85/100

CSC 111 Fall 2011
Midterm 1

Your Name [_(handwritten signature)_]   **UVicID** [V 007_____]

Instructions

- This midterm consists of 3 pages and 13 questions.
- The first 6 questions are worth 5 points each for a total of 30 points. The last 7 questions are worth 10 points each for a total of 70 points. The complete midterm is worth 100 points.
- You have 70 minutes for this midterm. Time management—approximately 5 minutes per question.
- This is a closed books, closed notes, no gadgets, and no electronic devices midterm.
- Turn in your completed midterm at the front of the class.
- Leave through the front door on your left.

1. Who invented the C programming language? Check the correct circle.

   0

   ○ Niklaus Wirth

   ✓ James Gosling

   ○ Brian Kernighan

   ✓ Dennis Ritchie

2. Consider the following syntactically correct C program.

   ```c
   #include <stdio.h>
   int main(void) {
       printf("Practice is the best of all instructors\n");
       return 0;
   }
   ```

   How many function names appear in this program? Check the correct circle.

   5

   ○ 0

   ○ 1

   ● 2

   ○ 3

5

3. What are the values of the following C expressions?
   Assume the following C declarations and initializations:

   ```
   int x = 3;
   int a;
   ```

   2

   a += 2*(a = 19) → 57 ✓

   (0<=x && x<=9) → 3? ✗

   (4*7)/2 → 21 ✗

   (7/2)*4 → 14 ✗

4. For the following statements, check the correct circle.
   The C preprocessor

   0

   - ☑ compiles C programs
   - ☑✓ includes text files using #include and substitutes text using #define directives
   - ○ builds an application or an executable
   - ○ checks for semantic errors

5. How many proper identifiers are in each line according to C syntax? Enter the number for each line in the circle at the front of the line.

   3

   - ⑤ ABC aBc _sysRoutine B777 k 7⁄11 int mega %d
   - ⓪ ✗ This/* is a */an identifier with a caveat
   - ⑤ 17.4 nine 19.6 forty-four hexadecimal k modulo output
   - ⑥ ✓ Check for C++ comments and keywords such as for while long

S

6. What is the basic structure of a function in C? Check the correct circle.

○ ✓
```
void name ("declarations"){
    "parameters"
    "statements"
}
```

○
```
void name("parameters")[
    "declarations"
    "statements"
]/*name*/
```

○ ✓
```
void name("parameters"){
    "declarations"
    "statements"
}/*name*/
```

○
```
void name("statements")
{
    "declarations"
    "parameters"
}/*name*/
```

6

7. What is the output of the following syntactically correct C program?

```
#include <stdio.h>
int main(void) {
    int k = 29;
    while (k > 7) {
        printf("%d", k);
        k = k - 3;
    }
    printf("\n");
    return 0;
} /* main */
```

29   26   23   20   17

14   11   8

Output: | 29   26   23   20   17   14   11   8 |

10

3

15

8. What is the output of the following syntactically correct C program?

```
#include <stdio.h>
int main(void) {
    int k = 5;
    while (k < 15) {
        if (k%2 == 0) printf("%d", k);
        k = k + 1;
    }
    printf("\n");
    printf("\n");
    return 0;
} /* main */
```

5

5

Output: ⑤ 6 ⑦ 8 ⑨ 10 ⑪ 12 ⑬ 14

9. What is the output of the following syntactically correct C program?

```
#include <stdio.h>
int main(void) {
    int k = 29;
    while (k > 31) {
        printf("%d", k);
        k = k - 3;
    }
    printf("none\n");
    return 0;
} /* main */
```

10

Output: none

10. What is the output of the following syntactically correct C program?

```
#include <stdio.h>
int main(void) {
    int k = 5;
    while (k < 12) {
        printf("%d", k);
        k = k + 1;
    }
    printf("\n");
    return 0;
} /* main */
```

