Problem 1

johnny vastola

Binary	Unsigned	Signed	1’s Complement	2’sComplement	Biased
1001 0110	150	-22	105	-106	23
0110 1001	105	105	105	105	-22
0011 1010	58	58	58	58	-69
1111 0000	240	-112	15	-16	113

Fill T/F in the following table:

Property	unsigned	Signed	1’s Comp	2’s Comp	Biased
Can represent positive numbers	t	t	t	t	t
Can represent negative numbers	f	t	t	t	t
Has more than one representation for 0	f	f	t	f	f
Use the same addition process as unsigned	t	t	f	f	t

c. What is the value in decimal of the most negative 16-bit 2’s complement integer?

-32768

d. What is the value in decimal of the most positive 16-bit signed integer?

32767

Problem 2

```
char** copyStrArray(int argc, char **argv)
{
   int i;
   char **str;
   // allocating the number of strings as memory
   str = malloc(argc - 1);
   // looping from 0 to argc - 1
   for(i = 0; i < argc - 1; i++)
   {
```

```
  {
    // allocating memory as length of each string
    // +1 as null character
    str[i] = malloc(strlen(argv[i + 1]) + 1);
    // copying the string
    strcpy(str[i], argv[i+1]);
  }
  return str;
}
```

Problem 3

a.

```
 char* upcase(char* str) {
        char* p;
        char* result;
        result = (char*)
malloc(_strlen(str)+1__);
        strcpy(__result_, ____str_);
        for( p=result; *p!='\0'; p++ ) {
/* Fill-in 'A' = 65, 'a' = 97, 'Z' = 90 , 'z' = 122 */
if(*p>='a' && *p<='z') *p=*p-32;}
        return result;
```

b.

```
void upcase_by_ref( char** n ) {
  *n=upcase(*n)
}

void upcase_name(char* names[], int i) {
   upcase_by_ref( &(names[i]));
}
```

Problem 4

a.

```
void setName(Student *s, const char *name) {

strcpy(s->name, name);

}

unsigned long getStudentID(const Student *s) {

return s->sid;

}

void setStudentID(Student *s, unsigned long sid) {

s->sid = sid;

}
```

b.

```
Student* makeDefault(void) {
 Student s;
 setName(&s, "John");
 setStudentID(&s, 12345678);
 return &s;
}
```

You can't return a pointer to a locally declared variable (Student s).