Practice Problems for Finals

Arrays: Review

- What will be the output of the program?

```
int a[10] = {10};
for (int x = 0; x < 10; x++)
{
   cout << a[x] << endl;
}
```

Arrays: Review

- What will be the output of the program?

```
string arr = "Hello\0World";
cout << arr;
```

Arrays: Review

- What will be the output of the program?

```
char name[5] = {'a', 'b', 'c', 'd', 'e'};
cout << name << endl;
```

Arrays: Review

- What will be the output of the program if the array begins at address 65486?

```
int main()
{
    int arr[] = {12, 14, 15, 23, 45};
    cout << arr << endl << &arr;
}
```

Arrays: Review

- What will be stored in beta[3][3]?

```
for (int i = 0; i < 3; i++)
    for (int j = 0; j < 3; j++)
        beta[i][j] = i * j;
```

Arrays: Review

- What will be the output?

```
void print(int arr[], int size)
{
    for(int x = 0; x < size; x++)
    {
        cout << arr[x] << endl;
    }
}
```

```
main()
{
    int arr[3][3] = {
                        {1,2,3},
                        {4,5,6},
                        {7,8,9}
                    };
    print(arr[2], 3);
    print(arr[1], 3);
    print(arr[0], 3);

}
```

Arrays: Review

- Write a function
  bool myFunction(int arr[3][3], int colId, int rowID)
  That will return true if the specific row is equal to the specific column passed a parameter.

- Write a function
  bool myFunction1(int arr[3][3], in colId)
  That will return true if any of the row is equal to the specific column passed as a parameter
- Write a function
  bool myFunction2(int arr[3][3])

That will return true if any of the row is equal to any of the column in the 2D array.

Structures: Review

- Consider the following statements:

```
struct nameType
{
   string first;
   string last;
};
```

```
struct courseType
{
     string name;
     int callNum;
     int credits;
     char grade;
};
```

```
struct studentType
{
     nameType name;
     double gpa;
     courseType course;
};
```

```
studentType student;
studentType classList[100];
courseType course;
nameType name;
```

Structures: Review

- Mark the following statements as valid or invalid. If a statement is invalid, explain why

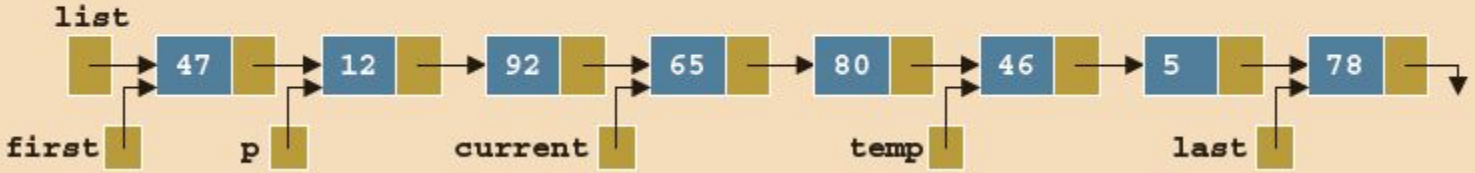
a. student.gpa = 3.76;
b. student.name.last= "Anderson";
c. classList[1].name = student;
d. classList[0].callNum = 0;
e. student.name = classList[10].name;
f. course = classList[0];
g. cin << classList[0];
h. for (int j = 0; j < 100; j++)
        classList[j].course = course;
i. classList.name.last = " ";
j. course.credits = studentType.course.credits;

Linked List:

1. Suppose that you have the following definitions:

```
struct nodeType
{
    int info;
    nodeType *link;
};
```

```
nodeType *list, *first, *current, *last, *temp, *trail, *p, *q;
```



Linked List:

```
struct nodeType
{
    int info;
    nodeType *link;
};
```

What will be the Output of the following:

a. cout << p->info;
b. q = p->link;
   cout << q->info << " " << current->info;
c. cout << current->link->info;
d. trail = current->link->link;
   trail->link = nullptr;
   cout << trail->info;
e. cout << last->link->info;
f. q = current->link; cout << q->link->link->info

Linked List:

```
struct nodeType
{
    int info;
    nodeType *link;
};
```

What is the value of each of the following relational expressions?

a. p->link->link == current
b. first->link->link->info == 92
c. temp->link == 0
d. last->link == nullptr
e. list->link == p
f. p->link->link->link->info == temp->info