COMPUTER SCIENCE LEVEL 4 – LABORATORY ASSIGNMENT 1

Hamza Bhatti (21223241)

Contents

Introduction

This logbooks contains a large number of code that have been written using Java, C and Scheme programming languages.

The sections of this document are split up into the exercises that had been completed using the three different languages. The code is presented with the original formatting, along with examples of the outputs that can be produced when the programs have been run.

Exercise 1

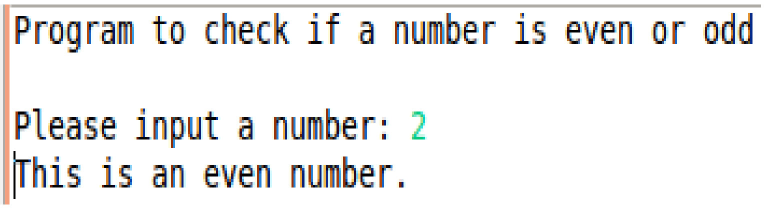
Program to check if a number is even or odd.

Java

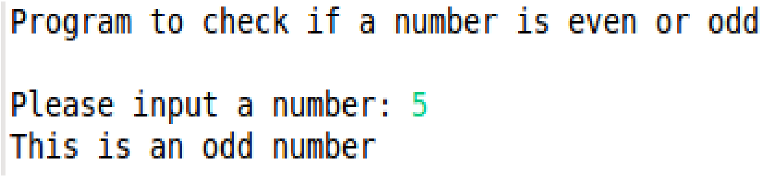
Code:

```
package week6;

public class Week6_Ex1
{
    public static void main(String[] args)
    {
        System.out.println("Program to check if a number is even or odd\n");

        //User is asked to enter a number
        int a = Console.askInt("Please input a number: ");

        //If number is not divisible by 2 it is odd
        if (a %2 != 0)
        {
            System.out.println("This is an odd number");
        }
        //Otherwise the number is an even number
        else
        {
            System.out.println("This is an even number.");
        }
    }
}
```

Output:



```
Program to check if a number is even or odd

Please input a number: 2
This is an even number.
```



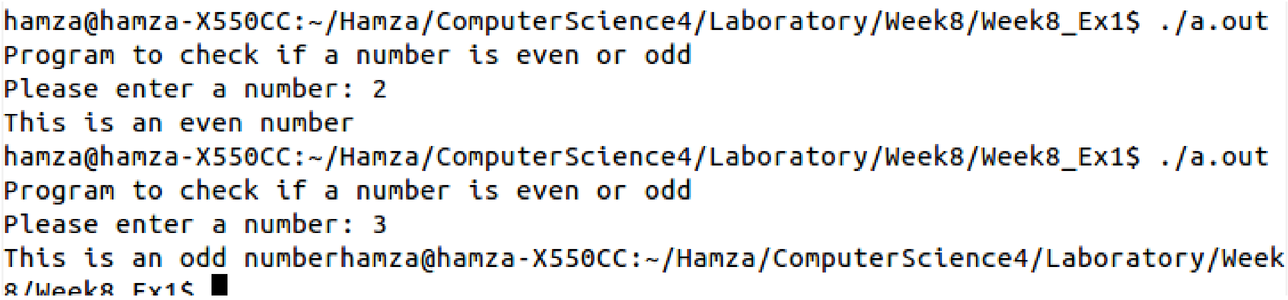
```
Program to check if a number is even or odd

Please input a number: 5
This is an odd number
```

C

Code:

```
#include <stdio.h>

int main()
{
 printf("Program to check if a number is even or odd\n");
 int number;

 //User is asked to enter a number
 printf("Please enter a number: ");
 //Number is stored in the container number
 scanf("%d", &number);

 //If the number when divisible by 2 does not give a 0, it is odd
 if (number %2 !=0)
 {
  printf("This is an odd number");
 }
 //Otherwise the number is even
 else
 {
  printf("This is an even number\n");
 }
 return 0;
}
```

Output:



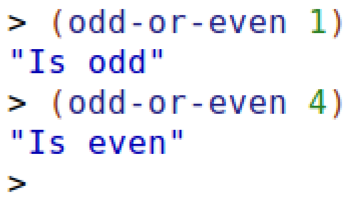
```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex1$ ./a.out
Program to check if a number is even or odd
Please enter a number: 2
This is an even number
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex1$ ./a.out
Program to check if a number is even or odd
Please enter a number: 3
This is an odd numberhamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week
8/Week8_Ex1$ 
```

