Lecture 9.4

Templates

Templates

- Templates can be used with classes as well as with functions.
- Function templates are used when data types of arguments or return types are required to be generic.

What is template

- The programmer writes a single function template definition. Based on the argument types provided in calls to this function, the compiler automatically generates separated object code functions to handle each type of call appropriately.

Problem without template

- Suppose you have been asked to create linked list of integers, floats and strings.

Struct iNode{ iNode * pNext; iNode * pPrev; int nData;};	Struct fNode{ fNode * pNext; fNode * pPrev; float nData;};	Struct sNode{ sNode * pNext; sNode * pPrev; string nData;};

- You will have to rewrite entire code for node 3 times.

Solution

- Write just one templated node struct.

```
template<class T>
Struct Node{
    Node * pNext;
    Node * pPrev;
    T nData;
};
```


Function implementation with template

```
template <class T>
void AppendNode(NODE<T> * pNode);

template <class T>
void InsertNode(NODE<T> *pNode, NODE<T> *pAfter

template <class T>
void InsertNodeAt(NODE<T> *pNode, NODE<T> *pAfter);

template <class T>
void RemoveNode(NODE<T> *pNode);

template <class T>
void DeleteAllNodes();
```


Implementation of AppendNode

```
template <class T>
void AppendNode(NODE<T> *pNode)
{
  if (pHead == NULL) {     //if list is empty
    pHead = pNode;       //make head point to pNode
    pNode->pPrev = NULL;
  }
  else {
    pTail->pNext = pNode;  //make tail point to pNode
    pNode->pPrev = pTail;
  }
  pTail = pNode;       //tail is now pNode
  pNode->pNext = NULL;  //pNode next now points to NULL
}
```


Calling templated node struct from main

```
int main( )
{
    NODE <int> * iNode;
   NODE <string> * sNode;
   NODE <float> * fNode;

   //your code…

}
```

Dev C++ project

- We need a trick to include templated files in dev c++ project.
- There are two ways for this:
- Put the implementation of methods in the header file.
- Or
- Call implementation file together with header file. Eg.

```
#include “node.h”
#include “node.cpp”
```