Complete the following program **prog.c**

```
/* 1- include the needed header files */

/* 2- define Node structure that contains:
      an integer data and a next node pointer */

/* 3- define Linked List structure that contains:
      a header node pointer */

/* 4- function insert_after_i inserts d after the node at position
      i of the linked list ll and returns nothing. Note1: if there
      is no position i, insert the node at the end. Note2: negative
      values of i cause the insert to occur at the front of ll */
void insert_after_i(linkedlist ll, int d, int i){ /* your code */ }

/* 5- function delete_from_front deletes the first node of the
      linked list ll and returns nothing */
void delete_from_front(linkedlist ll){ /* your code */ }

/* 6- function delete_duplicates removes all duplicates from the
      ordered linked list ll and returns number of deleted nodes */
int delete_duplicates(linkedlist ll){ /* your code */ }

/* 7- function delete_list deletes all nodes of the linked list ll,
      releases their memory and returns nothing */
void delete_list(linkedlist ll){ /* your code */ }

/* 8- function split_alternate copies all nodes at odd positions
      of linked list ll to the empty list odd int same order, and
      the rest of the nodes to the empty list even in the reversed
      order, and returns nothing */
void split_alternate(linkedlist ll,linkedlist odd,linkedlist even){
  /* your code */
}

/* 9- function print_list prints the values stored in the nodes of
      the linked list ll to standard output in the format:
      data1 -> data2 -> … -> NULL and returns nothing */
void print_list(linkedlist ll){ /* your code */ }

/* 10- function main:
      - creates three empty linked lists: L1, L2 and L3
      - inserts into L1 the values: 10, 1, 10, 1, 20, 2, 20, 3
      - splits L1 into L2 for the odd positions and L3 for the rest
      - prints the data stored in the three lists
      - deletes the duplicates from L2 and L3 independently
      - prints the data stored in the three lists  */
int main(){
   /* your code */
  return 0;
}
```