Scheme

Code:

```
#lang racket

;;;Program to check if a number is even or odd
;;WORKS

; Creating a function called odd-or-even and it accepts
; a variable n
(define(odd-or-even n)
  ;Setting a condition where..
  (cond
    ; If the number is not divisible by 2 (no remainder)
    ; it is and odd number, else its even
    [(not(= (modulo n 2) 0))"Is odd"]
    [else "Is even"]))
```

Output:



```
> (odd-or-even 1)
"Is odd"
> (odd-or-even 4)
"Is even"
>
```

Exercise 2

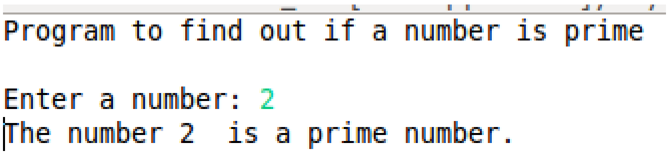
Program to find out if a number is prime.

Java

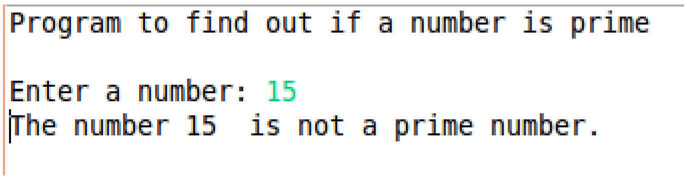
Code:

```
package week6;

public class Week6_Ex2
{
    public static void main(String[] args)
   {
         System.out.println("Program to find out if a number is prime\n");

         //User is asked for a number
        int number = Console.askInt("Enter a number: ");

        //Declaring a string called message to used later
        String message;

        //If statement which uses a function called isPrime and uses our variable number as parameter
        if (isPrime(number))
        {    //If the function returned a true value, number was in fact a prime
           message = " is a prime number.";
        }
        //Otherwise, if the function gave a false value
        else
        {    //The number variable was not a prime number
           message = " is not a prime number.";
        }

        System.out.println("The number " + number +" "+ message);
    }

    //Function called isPrime which accepts an int value (in this case the variable number)
    public static boolean isPrime(int n)
    {
         //If n / number is greater or equal to 1
         if (n <= 1)
        {    //return false as 1 is not a prime
             return false;
        }
         //When the new variable i = 2, but i is greater than the method Math.sqrt which takes our number
         // add 1 to i
        for (int i = 2; i < Math.sqrt(n); i++)
        {    //If n/number is divisible by i = 2 (at the start) and gives 0
         if (n % i == 0)
         {   //It is not prime
             return false;
           }
        }
         //Otherwise it is a prime
         return true;
         // Exit the function and return to the top
     }
}
```

Output:



```
Program to find out if a number is prime

Enter a number: 2
The number 2  is a prime number.
```



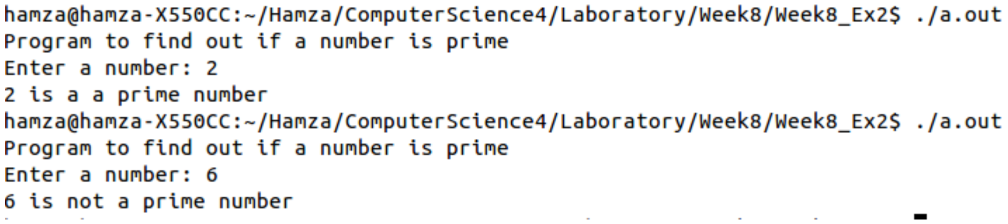
```
Program to find out if a number is prime

Enter a number: 15
The number 15  is not a prime number.
```

