

Worksheet 1

Problem 1.

Assume that the following lines of code are inside the main function, with `#include <iostream>`, `using namespace std`, and all the string variables used having been previously declared.

```
cout << "Enter your name: ";
getline(cin, name);

cout << "\nEnter your UID: ";
int UID;
cin >> UID;

cout << "\nEnter your Major: ";
getline(cin, major);

cout << "\nEnter your residence hall: ";
getline(cin, hall);

cout << "\n" << UID << " is the ID of " << name << ", a " << major << " student
who lives in " << hall << endl;
```

- (a) Circle where the bug occurs.
- (b) Explain what you think will happen when running the program.
- (c) Is this a logic error or a compilation error? Why?
- (d) Add a fix to the problem you found in part (a).

Problem 2.

What is the output of the following code?

```
int cookies = 12;
int mms = 120;

if (mms % cookies != 0) {
    cout << "Can't evenly split M&Ms for each cookie!" << endl;
} else {
    cout << "We have " << mms/cookies << " M&Ms per cookie." << endl;
}
```

Problem 3.

This code snippet takes a certain “hour” and “weekday” and tries to tell you if you can buy turnips from Daisy Mae, the turnip seller.

```
int hour;
string weekday;
cin >> hour;
cin >> weekday;

if (weekday != "Sunday" || hour >= 12) {
    cout >> "Daisy Mae is not here!" >> endl;
} else {
    if (hour = 11) {
        cout >> "It's almost 12! Hurry up!" >> endl;
    } else {
        cout >> "Buy turnips with Bells." >> endl;
    }
}
```

(a) Find the 5 lines with mistakes in the code and fix them.

(b) Will this code compile? Why or why not?

(c) After you fix the bugs, imagine you input 11 for the hour, then “Monday” for the weekday. What will this program say to you?

Problem 4.

What will this program output? Can you explain every line of output?

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

int main() {
    int elligent = 64;
    int eresting = 0;
    double rainbow = 64.0;
    double stuf = 0.0;

    cout << elligent << endl;
    cout << rainbow << endl;

    eresting = elligent/2.5;
    stuf = rainbow/3;
    cout << rainbow/3 << endl;
    cout << stuf << endl;
    cout << elligent/2.5 << endl;
    cout << eresting << endl;
}
```

Problem 5.

Write a program that asks for a number between 0 and 99 and takes an integer input. If you input a number greater than or equal to 100, it will print "Liar, liar, plants for hire" and stop. Otherwise, it tells you if your number has a "tens" digit that would round up to 100.

Problem 6.

Write a program that takes in two numbers and a command of type string ("Add", "Subtract", "Multiply", "Divide"). Inputting an invalid command should cause the program to print out "Invalid command!" and stop.

Solution to Problem 1.

- (a) The bug occurs after line 6 in the problem above.
- (b) Since there is an integer input followed by a string input, the line `cin.ignore(10000, '\n');` must be inserted.
- (c) This is a logic error.
The program will still run, but it will assign an empty string to the identifier, `major`.
- (d) The full program (with string variables declared) is shown below.

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

int main()
{
    cout << "Enter your name: ";
    string name;
    getline(cin, name);

    cout << "\nEnter your UID: ";
    int UID;
    cin >> UID;
    cin.ignore(10000, '\n');

    cout << "\nEnter your Major: ";
    string major;
    getline(cin, major);

    cout << "\nEnter your residence hall: ";
    string hall;
    getline(cin, hall);

    cout << "\n" << UID << " is the ID of " << name << ", a " << major << " student
    who lives in " << hall << endl;
}
```

Solution to Problem 2.

The if-statement in the program specifies a condition where the remainder of mms/cookies is not equal to zero.

Since this is false for the values above, the output of the program would fall under the `else` category. The result of mms/cookies is 10, which is an integer because both of the identifiers are integers. Thus, the output of the program would be:

```
We have 10 M&Ms per cookie.
```


Solution to Problem 3.

(a) There are actually six lines with four different mistakes.

1. After line 3, we need to include the line `cin.ignore(10000, '\n');` since an integer input is followed by a string
2. In lines 7, 10, and 12, the right shift operators (`>>`) should be replaced with left shift operators (`<<`) since we are using `cout`
3. In line 9, the equal sign should be denoted as `==` instead of `=`
4. After line 13, there should be another closed braces

(b) The code will fail to compile because `>>` is not defined under `using namespace std;`

(c) The output of this code would be:

Daisy Mae is not here!

The final program is shown below:

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

int main(){
    int hour;
    string weekday;
    cin >> hour;
    cin.ignore(10000, '\n');
    cin >> weekday;

    if (weekday != "Sunday" || hour >= 12) {
        cout << "Daisy Mae is not here!" << endl;
    } else {
        if (hour == 11) {
            cout << "It's almost 12! Hurry up!" << endl;
        } else {
            cout << "Buy turnips with Bells." << endl;
        }
    }
}
```

Solution to Problem 4.

The program would output the following:

```
64
64
21.3333
21.3333
25.6
25
```

The first output is `elligent`, which is an integer assigned as 64. The compiler writes it out as `64`.

The second output is `rainbow`, which is a double assigned as 64.0. Since the final output has no nonzero decimal places, the compiler truncates it to `64`.

The third output is `rainbow/3`, which is a double divided by an integer. Thus, the output must be a double. The program truncates it to four decimal places as `21.3333`.

The fourth output is `stuf`, which is last defined as `rainbow/3`. Since `rainbow/3` is a double, and `stuf` is defined as a double on the top of the program, the output must also be a double. The program truncates it to four decimal places as `21.3333`.

The fifth output is `elligent/2.5`. Since `elligent` is an integer and `2.5` is a double, the output must be a double. Since the output has one decimal place, the program writes it out as `25.6`.

The sixth output is `eresting`, which is last defined as `elligent/2.5`. The answer is 25.6, but is truncated to the ones place because `eresting` was originally defined as an integer. Thus, the final output is `25`.

Solution to Problem 5.

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

int main()
{
    cout << "Enter a whole number between 0 and 99. " << endl;
    int number;
    cin >> number;

    if (number >= 100){
        cout << "Liar, liar, plants for hire." << endl;

    } else
    if (number < 50){
        cout << "Your number rounds to 0!" << endl;

    } else {
        cout << "Almost to 100!" << endl;
    }
}
```

Solution to Problem 6.

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

int main()
{
    cout << "Enter your first number. " << endl;
    int number1;
    cin >> number1;

    cout << "Enter your second number. " << endl;
    int number2;
    cin >> number2;
    cin.ignore(10000, '\n');

    cout << "Type \"Add\", \"Subtract\", \"Multiply\", or \"Divide\". " << endl;
    string operation;
    cin >> operation;

    if (operation == "Add" || operation == "Subtract" || operation == "Multiply" ||
        operation == "Divide"){

        if (operation == "Add")
            cout << "Result: " << number1 + number2 << endl;

        if (operation == "Subtract")
            cout << "Result: " << number1 - number2 << endl;

        if (operation == "Multiply")
            cout << "Result: " << number1 * number2 << endl;

        if (operation == "Divide")
            cout << "Result: " << number1 / number2 << endl;

    } else
        cout << "Invalid command!" << endl;
}
```