

ECE 1320 Optimization Methods
Winter 2003

Homework 5: Due in class Thursday February 13 2003

- This test contains 4 problems. They allow you to earn 100 points.
- Show your work, as partial credit can be given. You will be graded not only on the correctness of your answer, but also on the clarity with which you express it. **Be neat.**
- **No late submissions will be accepted.**
- Only homework returned in a 9in \times 12in envelope will be accepted. (If you cannot find such envelope, ask the Instructor.) Please, write your name and the class name (ECE 1320) on the envelope (write clearly, please).
- For the six problems an e-mail to the TA should be sent that contains the code and the executable of a program that implements the solutions to the problems as functions.

Write your name here: _____

- **Problem # 1 [20 points]**. Write a C++ function that given a list of integers as defined in class returns the sum of the even elements of the list (i.e., the second, plus the fourth, plus ...).
- **Problem # 2 [25 points]**. Modify the definition of the classes we have seen for a linked list of integer (i.e., `lNode` and `lList`) to define a new class *grape* as defined by the following C struct:

```

struct gNode {
    int data;
    lNode *down;
    gNode *next;
};

typedef gNode *grape;

```

where `lNode` is as defined in class. A grape is thus a linked list of “g elements” (G list) each of which has a “down list” which is a linked list of “l elements” (L list). Consider a grape with a G list made up of n elements, $n \geq 0$. A grape is called a *wistaria* if the L list of the i th element of the G list is longer of the L list of the $(i + 1)$ th element of the G list, for each i , $n > i \geq 0$. Write a boolean C++ function for the class *grape* that given a grape object returns `true` if the grape is a wistaria, `false` otherwise. (An empty G list can be considered a wistaria, the empty one.)

- **Problem # 3 [30 points]**. Write a recursive C++ function that given a list of integers ℓ , as defined in class, reverses the element of ℓ . For instance, if $\ell = \langle 3, 2, 1 \rangle$ the function should modify the list so that $\ell = \langle 1, 2, 3 \rangle$. The function should make no use of any data structure.
- **Problem # 4 [25 points]**. Write a C++ function that given two lists of integers sorted in increasing order merges them into a list which is sorted in decreasing order.