MIDTERM EXAM 1 EMPLID

CSCI 135 NAME: FIRST LAST

1. (18%) Write the following Linux terminal instructions:

(a) Compile a program called **`myprog.cpp`** into an executable file with a default name.

```
g++ myprog.cpp
```

(b) Run this executable.

```
./a.out
```

(c) Run this executable with input redirection from the text file **`data.txt`** .

```
./a.out < data.txt
```

(d) Compile the same program **`myprog.cpp`** into an executable file called **`myprog`** .

```
g++ -o myprog myprog.cpp
```

(e) Run the executable **`myprog`** .

```
./myprog
```

(f) Run the executable **`myprog`** with input redirection from the text file **`data.txt`** .

```
./myprog < data.txt
```

2. (9%) Write a C++ function that calculates: $$x = \frac{b+\sqrt{b^3-1}}{n}$$

```
#include <cmath>
double foo(double n, double b)
{
   return (b + sqrt(pow(b, 3) - 1)) / n;




}
```

3. (15%) Suppose your program has the following declarations to represent information about a person:

```
string first_name;
int birth_year;
bool female; //true if woman, false if man
```

Write C++ logical conditions corresponding to each of the following sets. Your answers should be as compact as possible and cover all cases.
For example, all persons born after 1997 and before 2001:
`birth_year > 1997 && birth_year < 2001`

(a) All men, who were born in 2001 or later and whose first name starts with the letter 'S' or letter 'P'.

```
!female && birth_year >= 2001 && (first_name[0] == 'S'|| first_name[0] == 'P')
```

(b) All women, born last year, whose first name ends with the letter 'y'.

```
female && birth_year == 2018 && first_name[first_name.length() - 1] == 'y'
```

(c) All persons named Alex, who turned 18 or 19 last year.

```
first_name == "Alex" && (birth_year == 1999 || birth_year == 2000)
```

4. (17%) Write a function: `string remove_a(string word)` that removes all occurrences of the letter '`a`' from string `word`.

```
#include <string>
#include <iostream>
using namespace std;
string remove_a(string word) {
  string new_word;
  for (int i = 0; i < word.length(); i++) {
    if (word.substr(i, 1) != "a") {
      new_word = new_word + word.substr(i, 1);
    }
  }
  return new_word;

}
int main() {
   cout << remove_a("Isabella") << endl;
   return 0;
}
```

NAME: FIRST LAST

5. (18%) Consider the following program fragment:

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

int foo(int n, int m);

int main()
{
    int a = 2; //SPECIAL LINE
    int b = 3;
    for (int i = 0; i < 2; i++) {
        b = foo(i, b);
        cout << a << " " << b << endl;
    }
    return 0 ;
}

int foo(int n, int m)
{
    static int x = 0;
    x = x + 2;
    n = n + 2;
    return (x + n + m);
}
```

(a) What does the program output?

```
2 7
2 14
```

(b) Underline all formal parameters in the program.

(c) Circle all actual arguments in the program.

(d) Draw a dashed box around all prototypes in the program.

(e) Draw a solid box around the scope of the variable declared on `//SPECIAL LINE`

(f) What is the value of variable `x` just before the `foo()` function returns from its last call?

`4`

6. (23%) Write a program that asks the user for a positive integer height. If they enter an illegal value, they must be prompted to enter a good one until they do. It then prints an equilateral right triangle of stars of that height. For example, if the height is `5`, the program should print:

```
    *
   * *
  * * *
 * * * *
* * * * *
```

```
#include <string>
#include <iostream>
using namespace std;
int main() {
//    cout << "Enter the height of the triangle: ";
    int height;
//    cin >> height;
    do {
        cout << "Enter the height of the triangle: ";
        cin >> height;
    } while (height < 0);
    for (int i = 0; i < height; i++) {
        for (int j = 0; j <= height; j++) {
            if (j < height - i)
                cout << " ";
            else cout << "*";
        }
        cout << endl;
    }
    return 0;
}
```