User Id : weizhe.goh@digipen.edu | Started : 2020.06.03 10:01:05 | Submitted : 2020.06.08 00:13:24 | Score : 100%

DigiPen

Functions

Assignment

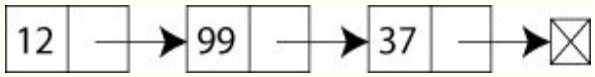
★ Rules 1 ☑

Read carefully and check all rules you agree with:

- ☐ Your code must represent your own individual work. Cheating of any kind (copying someone else's work, allowing others to copy your work, collaborating, etc.) will not be tolerated and will be dealt with SEVERELY.
- ☐ Each exercise has description which must be strictly followed.
- ☐ All programs must pass all tests in the main function (when given) to get the final grade. You are not allowed to make any change in the main function in this case.
- ☐ Keep the code proper formatted (correct indentation, max line width is 40 characters).
- ☐ Every week the instructor is available **during the lab time** to discuss following matters:
  - your disagreement with rule in this card,
  - misunderstanding of the current assignment specs,
  - solution for given problems.

Intro 1

- Linked list is a basic abstract data structure consisting of a group of nodes, which together represent a sequence of data records.



From Wikipedia

- In the simplest form, each node is composed of a data portion and a link to the next node in the sequence.
- A node can be declared using a struct data type in the following manner:

```
struct Node {
   int data;  // .. Any data fields ..
   Node *next;
};
```

Intro 2

- Unlike fixed-size arrays, linked list can increase or decrease in size during run-time, and data can be inserted to or removed from any point along the linked list.
- It should be noted that there can be more than one way to link the nodes in a linked list, and the given example would be commonly called a **singly-linked list**.
- Typically, a node in a singly linked list is pointing to the next node in the list.
- To get the access to the all nodes in the list we need to know the first one which is usually referenced by a pointer called **head**.
- The last node has a nullptr value for the next field. This indicates the end of the list. The last node sometimes is called **tail**.

FAQ

Q: What would be an empty linked list?

A The head is a nullptr pointer:

```
Node * head = nullptr;
```

Q. How is a linked list created?

- A linked list is created by starting with an empty linked list and gradually adding nodes to the end of the list using dynamic memory allocation for all new nodes.

★ Code 1 ☑

- Make your code for C++ compiler using modern features as mush as possible.

Run

```
#include <iostream>
#include <cstdlib>

namespace myList
{
  struct Node
```

Specs 1

- [ ] Create and initialize a pointer to the first element of an empty list. Name this pointer **head**.
- [ ] In the main function using head create list of 3 nodes with data 1, 2 and 3, so that head points to the first node with data 1. First node is linked with second node with data 2, and so on.
- [ ] Develop function **print** that takes the address of a node of a list to print the node's data and data of all following linked nodes.
- [ ] Call the print function for created in the main function list to print "1,2,3".
- [ ] All the list nodes must be allocated using **new** operator. Make sure that **delete** operator is used to deallocate each node at the end of the main function.
- [ ] Use a namespace to scope your code.

```
  {
    int data;
    Node *next;
  };

  void print(Node *list)
  {
    Node* current = list;

    if(list == NULL)
    {
      std::cout << "List is Empty"
      <<std::endl;

      return;
    }

    while(current)
    {
      if(current->next!=NULL)
      {
        std::cout <<current->data <<",";
      }
      else
      {
        std::cout << current->data;
      }
      current = current->next;
    }
  }
}

using namespace myList;

int main()
{
  Node *head = nullptr;
  head = new Node;
  Node *node1 = new Node;
  Node *node2 = new Node;

  head->data = 1;
  node1->data = 2;
  node2->data = 3;

  head->next = node1;
  node1->next = node2;
  node2->next = NULL;

  print(head);

  delete node2;
  delete node1;
  delete head;

  return 0;
}
```

```
1,2,3
```

Survey

- What is approximate number of hours you spent implementing this assignment?

1

- Indicate the specific portions of the assignment that gave you the most trouble

–

By signing this document you fully agree that all information provided therein is complete and true in all respects.	Responder sign: