Understanding:

1. I need to create a program retFun.cpp
    1. It needs to use a function to check string contents and return if they are the same
2. I need to create a program randFun.cpp
    1. It needs to use a function to return random numbers in a given bounds set.
3. I need to create a program mpg.cpp
    1. It needs to calculate mpg from liters and miles traveled.
    2. It should ask to run again.
4. I need to create a program size.cpp
    1. It needs to calculate hat size, jacket size, and waist size from users height, weight, and age.
    2. It should ask to run again.
5. I need to create a program finalist.cpp
    1. It should generate three, unique, random numbers between 1-25.
6. I need to create a project secretWord.cpp.
    1. It should ask for a word.
    2. it should check to make sure it contains letters only.
    3. It should ask for letters from player 2.
    4. It should display correct letters and guessed letters.
    5. It should let the user know if they won or lost and what the word was.
    6. It should ask the user if they want to play again.

Design:

retFun.cpp

```
headers and includes
using statement
bool StrComp(string string1, string string2)
{
        if(string1.length() != string2.length())
        {
                return false;
        }
        else
        {
                for(int i = 0; i < string1.length(); i++)
                {
                        if(string1.at(i) == string2.at(i))
                        {
                                continue;
                        }
                        else
                        {
                                return false;
                        }
        }
```

```
            return true;
            }
}
int main()
{
 //boilerplate create string and get user input
 bool same = StrComp(string1, string2);
 if(same)
 {
 cout << "The strings are the same." << endl;
 }
 else
 {
 cout << "The strings are different." << endl;
 }
 return 0;
}
```

randFun.cpp

```
includes and using statements

int rand_int(int min, int max)
{
    int randomNumber = rand() % ((max - min) + 1) + min;
    return randomNumber;
}

int main()
{
    //seed the random number generator
    srand(time(NULL));

    int lowerBound;
    int upperBound;

    cout << "Please enter two integers." << endl;
    do {
        do {

            if (!cin) {
                cin.clear();
                cin.ignore(INT_MAX, '\n');
            }

            cout << "Number 1: ";
            cin >> lowerBound;
            cin.ignore(INT_MAX, '\n');
        }
```

```
        while (!cin);

        do {

            if (!cin) {
                cin.clear();
                cin.ignore(INT_MAX, '\n');
            }
            cout << "number 2: ";
            cin >> upperBound;
            cin.ignore(INT_MAX, '\n');
        }
        while (!cin);
    }
    while(lowerBound >= upperBound);

    cout << rand_int(lowerBound, upperBound) << endl;

    return 0;
}
```

mpg.cpp

using and includes statements

```
const double GALLONS_IN_LITER = 0.264179;

double mpgCalc(double liters, double miles)
{

    double gallons = liters * GALLONS_IN_LITER;
    double mpg = miles / gallons;

    return mpg;
}

int main()
{
    double miles = 0;
    double liters = 0;
    char tryAgain = 'n';

    cout << "Welcome to the MPG tool!" << endl << endl;

    do {

        do
        {
            cout << "Please enter number of miles traveled: ";
```

```
        cin >> miles;

        cout << "Please enter number of liters consumed: ";
        cin >> liters;

        //We won't mess with divide by 0 errors
        if(miles == 0 || liters == 0)
        {
            cout << "0 values are invalid. Please try again!" << endl;
        }
    }
    while (miles == 0 || liters == 0);

    cout << "You got " << mpgCalc(liters, miles) << " MPG while traveling." << endl;

    cout << "Run again? (y/n): ";
    cin >> tryAgain;
    tryAgain = tolower(tryAgain);
  }
  while(tryAgain == 'y');
}
```

finalist.cpp

```
int rand_int(int min, int max)
{
  int randomNumber = rand() % ((max - min) + 1) + min;
  return randomNumber;
}

int main()
{
  srand(time(NULL));

  int numbers[] = {0, 0, 0, 0};

  for(int i = 0; i < 4; i++)
  {
    bool inArray;
    int rand;

    do {

      inArray = false;
      rand = rand_int(1, 25);

      for (int j = 0; j < 4; j++) {
        if (numbers[j] == rand)
```

```
            {
                inArray = true;
                srand(time(NULL));
            }
        }
    }
    while(inArray);

    numbers[i] = rand;

    cout << rand << endl;
  }
}
```

