ECE 15    Fall 15

Midterm

This is a closed-book exam: no notes, books, calculators, cellphones, or friends are allowed.
In problems 2–4, you can assume that the user’s input is correct.
If you need more space, please use the back of the page.
You have 80 minutes.

Good luck!

____________________________________________________________________

Name: ____________________________

PID: A___________________________

____________________________________________________________________

<table>
<thead>
<tr>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ________/40</td>
</tr>
<tr>
<td>2. ________/20</td>
</tr>
<tr>
<td>3. ________/20</td>
</tr>
<tr>
<td>4. ________/20</td>
</tr>
<tr>
<td>= ________/100</td>
</tr>
</tbody>
</table>
Problem 1 (40 points)

For each of the following programs, indicate whether or not it has errors. If there are errors, mark all of them (there may be more than one) on the program and briefly describe the problems. If there are no errors, write the program’s output. Hint: two programs contain errors.

(1) #include <stdio.h>
int main() {
    int a = 2;
    printf("%d=2 is %s", a, a == 2 ? "T" : "F");
    return 0;
}

No Error | Error

Output: _______ _______ _______ _______ _______ _______ _______ _______

(2) #include <stdio.h>
define a 1.464
int main() {
    printf("%.5f", 10+a);
    return 0;
}

No Error | Error

Output: _______ _______ _______ _______ _______ _______ _______

(3) #include <stdio.h>
int main() {
    int x = 2;
    int y = 1;
    int z;
    z = (x<y) ? 1 : 0;
    if(z)
        printf ("%d", ++x+y);
    else
        printf ("%d", ++y-x);
    return 0;
}

No Error | Error

Output: _______ _______ _______ _______ _______ _______ _______

(4) #include <stdio.h>
int main() {
    int a = 3;
    int b = 2;
    double z = 2.3;
    char c = ‘A’ - ‘a’ + ‘b’;
    int v, r;
    v = a > z > b;
    r = (c == ‘B’) && ((int)z == 2);
    printf("v=%d,r=%d", v, r);
    return 0;
}

No Error | Error

Output: _______ _______ _______ _______ _______ _______ _______ _______
(5) #include <stdio.h>
int main() {
    double number = 10.5;
    i = 1;
    while (i < 5) {
        number *= 2;
        printf("%d,",number);
        i++;
    }
    return 0;
}

(6) #include <stdio.h>
int main() {
    int value = 0;
    for (int i = 0; i<5; i++) {
        value += i % 2;
    }
    printf("%d", value);
    return 0;
}

(7) #include <stdio.h>
int main() {
    char c = 'A';
    int rem;
    for (int i = 0; i<7; ++i) {
        rem = i % 3;
        switch(rem) {
            case 0: printf("%c", c+1);
            break;
            case 1: printf("%c", c+2);
            break;
            default: printf("%c", c);
        }
    }
    return 0;
}

(8) #include<stdio.h>
int main() {
    int i = -1, j = 1, k = 0, m, n;
    m = ++i || j--;
    n = j-- && ++k;
    printf("%d,%d", m, n);
    return 0;
}
```c
#include <stdio.h>
int main() {
    int limit = 50;
    for (int i = 0; i <= limit; i+=5) {
        if (i == 30)
            break;
        if (i % 10 == 0)
            continue;
        printf("%d ",i);
    }
    return 0;
}
```

Output: ____________

```
#include <stdio.h>
int main() {
    double x = 1;
    for (int i = 5, i> 0, i-- ) {
        x /= (i-3);
    }
    printf("%lf", x);
    return 0;
}
```

Output: ____________
Problem 2 (20 points)

Write a program that asks the user for 15 characters (as usual, followed by a new line), and counts the number of identical consecutive character pairs entered, namely, the number of characters followed by the same character. For example, the word “bookkeeping” has three identical consecutive pairs (o, k, and e), and “skillless” also has three pairs (two l’s and s). Characters can be letters, digits, punctuation marks, spaces, etc.

Describe your rough approach in a few lines:

(-)$ a.out
15 characters: I_love_the_food
1
(-)$ a.out
15 characters: amaaaaaaaaaaaazinggg!!
7
(-)$ a.out
15 characters: Is_this_okay??
2
(-)$
Problem 3 (20 points)

Write a program that asks the user for a positive integer and prints the alternating sum of its digits. The alternating sum is calculated by starting with the rightmost digit, subtracting the second rightmost digit, adding the third, etc. For example, the alternating sum of 3625 is $5 - 2 + 6 - 3 = 6$.

Describe your rough approach in a few lines:
Problem 4 (20 points)

We are throwing a post-midterm party and ordering 2 KFC grilled-chicken pieces per guest. Chicken pieces come in packs of 2, 6, 12, and 24, and to save for our real post-final celebration, we would like to order the largest packs possible. Write a program that takes the number of guests and outputs the number of packs of each size. Note that you first need to get the largest number of 24-packs, then the largest number of 12-packs, etc., and recall that each guest consumes two pieces.

Describe your rough approach in a few lines: