Single Node < DataType > + \$val : DataType + \$next_: SingleNode < DataType > * + SingleNode() + SingleNode(DataType& x) + SingleNode(DataType& x, SingleNode < DataType > * next) 1:n SingleLinkList < DataType > + \$len : int + \$head_: SingleNode < DataType > * + \$tail_: SingleNode < DataType > * + SingleLinkList() + SingleLinkList(DataType x, int n); + SingleLinkList(const SingleLinkList < DataType > & list) + operator=(const SingleLinkList < DataType > & list) : SingleLinkList < DataType > & + GetElem(int n) const : DataType + Front() const : SingleNode < DataType > * + Back() const : SingleNode < DataType > * + FrontValue() const : DataType + BackValue() const : DataType + FindElem(DataType value): int + TopInsert(DataType value): void + BackInsert(DataType value): void + SelectInsert(DataType value): void + Insert(SingleNode < DataType > * node, DataType value) : void + Clear(): void + GetNthFromEnd(int n) const : DataType + RemoveNthFromEnd(int n): void + RemoveNthFromStart(int n): void + Delete(SingleNode < DataType > * node) : SingleNode < DataType > * + SortLink1(); void + SortLink2(): void + MergeSort(SingleNode < DataType > *SingleNode1) : SingleNode < DataType > * + Merge(SingleNode < DataType > * I1, SingleNode < DataType > * I2) : SingleNode < DataType > * +ReverseList(): void + GudgeEqual(const SingleLinkList < DataType > &a): bool + GetLength() const: int + Print() const : void + ~SingleLinkList() - IndexIsValid(int index) const : bool