EE 1910

Dr. Johnson

Homework 6

1 – Identify the type of each expression. 20pts

```
int a,b;
float c,d;
char e,f;
bool g,h;
```

a == b

int, int
== → int or bool

int

c + a * b / d – f

int and char
promoted to float

float

a >= c - f + a

<> = lowest
precedence
>= → int or bool

int

f / b * c <= a * e / b

int

a % b * c / f % g

all same precedence
L-> R
float mod anything is error
note: int mod bool is OK

compile error

2 – evaluate each statement and indicate the type of the result. 30pts

```
int a = 4;    float c = 1.1;    char e = 'a';
int b = 5;    float d = 2.2;    char f  = 'C';
int foo;      float boo;        char soo;
```

boo = b - a;

5 - 4 → 1
boo float → 1.0

1.0	float

boo = b*a;

5*4 → 20
boo float → 20.0

20.0	float

foo = c / d;

1.1/2.2 = 0.55
foo int → 0

0	int

boo = b % (int) c + 2;

(5 % 1)+ 2
0 + 2 → 2
boo float → 2.0

2.0	float

foo = (int) b / c * a;

(5 / 1.1) * 4
4.545 * 4 = 18.18…
foo int → 18

18	int

3 – Evaluate the following expressions individually. 30pts

int a = -3; int b = 6; int c = 1; int d = -2;

a + b > c + d

(-3 + 6) > (1 + -2)
3 > -1 → T

True or 1

a – 2 * b + b > c * 2 /3

(-3 – (2 * 6) + 6) > ((1 * 2) / 3)
(-3 – 12 + 5) > 0
-10 > 0 → F

False or 0

3 * b / 4 % 5 && b

((3 * 6 / 4) % 5) && 6
(4 % 5) && 6
4 && 6 → T

True or 1

d && c < (b + 5) || b

-2 && ((1 < 11) || 6)
-2 && (T || 6)
-2 && T → T

True or 1

(4 + 5 * b >= c – 4) && (c – 2)

((4 + 5 * 6) >= (1 – 4)) && (1- 2)
(34 >= -3) && -1
T && T → T

True or 1

4 – provide a single code statement to achieve each result. 20pts

```
int a, b, c;
float d,e,f;
char g, h;
```

multiply a by 8 without using the * or + operation

```
a = a << 3;
```

if c students share b burritos equally, set a equal to the minimum number of burritos left over

```
a = b % c;
```

set d equal to the number of whole cups of Cheerios in a box if e is the density of Cheerios in cups/volume and f is the volume in a box

```
d = (int)(f * e);
```

set c equal to 2 if d and f are equal otherwise set c equal to 8
(only operations we covered allowed)

```
c = 8 - 6*(d==f);
```

Given that g is any lower case letter, convert it to an upper case character

```
g -= 0x20; (could use 32)
```