C

Code:

```
#include <stdio.h>

int main()
{
 printf("Program to find out if a number is prime\n");

 int number , i;
 int indicator = 0;

 //User enters a number. The number is stored in the container number
 printf("Enter a number: ");
 scanf("%d" ,&number);

 //The for loop checks if number PERFECTLY is divisible by 2 onwards,
 // if it is not perfectly divisible until i<=number/2, the number
 // is prime
 for(i=2; i<=number/2 ; ++i)
   {
     if(number%i == 0)
    { //If number is divisible by 2 the indicator will be 1
      indicator = 1;
      break;
    }
   }
 //If the indicator is 0 (not 1) the number entered is prime
 if(indicator == 0)
   {
     printf("%d is a a prime number\n" ,number);
   }
 //Otherwise the indicator was 1 and the number is not a prime number
 else
   {
     printf("%d is not a prime number\n" ,number);
   }
 return 0;
}
```

Output:



```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex2$ ./a.out
Program to find out if a number is prime
Enter a number: 2
2 is a a prime number
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex2$ ./a.out
Program to find out if a number is prime
Enter a number: 6
6 is not a prime number
```

Exercise 3

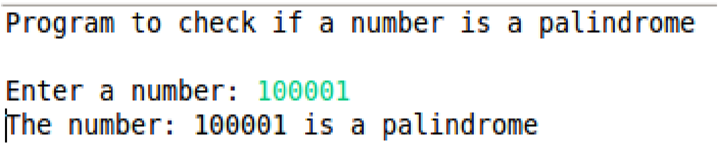
Program to check if a number is a palindrome.

Java

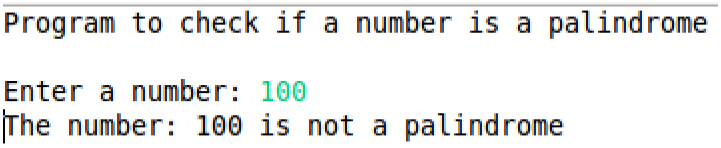
Code:

```
package week6;

public class Week6_Ex3
{
    // check if number is palindrome
    public static void main(String[] args)
    {
        System.out.println("Program to check if a number is a palindrome\n");

        //User is enters a number
        int number = Console.askInt("Enter a number: ");

        //Number is converted to string
        String number1 = String.valueOf(number);

        //Reverse number1 with >> (new StringBuilder(part).reverse().toString());
        String reverse = new StringBuilder(number1).reverse().toString();

        //The reversed string is now converted into an integer
        int reverse1 = Integer.parseInt(reverse);

        //Comparing original variable number1 to reverse1(which are both integers
        //If number1 is equal to reverse1 then it is a palindrome
        if (number == reverse1)
        {
            System.out.println("The number: " + number + " is a palindrome");
        }
        //Or else its not a palindrome
        else
        {
            System.out.println("The number: "+ number + " is not a palindrome");
        }
    }
}
```

Output:



```
Program to check if a number is a palindrome

Enter a number: 100001
The number: 100001 is a palindrome
```



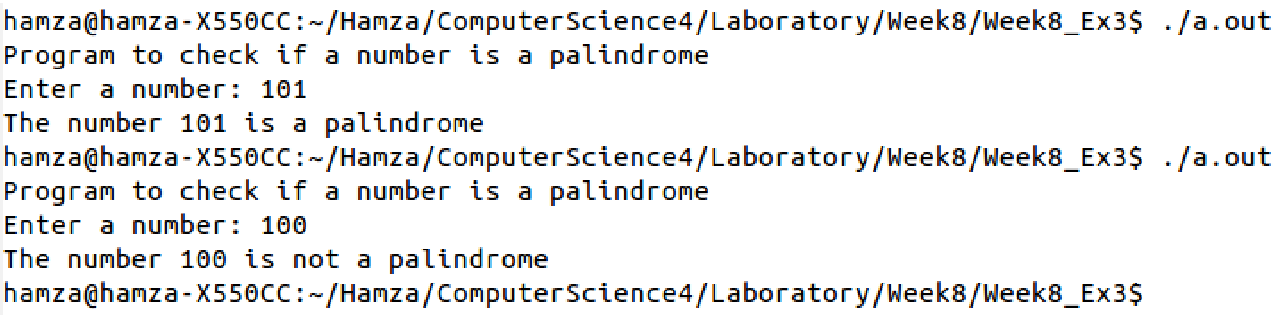
```
Program to check if a number is a palindrome

Enter a number: 100
The number: 100 is not a palindrome
```