secretword.cpp

```
using namespace std;

const int GUESS_NUM = 10;

bool alreadyGuessed(char guess, char (&guesses)[GUESS_NUM])
{
  for(int i = 0; i < GUESS_NUM; i++)
  {
    if(guesses[i] == guess && guesses[i] != 0)
    {
      return true;
    }
  }

  return false;
}

bool isAlphaString(string verify)
{
  bool alpha = true;

  for(int i = 0; i < verify.length(); i++)
  {
    if(!isalpha(verify.at(i)))
    {
      alpha = false;
    }
  }

  return alpha;
}
```

```
bool userWon(string verify, char (&guesses)[GUESS_NUM])
{
    int correctGuesses = 0;

    for(int i = 0; i < GUESS_NUM; i++)
    {
        for(int j = 0; j < verify.length(); j++)
        {
            if(verify.at(j) == guesses[i])
            {
                correctGuesses++;
            }
        }
    }

    if(correctGuesses == verify.length())
    {
        return true;
    }
    else
    {
        return false;
    }
}


int main()
{
    char playAgain = 'n';

    do {
        string secretWord;
        bool won = false;
        int num_guesses = 0;

        do {
            secretWord = "";

            cout << "Please enter a word (no numbers): ";
            cin >> secretWord;

        }
        while (!isAlphaString(secretWord));

        system("clear");

        char guesses[GUESS_NUM] = {0};

        while (num_guesses < GUESS_NUM) {
```

```
bool boolAlreadyGuessed = false;

char guess;

cout << "The word currently looks like: ";
for (int i = 0; i < secretWord.length(); i++) {
    bool letterMatch = false;

    for (int j = 0; j < GUESS_NUM; j++) {
        if (secretWord.at(i) == guesses[j]) {
            letterMatch = true;
            cout << guesses[j] << " ";
        }
    }

    if (!letterMatch) {
        cout << "_ ";
    }
}

cout << endl;

if (num_guesses > 0) {
    cout << "You already have guessed: ";
    for (int i = 0; i < GUESS_NUM; i++) {
        if (guesses[i] != 0) {
            cout << guesses[i] << " ";
        }
    }

    cout << endl;
}

cout << "Please enter a letter guess: ";
cin >> guess;

boolAlreadyGuessed = alreadyGuessed(guess, guesses);

if (boolAlreadyGuessed) {
    cout << "You already guessed that letter:" << endl;
    continue;
}
else {
    guesses[num_guesses] = guess;
}

if (userWon(secretWord, guesses)) {
    num_guesses++;
    cout << "You won with " << num_guesses << " guesses!" << endl;
```

```
            won = true;
            break;
        }

        num_guesses++;
        system("clear");
    }

    if (!won) {
        cout << "You didn't win! The secret word is: " << secretWord << endl;
    }
    else {
        cout << "The secret word is: " << secretWord << endl;
    }

    cout << "Play again? (y/n): ";
    cin >> playAgain;
    playAgain = tolower(playAgain);

  }
  while(playAgain == 'y');
}
```

Testing

retFun.cpp

Input	Expected Output	Actual Output
hello hello	they are same	same
hello goodbye	they are different	they are different
hello heloo	they are different	they are different

randFun.cpp

Input	Expected Output	Actual Output
1 25	random number between 1 and 25	same

mpg.cpp

Input	Expected Output	Actual Output
4 miles 2 liters	7.57(plus extra dec) MPG	7.57062

size.cpp

Input	Expected Output	Actual Output
weight: 160 height: 60 age: 20	hat: 7.733 jacket: 33.333 waist: 28.0702	same
weight: 160 height: 60 age: 30	hat: 7.733 jacket: 33.333 waist: 28.1702	same
weight: 160 height: 60 age: 40	hat: 7.733 jacket: 33.4583 waist: 28.6702	same

finalist.cpp

no input. Three random numbers out. Expected results were achieved.

secretWord.cpp

Input	Expected Output	Actual Output
.733 jacket: 33.333 waist: 28.1702	correct in 4 guesses	same
hello a b c d e f g h i j	incorrect. right word is hello	same

1. What did you learn?
    1. I didn't really learn anything this time around. I have been using functions in my past assignments, so this was pretty routine. There are places in my assignments that I could use functions, but I did it pretty last minute, so some refactoring is probably needed.
2. Was your understanding complete at the start of the project.
    1. I believe my understanding was complete at the start of the project.
3. Was your initial design adequate
    1. Besides minor tweaks, yes it was.
4. Did your tests work as expected
    1. Yes
5. Did implementation go without any problems
    1. I design in psuedo-code that is quite close to c++. This make implementation easy. My logic all seemed to translate really well to the actual projects.
6. What techniques have we covered this week and in past weeks that helped you approach the problem, did you need to use outside sources to help solve the problem (list sites, books, or other materials that were helpful)? Does this project seem related to previous projects and do you see any names for future projects that it might be related to?
    1. I was able to reuse code I wrote from last week. Otherwise, I didn't use any other sources.