Homework 3

5.32) *Guess The Alphabet*

Code

```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <ctype.h>

// initialize functions
void introduction();
char getUserGuess();
void guessTheLetter(int);
int compareLetters(char, char);


int main(void)
{
    int random;

    // display intro
    introduction();

    // choose a random letter
    srand(time(NULL));
    random = rand()%(26)+65; //26 is letters in English alphabet, 65 is ASCII for A

    printf("\n***Random letter is %c***\n", random); // Shows the random letter if needed
    // call the guessTheLetter function and pass it the random letter
    guessTheLetter(random);

    return 0;
} // end function main


void guessTheLetter(int random)
{
    char guess;
    int count = 0;
    int win = 0;

    // loop until user wins
    while (win == 0)
    {
        // call getUserGuess function and get a guess
        guess = getUserGuess();

        // turn guess into uppercase
        guess = toupper(guess);

        // call the compareLetter function and pass it the random and guess
        win = compareLetters(random, guess);

        count++; // increment count

```

```
52          // check if the user has won or not
53          if (win == 0)
54          {
55
56              if (guess < random) {
57                  printf("%d. Way behind! Please try again.\n",count);
58
59              }
60              else if (guess > random) {
61                  printf("%d. Way ahead! Please try again.\n",count);
62              }
63          }
64
65          else if (win == 1)
66          {
67              printf("%d. Great! Your guess is right!\n", count);
68
69              if (count < 5)
70                  printf("Either you know the secret or you are lucky!\n\n");
71              else if (count == 5)
72                  printf("I guess you know the secret!\n\n");
73              else
74                  printf("You should be able to do better!\n\n");
75
76          }
77      } // end while
78
79  } // end function guessTheLetter
80
81
82  void introduction()
83  {
84      printf("\nI have selected a character between A and Z.\n");
85      printf("Try to guess the character selected by me!\n");
86      printf("What is your guess?\n");
87      printf("_____\n");
88  } // end function introduction
89
90  char getUserGuess()
91  {
92      char guess;
93      // prompt for user input
94      printf("Enter your guess: ");
95      scanf("%c", &guess);
96      getchar(); // getchar() added because scanf also takes newline as input
97      return guess;
98  } // end function getUserGuess
99
100 int compareLetters(char random, char guess)
101 {
102     if (random == guess)
103         return 1;
104     else
105         return 0;
106 } // end function compareLetters
```

Output

- If the count is 5 or fewer:

```
I have selected a character between A and Z.
Try to guess the character selected by me!
What is your guess?
***Random letter is N***
Enter your guess: N
1. Great! Your guess is right!
Either you know the secret or you are lucky!
```

- If the count is 5:

```
What is your guess?
***Random letter is X***
Enter your guess: a
1. Way behind! Please try again.
Enter your guess: B
2. Way behind! Please try again.
Enter your guess: Z
3. Way ahead! Please try again.
Enter your guess: T
4. Way behind! Please try again.
Enter your guess: X
5. Great! Your guess is right!
I guess you know the secret!
```

- If the count is higher than 5:

```
What is your guess?
***Random letter is F***
Enter your guess: A
1. Way behind! Please try again.
Enter your guess: B
2. Way behind! Please try again.
Enter your guess: C
3. Way behind! Please try again.
Enter your guess: X
4. Way ahead! Please try again.
Enter your guess: Z
5. Way ahead! Please try again.
Enter your guess: f
6. Great! Your guess is right!
You should be able to do better!
```

Lowercase letters are also accepted and turned into uppercase.

5.35) *Fibonacci*

Code

```c
#include <stdio.h>

unsigned long long int fibonacci(unsigned int);

int main (void)
{
    unsigned int n;
    // prompt for user input
    printf("Enter the n: ");
    scanf("%u", &n);
    // display the number
    printf("%llu", fibonacci(n));

}// end function main

unsigned long long int fibonacci(unsigned int n)
{
    unsigned long long int x0 = 0; // 1st term
    unsigned long long int x1 = 1; // 2nd term
    unsigned long long int x2; // next term

    // loop until nth term
    for (unsigned int i = 2; i <= n; i++)
    {
        x2 = x0 + x1;
        x0 = x1;
        x1 = x2;
    }

    return x0;
} // end function fibonacci
```

Output

- The n^{th} Fibonacci number is calculated stating from 1.

```
Enter the n: 1
0
```

```
Enter the n: 2
1
```

```
Enter the n: 3
1
```

```
Enter the n: 4
2
```

- The largest Fibonacci number that I could have print was 94^{th} term 12200160415121876738 with 20 digits. After that it started to make no sense.

```
Enter the n: 94
12200160415121876738
```

5.39) *Recursive Greatest Common Divisor*

Code

```c
#include <stdio.h>

int gcd(int x, int y);

int main()
{
    // initialize variables
    int x, y;

    // prompt for user input x and y
    printf("Enter integer x: ");
    scanf("%d", &x);

    printf("Enter integer y: ");
    scanf("%d", &y);
    printf("_____________________\n");
    // call the gcd function and display result
    printf("GCD is %d for %d and %d.", gcd(x, y), x,y);

} // end function main
// recursive function to return gcd of x and y
int gcd(int x, int y)
{
    // if one of the integers is 0
    if (x == 0) {
       return y;
    }
    if (y == 0) {
       return x;
    }
    // base case
    if (x == y)
        return x;

    //if x is greater
    if (x > y) {
        return gcd(x-y, y); // recursive case
    }
    return gcd(x, y-x); // recursive case

} // end function gcd
```

Output

```
Enter integer x: 10
Enter integer y: 6
_____________________
GCD is 2 for 10 and 6.
```

- Returns gcd successfully.

```
Enter integer x: 10
Enter integer y: 0
_____________________
GCD is 10 for 10 and 0.
```

- If y = 0 GCD equals to x.