C

Code:

```
#include <stdio.h>

int main()
{
 printf("Program to check if a number is a palindrome\n");

 //Declaring variables
 int number,  reverse, numberv2 ;

 //Asking user to input a number
 printf("Enter a number: ");
 //Number is stored in container number
 scanf("%d", &number);

 numberv2 = number;
 reverse = 0;

 //Reversing the number inside a while loop
 while (numberv2 != 0)
 { // When reverse = 0 use base of 10
  reverse = reverse * 10;
  //Using %10 to obtain length of number and reverse it
  reverse = reverse + numberv2%10;
  numberv2 = numberv2/10;
 }

 //If the number is equal to the reverse
 if (number == reverse)
 { //It is a palindrome
  printf("The number %d is a palindrome\n", number);
 }
 //Otherwise if number and reverse are not equal
 else if (number != reverse)
 { //The number is not a palindrom
  printf("The number %d is not a palindrome\n", number);
 }

 return 0;
}
```

Output:



```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex3$ ./a.out
Program to check if a number is a palindrome
Enter a number: 101
The number 101 is a palindrome
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex3$ ./a.out
Program to check if a number is a palindrome
Enter a number: 100
The number 100 is not a palindrome
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex3$
```

Exercise 4

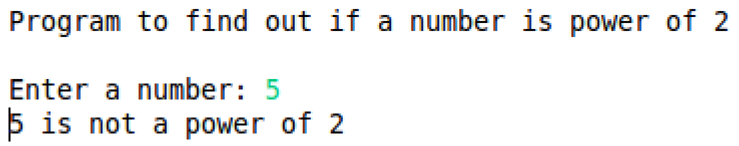
Program to find out if a number is power of 2.

Java

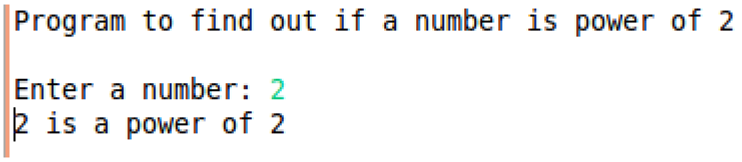
Code:

```java
package week6;

public class Week6_Ex4
{
   public static void main(String[] args)
   {
    System.out.println("Program to find out if a number is power of 2\n");

    //User inputs a number
    int number = Console.askInt("Enter a number: ");

    //Checking if the number is a power of 2
    //If the number variable and the number -1 eqaual to 0, then number is a power of 2
    if ((number & (number - 1 )) == 0)
    {
        System.out.println(number + " is a power of 2");
    }
    //Or else it is not a power of 2
    else
    {
        System.out.println(number + " is not a power of 2");
    }
   }
}
```

Output:



```
Program to find out if a number is power of 2

Enter a number: 5
5 is not a power of 2
```



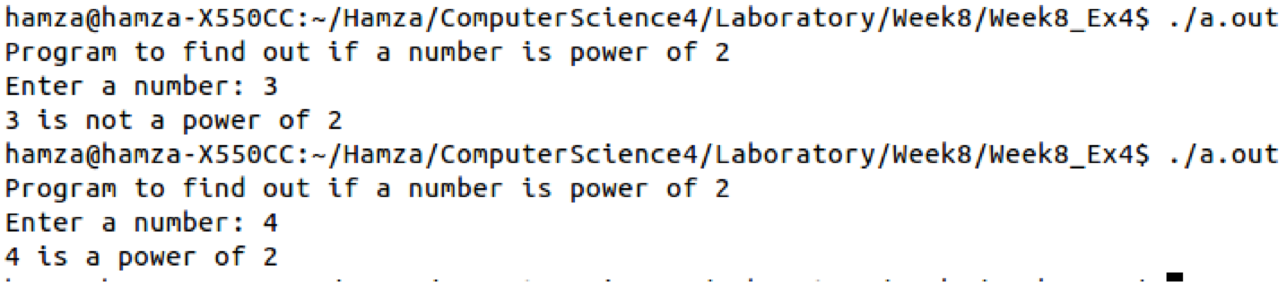
```
Program to find out if a number is power of 2

Enter a number: 2
2 is a power of 2
```

C

Code:

```
#include <stdio.h>

int main()
{
 printf("Program to find out if a number is power of 2\n");

 int number;

 //User is asked to enter a number
 printf("Enter a number: ");
 //Number is stored in the container number
 scanf("%d", &number);

 //If the number and the number-1 is equal to 0 it is a power of 2
 if ((number & (number - 1))== 0)
 {
  printf("%d is a power of 2\n" ,number);
 }
 //Other wise it is not a power of 2
 else
 {
  printf("%d is not a power of 2\n" ,number);
 }
 return 0;
}
```

