Midterm

Started: Oct 10 at 2:16pm

Quiz Instructions

Question 1	2 pts

If a method is labeled "const", it is not allowed to modify the data in its class.

○ True

○ False

Question 2	2 pts

If you declare a "const_iterator", then you cannot modify what it "points" to.

○ True

○ False

Question 3	2 pts

If the variables a and b are instances of a class, then "a = b" calls the class' copy constructor.

○ True

○ False

Question 4   2 pts

If the variable x is an instance of a class and b is a procedure, then "b(a)" calls the class' copy constructor.

○ True

○ False

Question 5   2 pts

If you define a single constructor of a class, and that constructor has parameters, then you can still declare a local variable, whose type is that, class without parameters.

○ True

○ False

Question 6   2 pts

If you detect an error in a constructor, then you should return NULL.

○ True

○ False

Question 7   2 pts

If c is a class and you've declared:

```
vector <c> v";
```

then you have to define a copy constructor for c to use v.

○ True

○ False

Question 8   2 pts

If c is a class and you've declared "vector <c *> v", then the following loop causes a memory leak:

```
v.resize(10);
for (i = 0; i < 5; i++) v[0] = new c;
```

○ True

○ False

Question 9   2 pts

If a method is "protected", then you cannot call it from another method of the same class.

○ True

○ False

Question 10

2 pts

If you have declared:

```
const int *p;
int i;
```

then you can do:

```
p = &i;
```

but not:

```
p = new int;
```

○ True

○ False

Question 11

2 pts

If c is a class and you've declared:

```
vector <c> v;
c x;
```

and then later you say:

```
v.resize(5, x);
```

c's destructor will eventually be called six times.

○ True

○ False

Question 12   2 pts

If c is a class and you've declared:

```
vector <c> v;
```

and then later, when v's size is zero, you say:

```
v.resize(5);
```

c's copy constructor will called five times.

○ True

○ False

Question 13   10 pts

We compile the following program to a.out:

```
#include <vector>
#include <iostream>
using namespace std;

int main()
{
  size_t i;
  int sz, j, n;
  vector < vector <int> > v;

  cin >> i;
  v.resize(i);

  for (i = 0; i < v.size(); i++) {
    cin >> sz;
    for (j = 0; j < sz; j++) {
      cin >> n;
      v[i].push_back(n);
```

1: 5
2: 4
3: 4
4: 9

Question 14

To the right is a hash table that stores strings. An empty string means that the hash table entry is empty. Please answer the questions below. Do not answer the questions as if one affects the other. Answer them all with respect to the table at the right. For example, you should not answer part 5 as if "Fred" were inserted into the table. Instead, you simply answer with respect to the table to the right.

When you answer a question, answer it on its own line with its question number at the beginning of the line, followed by a space.

Please answer the following questions

1: What is the load factor of the table? You can give your answer as a fraction or as a decimal.

2: Suppose we try to find "Frank", who has a hash value of 92863224. What is the first index that we look at to find "Frank"?

3: Suppose we are using linear probing. How many hash table entries must we check before we can determine that "Frank" is not in the table?

4: Suppose "Fred" has a hash value of 1198744606. If we insert "Fred" into the table using linear probing, into what index does "Fred" go?

5: Suppose "Betty" has a hash value of 1939399752. If we insert "Betty" into the table using quadratic probing, into what index does "Betty" go?

6: Suppose "Luther" has a hash value of 2584999427, and that when we hash "Luther" using a second hash function, its hash value is 2470113361. If we insert "Luther" into the table using double hashing, into what index does "Luther" go?

```
     ----------------------------
 0 | Hunter                     |
 1 |                            |
 2 |                            |
 3 | John                       |
 4 |                            |
 5 |                            |
 6 | Evelyn                     |
 7 | Lilly                      |
 8 |                            |
 9 | Allison                    |
10 | Ella                       |
11 |                            |
12 | Natalie                    |
13 | Jack                       |
14 | Avery                      |
15 |                            |
16 |                            |
```

```
17 |                                |
18 |  Andrew                        |
19 |                                |
      ------------------------------
```

p   0 words   </> ↗ ⋮
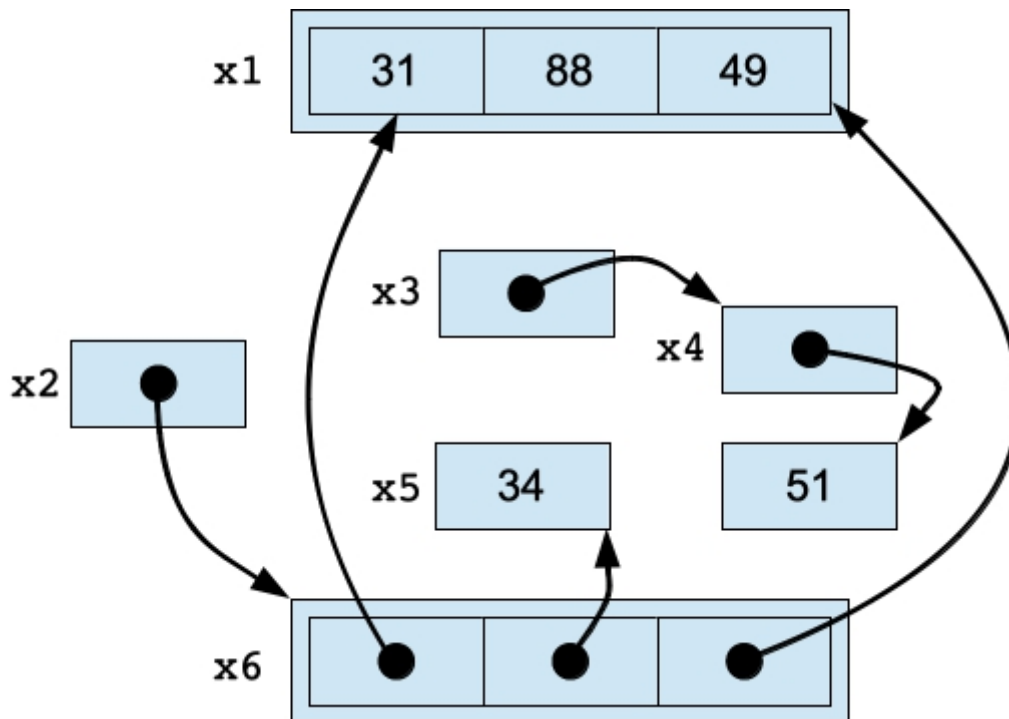
Question 15   12 pts

