

Leave A Legacy



Let's make this the project that

OUR

Illini era made happen

Announcements

MP3 available, due 2/22, 11:59p. EC: 2/12, 11:59p.

MP 3.1 will be on Exam 1.

Exam 1: 2/19, 7-10p, in rooms tba. 75min exam, given 3hr.

TODAY: templates-fin
linked memory

Class templates:

```
template <class T>
class mypair {
private:
    T a, b;
public:
    mypair(T first, T second);
    T getmax();
};
```

```
template <class T>
T mypair<T>::getmax() {
    T retmax;
    retmax = (a>b? a : b);
    return retmax;
}

template <class T>
mypair<T>::mypair(T first, T second){
    a = first;
    b = second;
}
```

Challenge1: write the function signature for the copy constructor (if we needed one) for this class.

_____ :: _____ (_____)

Challenge2: How do you declare a dynamic array of `mypairs` of integers?

Challenge3: How do you allocate memory if you want that array to have 8 elements?

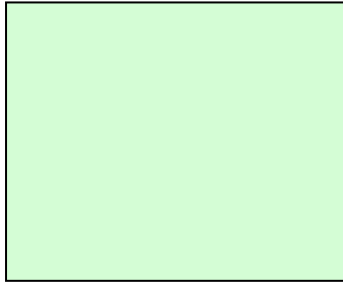
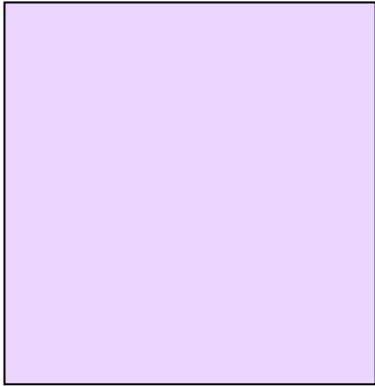
A note on templates:

```
template <class T, class U>
T addEm(T a, U b) {
    return a + b;
}

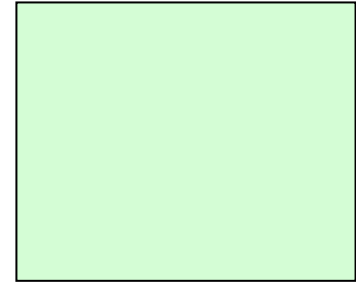
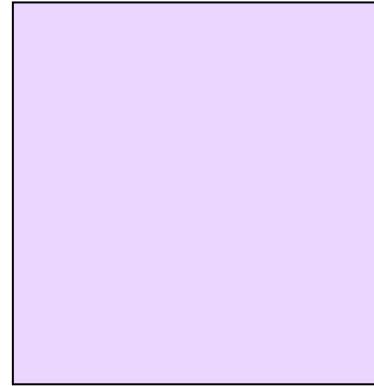
int main() {
    addEm<int, int>(3, 4);
    addEm<double, int>(3.2, 4);
    addEm<int, double>(4, 3.2);
    addEm<string, int>("hi", 4);
    addEm<int, string>(4, "hi");
}
```

Template compilation:

Old:

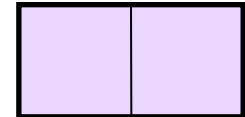


New:



Toward a new memory model:

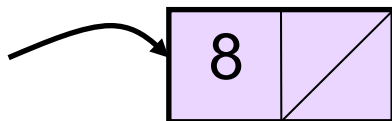
```
struct listNode {  
    LIT data;  
    listNode * next;  
    listNode(LIT newData) : data(newData), next(NULL) {}  
};
```



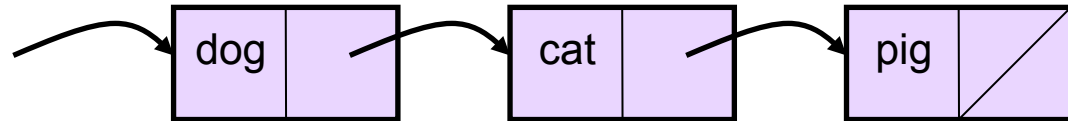
What is the result of this declaration?

```
listNode<int> nln(5);
```

What declaration and initialization would result in this memory configuration?



Example 1: `insertAtFront<farmAnimal>(head, cow);`



*`void insertAtFront(listNode * curr, LIT e) {`*

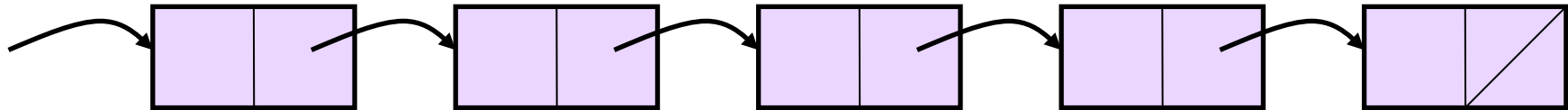
}

Running time?

```
struct listNode {  
    LIT data;  
    listNode * next;  
    listNode(LIT newData) : data(newData), next(NULL) {}  
}
```

8 4 2 6 3 0 1 2

Example 2:

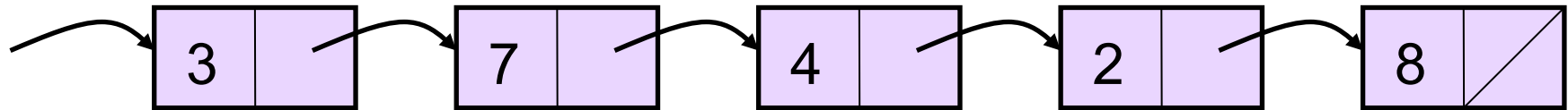


*void printReverse(listNode * curr) {*

3
Running time?

```
struct listNode {  
    LIT data;  
    listNode * next;  
    listNode(LIT newData):data(newData),next(NULL){}  
}
```


Example 3: Find kth position (we'll need this later)



*//returns pointer to node k steps forward from *curr
listNode * findKth(listNode * curr, int k) {*

}
Analysis:

Find kth in array: