Ava Zahedi
ME 333
Quiz 1

1. Integer overflow is when you are trying to store or calculate an integer value that exceeds the max possible value based on the allocated memory.
   For example:
   int a = 2^31 - 1;
   int b = 5;

   a+b will not give you the integer 2^31 + 4 because signed ints cannot store values that large due to their 4-byte size

2. int my_variable = 100;
   Declares a variable of type int called "my_variable" and assigns the value to 100. It only exists in the scope that it was declared in. For example if it was declared in a function it only lives in that function, or if it is inside a single C program it is only valid there.

   #define MY_VAR 100
   Creates a variable to be used in the header file such that MY_VAR is a variable that can be used in any file that includes that header. This way, MY_VAR can be used in multiple C files with the same name and value.

3. The compilation process consists of 4 main stages:
   a. Preprocessing
      Cleans up the code in preparation for compiling, including header files, removing comments, and accounting for helper functions.
   b. Compiling
      Converts the C code into assembly language so that it's closer to the computer's low-level language.
   c. Assembling
      Converts assembly into object code, which is even lower level and essentially bits.
   d. Linking
      Combines multiple object code files together to get one comprehensive executable.

4. Answers are bolded
   unsigned char q = 4;
   a. q = q | 0b1101;
      q = 4 = 0b0100
      0b0100 | 0b1101 = 0b1101
      q = 0b1101
   b. q = q << 1;

Left shifting multiplies by 2

q = 8

q = 0b1000

c. q = q + 128

Unsigned chars can go 0 to 255 so this is fine

q = 132

q = 0b10000100

5.
```
#include <stdio.h>

int i;
for (i=200; i<300; i+=3)
{
    printf(“%d ”, i);
}
```

6.
```
typedef struct {
    char fname[100];
    char lname[100];
    char letter_grade;
    float num_grade;
} Student
```

7.
```
Student studs[5];
int i;
for (i=0; i<5; i++)
{
    Student s = Student();
    s.num_grade = 0.0;
    studs[i] = s;
}
```

8.
```
#include <stdio.h>

void fill_info(Student *s[])
{
    int num_students = 0;
    while (num_students < 5)
    {
        char temp_f[100];
        char temp_l[100];
        float temp_grade;
        printf(“Enter data: firstname lastname percentage \n”);
```

```
                scanf(“%s %s %f”, temp_f, temp_l, &temp_grade);
                if (temp_grade >= 0 && temp_grade <= 100)
                {
                        s[num_students]->fname = temp_f;
                        s[num_students]->lname = temp_l;
                        s[num_students]->num_grade = temp_grade;
                        num_students++;
                }
                else
                {
                        printf(“Grade value not valid, enter data again \n”);
                }
        }
    }
```

```
9.  Student change_letter_grade(Student s)
    {
        if (s.num_grade >= 90)
        {
                s.letter_grade = A;
        }
        else if (s.num_grade >= 80)
        {
                s.letter_grade = B;
        }
        else if (s.num_grade >= 70)
        {
                s.letter_grade = C;
        }
        else if (s.num_grade >= 60)
        {
                s.letter_grade = D;
        }
        else
        {
                s.letter_grade = F;
        }

        return s;
    }
```

```
10. int i;
    for (i=0; i<5; i++)
```

```
{
        studs[i] = change_letter_grade(studs[i]);
}
```

11.

```
int i;
for (i=0; i<5; i++)
{
        printf(“%s %s %c \n”, studs[i].fname, studs[i].lname, studs[i].letter_grade);
}
```