5 6 7 8 9 10 11

10

Output: 5 6 7 8 9 10 11

4

25

11. Complete the control flow diagram or call graph for the program below. As the legend indicates rectangles denote functions and arrows denote call dependencies. The diagram already contains boxes for the 11 functions involved in the program. Thus, you only need to add arrows to the diagram.

```c
#include <stdio.h>
void f1a (void) {
    printf ("f1a\n");
}
void f1b (void) {
    printf ("f1b\n");
}
void f2a (void) {
    printf ("f2a\n");
}
void f2b1 (void) {
    printf ("f2b1\n");
}
void f2b2 (void) {
    printf ("f2b2\n");
}
void f2b (void) {
    f2b1 ();
    f2b2 ();
}
void f1 (void) {
    f1a ();
    f1b ();
}
void f2 (void) {
    f2a ();
    f2b ();
}
void f3 (void) {
    printf ("f3\n");
}
int main (void) {
    f1 ();
    f2 ();
    f3 ();
    return 0 ;
}
```


Legend

function

function call
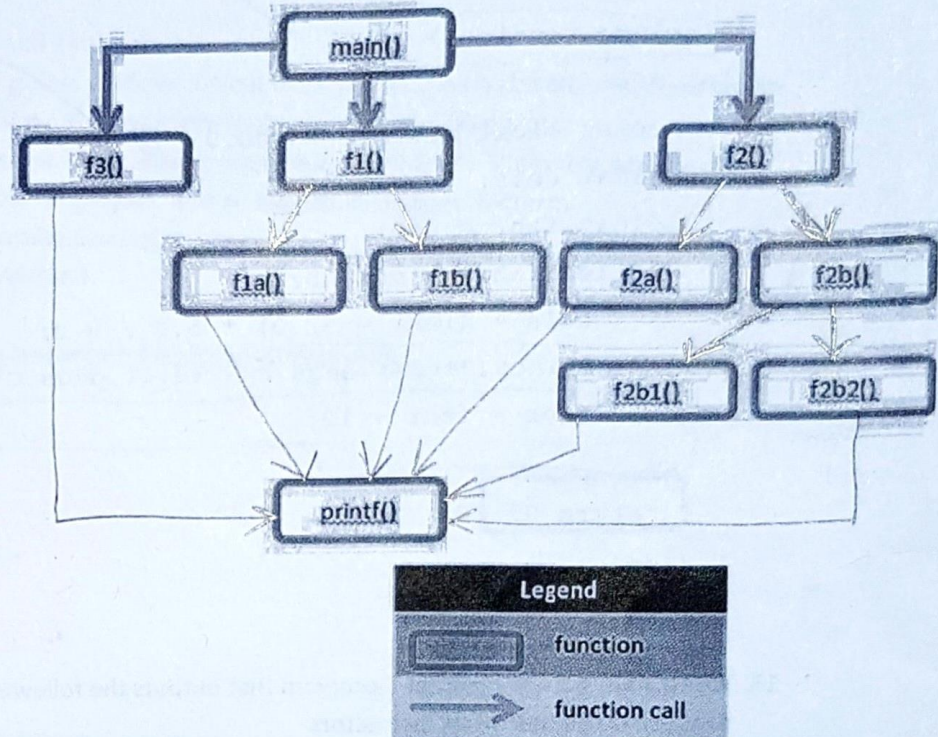
12. Consider the following syntactically correct C program. The return statement is identified using box
    and labeled with a). Draw boxes and label the boxes with b), c), d) and e) to:
    a) identify the return statement
    b) identify the names of any read-only variables
    c) identify entire while loops
    d) identify function calls
    e) identify preprocessor directives

```c
#include <stdio.h>   e)  ✓
int main(void) {
    const float maxfahr = 100.0;   b)  ✓
    float cels;
    float fahr = 0.0;
    while (fahr <= maxfahr) {
        cels = (fahr - 32.0) * 5.0 / 9.0;
        printf("%6.1f degs F = %6.1f degs C\n", fahr, cels);   ✓
        fahr = fahr + 10;
    }
    return 0;   a)
} /*main*/
```

c) ✓

10

13. Write a syntactically correct C program that outputs the following proverb:
    Practice is the best of all instructors

```c
# include <stdio.h>
int main () {
    printf("Practice is the best of all instructors\n");
    return 0;
}
```

1
2
3
3
1

3

10

20