Above is a picture of the variables in our program, which are integers, pointers and vectors. Please answer the sub-questions below. Please answer each sub-question on its own line, preceded by the sub-question number and a space.

For Sub-questions 1 through 6, please use the following multiple choice answers -- put the letter in your answer box for these. Don't type in the type declaration.

```
A: int        H: vector <int>
B: int *      I: vector <int *>
C: int &      J: vector <int> *
D: int **     K: vector <int *> *
E: int &&     L: vector < vector <int> >
F: int ***    M: vector < vector <int *> >
G: int &&&    N: vector < vector <int> > *
              O: vector < vector <int *> > *
```

Subquestion 1: What is the type declaration for the variable x1?
Subquestion 2: What is the type declaration for the variable x2?
Subquestion 3: What is the type declaration for the variable x3?
Subquestion 4: What is the type declaration for the variable x4?
**Subquestion 5:** What is the type declaration for the variable x5?
Subquestion 6: What is the type declaration for the variable x6?
Subquestion 7: Which pointer was created with **new**? Please answer with the name of a variable.

Subquestion 8: Did $x1[0]$ need to be set before $x6[0]$? Please answer Yes or No.

Subquestion 9: Did $x5$ need to be set before $x6[1]$? Please answer Yes or No.

p   0 words   </>

Question 16   10 pts

Please tell me the output of the program below. Put your answer all on one line -- if there's a newline in the program's output, simply replace it with a space, or even no space. Either will be fine.

So, for example, if the output is:

A
B
C

Then your answer should be "ABC" or "A B C".

```cpp
#include <iostream>
using namespace std;

class Data {
  public:
    Data();
    Data(const Data &m);
    ~Data();
};

Data::Data()
{
  cout << "A" << endl;
}

Data::Data(const Data &m)
{
  cout << "B" << endl;
}

Data::~Data()
{
  cout << "C" << endl;
}
```

```cpp
void proc1(Data f)
{
  cout << "D" << endl;
  throw (string) "E";
}

int main()
{
  Data f;

  try {
    cout << "F" << endl;
    proc1(f);
    cout << "G" << endl;
  } catch (const string &s) {
    cout << s << endl;
    return 0;
  }

  cout << "H" << endl;
  return 0;
}
```

p
0 words

Question 17

10 pts

Please write a procedure that does the following:

- It takes one parameter, which is a vector of strings. It should be a const reference parameter.
- It returns a list of strings.
- The list should contain the even-indexed strings in the vector, in the order in which the strings appear in the vector.

Please don't bother with **include** statements or **using** statements. I will be grading you on your syntax along with the correctness and efficiency of the program.

Be sure to format the "Paragraph" as "Preformatted". That way you get a nice, fixed-width font.

p   0 words   </> ↗ ⋮

Question 18

1 pts

Please write a C++ program that does the following:

- It reads lines of text on standard input.
- If any line does not begin with a capital letter, then it should throw the string "Bad Line" as an exception. It should not catch this exception.
- After reading in all the lines, it should first print all lines that begin with characters between 'A' and 'G' (there are seven of these characters) in the order that they were read.
- It should then print all lines that begin with characters between 'H' and 'N' (there are seven of these characters) in the order that they were read.
- It should then print all lines that begin with characters between 'O' and 'U' (there are seven of these characters) in the order that they were read.
- It should then print all of the other lines in the order that they were read.

So, for example:

```
UNIX> cat input-1.txt
Morgan Circa
Gabriel Prototype
Zoey Foppish
Paige Gurgle
Liam Ymca
Xavier Austin Bethought
Isabella Marks
Cooper Chinook
Madeline Allison Coconut
Abigail Marina
UNIX> a.out < input-1.txt
Gabriel Prototype            # Here are the A-G
Cooper Chinook
Abigail Marina
Morgan Circa                 # Here are the H-N
Liam Ymca
Isabella Marks
Madeline Allison Coconut
Paige Gurgle                 # Here is the O-U
Zoey Foppish                 # Here are the others
Xavier Austin Bethought
UNIX> cat input-2.txt
1, 2, Buckle My Shoe
UNIX> a.out < input-2.txt
libc++abi.dylib: terminating with uncaught exception of type std::__1::basic_stri
ng<char, std::__1::char_traits<char="">, std::__1::allocator >
Abort trap: 6
UNIX>
</char,>
```

You should be able to implement this with just one vector (it can be a vector of vectors), and you shouldn't have to run through all of the elements of this vector multiple times. I'm not saying that you *have* to do it this way, but you will lose some points if you use multiple vectors, or if you have one vector which you traverse multiple times.

And don't assume that there's an **isupper()** procedure -- you can write that expression yourself.

Edit   View   Insert   Format   Tools   Table

12pt ⌄   Paragraph ⌄   **B**   *I*   U̲   A ⌄   ✎ ⌄   T² ⌄   ⋮

p   ⌨   ♿ | 0 words | </>   ⤢   ⋮

Saving...   Submit Quiz