2.12, 2.14. 2.15, 6.1-6.4, 6.6, 6.7.

Programming (30 points)

All programs should be able to compile with no warnings when compiled with the –Wall option, e.g. gcc –Wall track.c. You should put your name(s) in a comment on the first line of each file. Since main() should be the first function in each file, you will need to provide a prototype above main() for each function you write. All main() functions must contain nothing but variable declarations (which may be initialized when declared), calls to your functions, and a return statement. You are not to use loops, arrays, nor selection statements for any program. You are never allowed to use global variables in this course.

You will find my executables in ~ssdavis/30/p2 in the CSIF. The prompts, and output format of each program must match the examples exactly. To ensure that the actual values calculated by the programs match the examples, you should use doubles for all real variables, and ints for all whole numbers. User inputs are in **bold**.

#1. Filename: graphics.c (5 points) (7 minutes) Write a program that produces the image below by calling five functions from main: rectangle(), head(), triangle(), lines(), and blank_lines(). All printfs() of the program must be only within those four functions. Note that to print a '\', you need to use an extra '\', i.e. "\\" because '\' is used to signal a special character.

```
[ssdavis@lect1 p2]$ graphics.out
 /---\
| 0 0 |
 \_-_/
  /\
 /  \
/____\
  /\
 /  \
/    \


 /---\
| 0 0 |
 \_-_/
|----|
|    |
|----|
  /\
 /  \
/    \
```

Continued on right column.

```
  /\
 /  \
/____\
|----|
|    |
|----|
|----|
|    |
|----|
  /\
 /  \
/    \
[ssdavis@lect1 p2]$
```

#2. Filename: clock.c (5 points) (5 minutes)

Write a program that reads two times, and then prints out the difference between them in clock format, i.e. with a colon between the hours and minutes. You will have two functions, get_time(), and show_difference(). get_time() will prompt and read a time in typical clock format. show_difference() will receive two times as minutes and hours, and print their difference. You will find abs() from stdlib.h useful. To use a zero instead of a space for filling needed space in a formatted number use %0…d, e.g. "%08d" would precede the integer with 0's as needed to have the number end at the eighth position.

```
[ssdavis@lect1 p2]$ clock.out
Please enter time #1 in clock format: 12:10
Please enter time #2 in clock format: 9:05
The difference is 3:05.
[ssdavis@lect1 p2]$ clock.out
Please enter time #1 in clock format: 9:08
Please enter time #2 in clock format: 11:55
The difference is -2:47.
[ssdavis@lect1 p2]$
```

#3. Filename: grades.c (8 points)(15 minutes)

We want to tally the percentage of each letter grade given in four ECS courses. You are to write a program that asks for the grades from the four classes, and then prints out the percentages of each letter grade. You must have a function get_grades() that prompts and reads the grades for one course. To pass the name of a course you will use string constants, e.g. "ECS 30", with the associated parameter being a `const char *course.` printf() uses "`%s`" to indicate a string source. You will also have a show_totals() function that prints the final percentages of all the grades.

```
[ssdavis@lect1 p2]$ grades.out
Please enter the grades for ECS 10 in order As Bs Cs Ds Fs: 20 30 30 10 4
Please enter the grades for ECS 20 in order As Bs Cs Ds Fs: 15 25 20 8 5
Please enter the grades for ECS 30 in order As Bs Cs Ds Fs: 100 120 110 40 25
Please enter the grades for ECS 40 in order As Bs Cs Ds Fs: 40 45 35 38 22
The percentages were A: 23%, B: 29%, C: 26%, D: 12%, F: 7%.
[ssdavis@lect1 p2]$
```

#4. Filename: loan.c (12 points) (20 minutes)

Many students must take loans to pay for college. Write a program that determines the final cost of such a loan. The user will enter the original loan amount, the number of months they wish to take to repay the loan, the interest rate on the loan, and the number of months before the payments will start. The program will then provide the amount of the monthly payments necessary to pay off the loan, and the total of all the payments (assuming that even the last payment is the same as the others). Your program should have three functions, get_loan(), calc_payments(), and show_totals(). get_loan() will prompt and read all of the necessary information from the user. calc_payments() will take the information, and return the amount of the monthly payments as well as the total of all the payments. show_totals () will print the size of the monthly of payments and total cost. Due to rounding errors, your answers are allowed to differ from mine by a few pennies. You will need to use the exp(), log(), and pow() functions from math.h. To use functions from math.h, you must have –lm on your compile line (that is an "l" as in library) to link with the math library, e.g., gcc –Wall –lm loan.c.

The germane formulas are:

$A = P(1 + r/n)^{nt}$, where A = final principal, P = original principal, r = annual interest rate, n = number of times interest is compound in year, and t is the number of years. This formula is useful for the college years when no payments are made.

$m = \frac{Ai}{1-e^{-nln(1+i)}}$, where m = monthly payment, A = principal at start of payments, n = number of payments, and i = monthly interest rate which is the annual interest rate divided by 12.

```
[ssdavis@lect1 p2]$ loan.out
Please enter the loan, and interest: 20000 0.0380
Please enter the number of months until you start payments: 42
Please enter the number of payments you wish to make: 60
For that loan, your monthly payment will be $372.37, for a total of $22342.10.
[ssdavis@lect1 p2]$
```