

CISS350: Data Structures and Advanced Algorithms
Quiz q10601

Name: YOUR EMAILScore:

Q1. Suppose you are working with *sorted* singly linked list (i.e., the keys in the singly linked list are in ascending order). Complete the `SLList::insert_sorted` method so that if `list` is sorted, then after calling `list.insert_sorted(key)`, `list` will be the original list but with `key` inserted into the right place, i.e., the keys in the new `list` are also in ascending order.

ANSWER:

```
#include <iostream>
#include <string>

class SLNode
{
public:
    SLNode(int key, SLNode * next)
        : key_(key), next_(next)
    {}
    int key_;
    SLNode * next_;
};

std::ostream & operator<<(std::ostream & cout, const SLNode & n)
{
    cout << n.key_;
    return cout;
}

class SLList
{
public:
    SLList()
        : phead_(NULL)
    {}
    void insert_sorted(int key)
    {
        // TODO
    }
    SLNode * phead_;
};

std::ostream & operator<<(std::ostream & cout, const SLList & list)
{

```

```
        std::string delim = "";
        cout << "[";
        SLNode * p = list.phead_;
        while (p != NULL)
        {
            cout << delim << (*p);
            delim = ", ";
            p = p->next_;
        }
        cout << "]";
        return cout;
    }

int main()
{
    SLList list;
    list.insert_sorted(10); // expected: [10]
    std::cout << list << '\n';
    list.insert_sorted(5); // expected: [5, 10]
    list.insert_sorted(8); // expected: [5, 8, 10]
    list.insert_sorted(3); // expected: [3, 5, 8, 10]
    list.insert_sorted(0); // expected: [0, 3, 5, 8, 10]
    list.insert_sorted(6); // expected: [0, 3, 5, 6, 8, 10]
    std::cout << list << '\n';
    return 0;
}
```

INSTRUCTIONS

In the file `thispreamble.tex` look for

```
\renewcommand\AUTHOR{}
```

and enter your email address:

```
\renewcommand\AUTHOR{jdoe5@cougars.ccis.edu}
```

(This is not really necessary since alex will change that for you when you execute `make`.) In your bash shell, execute “`make`” to recompile `main.pdf`. Execute “`make v`” to view `main.pdf`.

Enter your answers in `main.tex`. In the bash shell, execute “`make`” to recompile `main.pdf`. Execute “`make v`” to view `main.pdf`.

For each question, you’ll see boxes for you to fill. For small boxes, if you see

```
1 + 1 = \answerbox{}
```

you do this:

```
1 + 1 = \answerbox{2}
```

`answerbox` will also appear in “true/false” and “multiple-choice” questions.

For longer answers that need typewriter font, if you see

```
Write a C++ statement that declares an integer variable name x.  
\begin{answercode}  
\end{answercode}
```

you do this:

```
Write a C++ statement that declares an integer variable name x.  
\begin{answercode}  
int x;  
\end{answercode}
```

`answercode` will appear in questions asking for code, algorithm, and program output. In this case, indentation and spacing is significant. For program output, I do look at spaces and newlines.

For long answers (not in typewriter font) if you see

```
What is the color of the sky?  
\begin{answerlong}  
\end{answerlong}
```

you can write

```
What is the color of the sky?  
\begin{answerlong}  
The color of the sky is blue.  
\end{answerlong}
```

A question that begins with “T or F or M” requires you to identify whether it is true or false, or meaningless. “Meaningless” means something’s wrong with the question and it is not well-defined. Something like “ $1 + 2 = 4$ ” is either true or false (of course it’s false). Something like “ $1+2 = 4?$ ” does not make sense.

When writing results of computations, make sure it’s simplified. For instance write 2 instead of $1 + 1$.

HIGHER LEVEL CLASSES.

For students beyond 245: You can put L^AT_EX commands in `answerlong`.

More examples of meaningless statements: Questions such as “Is $42 = 1+2$ true or false?” or “Is $42 = \{2\}^{\{3\}}$ true or false?” does not make sense. “Is $P(42) = \{42\}$ true or false?” is meaningless because $P(X)$ is only defined if X is a set. For “Is $1 + 2 + 3$ true or false?”, “ $1 + 2 + 3$ ” is well-defined but as a “numerical expression”, not as a “proposition”, i.e., it cannot be true or false. Therefore “Is $1 + 2 + 3$ true or false?” is also not a well-defined question.

More examples of simplification: When you write down sets, if the answer is $\{1\}$, do not write $\{1, 1\}$. And when the values can be ordered, write the elements of the set in ascending order. When writing polynomials, begin with the highest degree term.

When writing a counterexample, always write the simplest.