

CS 60
FIRST QUIZ
July 5, 2007

WRITE ALL YOUR ANSWERS ON SPACE PROVIDED.
ANSWER ALL FOUR QUESTIONS. TOTAL POINTS IS 33.

NAME: _____

1 { Circle for each part True or False depending whether or not the statement is true or false. Each question is worth 1.5 points }

- { **True** or False } The line `int xx;` inside the main function declares variable `xx` as an auto variable.
- { True or **False** } The for statement `for(i = 0; i < 0; i++)` is an invalid statement in C.
- { **True** or False } Suppose that `x` is declared as an integer and `y` is declared as an integer with initial value of 7. Just after these two statements there is the statement `x = y = 5;`. Then the value of `x` is equal to 5 after executing the the third statement `(x = y = 5;)`.
- { True or **False** } When `x` is declared as an integer, then the value of `x` is equal to 0 after executing the statement `x = (3 == 5);`
- { True or **False** } When `x` is declared as an integer, then the value of `x` equal to 100 after executing the statement `x = 0101 && 0102;`
- { True or **False** } When `x` is declared as an integer, then the value of `x` equal to 100 after executing the statement `x = 0101 & 0102;`
- { True or **False** } In C the statement `x = 2 ^ 2` assigns `x` the value of 2.
- { True or **False** } In C the statement `x = (y = 2) + (z = 3);` assigns `x` the value of 2.
- { True or **False** } The line `int xxx[4] = {1, 5, 3, 7}` declares the array `xxx` with four elements and with the initial values `xxx[1]=1`, `xxx[2]=5`, `xxx[3]=3`, and `xxx[4]=7`.
- { **True** or False } The line `int x=99, y=99;` declares `x` and `y` to be integers and assigns them the initial value of 99.
- { **True** or False } The Unix (Linux) command `cd` is used to change the current directory.

- { True or False } The Unix (Linux) command `ls` is used to list the content of the current directory.

2 Statements [2 points]

Simplify the following statement (i.e., rewrite it so that it is easier to understand what it means)

```
if ( ( a > b && d < c ) || ( d >= c && a > b ) ) a=5; else b = 7;
```

under the assumption that variables a, b, c, d are integers.

```
if ( a > b ) a=5; else b = 7;
```

3 If statement [2 points]

Clearly indicate what is the value printed when the following block of code is executed.

```
{ int x=3;
  int y=2;
  if (x == 1)
    y++;
    x *= y+3;
  { int y;
    y = 1;
  }
  printf("%d %d\n",x,y);          -->      15   2
}
```

4 If statement [2 points]

Clearly indicate whether the else matches with the first if or the second if.

```
if (x == 1)
  if (y == 2)
    printf("***\n");
else                               -->      2nd
  printf("###\n");
```

5 If statement [2 points]

For the code given below clearly indicate what the `printf` command prints.

```

{int i,j;
  int n,m;
  n = 3;
  m = 4;
  for(i = 1, j = 2; i < n || j < m ; i++, j++)
  { printf("The value of i is %d The value of j is %d \n",i, j);
  }
}

```

The value of i is 1 The value of j is 2
The value of i is 2 The value of j is 3

6 Switch [4 points]

What is printed when the input value for `s` is 1? How about when the input value is 2? How about when the input value is 3? How about when the input value is -1 ?

```

{
  int s;
  scanf("%d",&s);
  switch( s )
  {case 1:  s = 5;
   case 2:  s += 5;
   default: s *= 2;
  }
  printf("s is %d\n",s);
}

```

For `s=1` it prints "s is 20, For `s=2` it prints "s is 14, For `s=3` it prints "s is 6, For `s=-1` it prints "s is -2

7 Switch [2 points]

What is printed after executing the following code?

```

{
  int i, s;
  s = 2;
  for ( i = 0; i<= 10; i++, s++)
    { i++; s++;

```

```
        if (i < 6) continue;
            else break;
        printf("i is %d\n",i);
    }
    printf("s is %d\n",s);           -->  s is 9
}
```