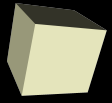




# Dynamic Memory

10/29/2007





# Opening Discussion

- Let's look at solutions to the interclass problem.





# Basics of Dynamic Memory

- Stack based variables are great, but they lack flexibility. Even with variable sizes for arrays there are things they can't do.
- A prime example is the problem of returning pointers from functions. Stack based memory can't be returned beyond its scope.
- There are also size limitations on the stack that can cause problems.
- Dynamic memory gets around these problems. With dynamic memory we can ask for a chunk of memory of a certain size and it will be safe to use until we tell the program that we are done with it.



- Dynamic memory comes from the heap.
- The heap is generally at the opposite end of memory that the program can access from the stack.
- It is less orderly than the stack. This provides flexibility, but has costs in speed.
- Getting memory from the heap can be an expensive operation, but a lot of the cost depends on the details of implementation.
- For small chunks of memory that you will be getting frequently the stack is generally better than the heap.



- We get dynamic memory in C with the malloc function. (You can also use calloc and realloc).
- Let's look at the man pages for malloc and write some code that uses malloc to see the syntax.
- Because malloc returns a void\* you will always need to do a type cast. (Unless you use the C99 standard.)
- All the memory you allocate needs to be given back to the system when you are done. You do this with free.
- Failure to call free leads to “memory leaks”, which are significant problems for long running programs.



# Arrays of Pointers

- Dynamic memory gives a lot of freedom when it comes to how we structure things in memory.
- We can make pointers to pointers that work as arrays of pointers.
- This allows us to make 2-D or higher arrays that aren't rectangular.
- Let's look at how we can do this in code.





- Why might heap allocation take longer than stack allocation?
- Remember that assignment #5 is due today.
- Quiz #4 is at the beginning of next class.
- Interclass Problem – Write a program in which you ask the user how many doubles they will enter, then allocate memory and read them in.