Output:



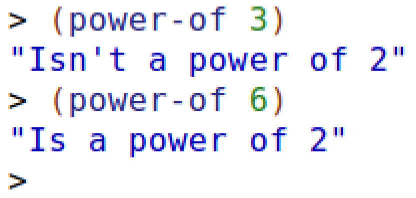
```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex4$ ./a.out
Program to find out if a number is power of 2
Enter a number: 3
3 is not a power of 2
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex4$ ./a.out
Program to find out if a number is power of 2
Enter a number: 4
4 is a power of 2
```

Scheme

Code:

```
#lang racket

;;; Program to find if a number is power of 2
;;WORKS

(define (power-of num)
 (cond
   ;;If the entered number has a remaider of 0 when divided by
   ;; 2, it will be a power of 2
   [(=(modulo num 2)0) "Is a power of 2"]
   ;;Otherwise it isnt a power of 2
   [else "Isn't a power of 2"]))
```

Output:



```
> (power-of 3)
"Isn't a power of 2"
> (power-of 6)
"Is a power of 2"
>
```

Exercise 5

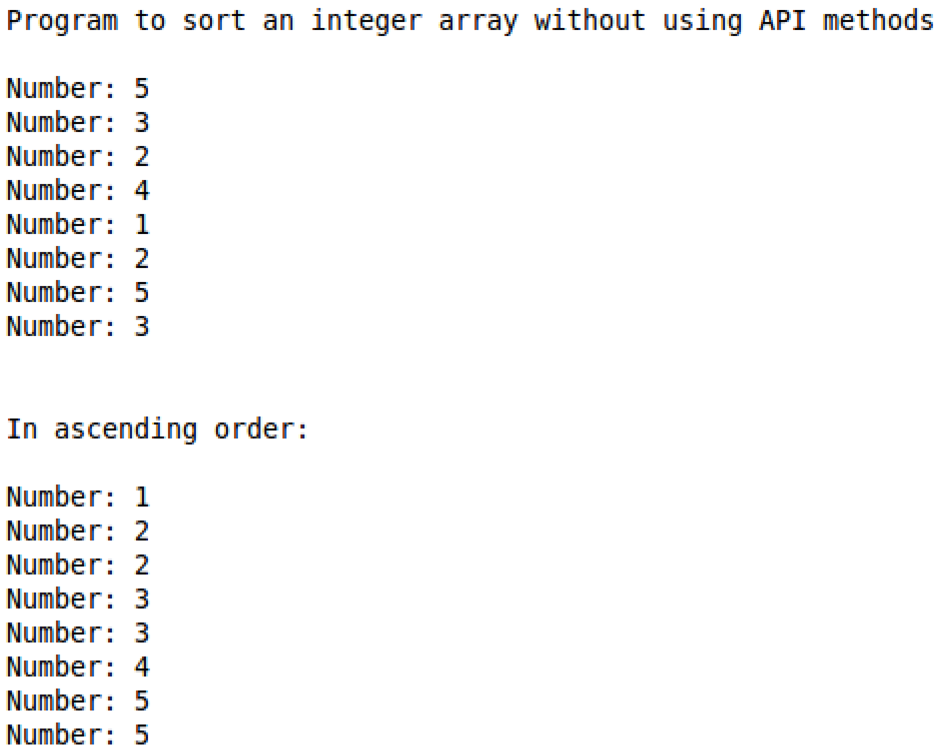
Program to sort an integer array without using API methods.

Java

Code:

```
package week6;

import java.util.Arrays;
public class Week6_Ex5
{
    public static void main(String[] args)
    {
        System.out.println("Program to sort an integer array without using API methods\n");

        //An array is created with these numbers
        int array[] = {5, 3, 2, 4, 1, 2, 5, 3};

        //For all the numbers in the array
        for (int number : array)
        {   //Print out the number
            System.out.println("Number: " + number);
        }
        System.out.println("\n"); //New line

        //Using the sort method for all the numbers in the array
        Arrays.sort(array);
        System.out.println("In ascending order: \n");
        //For all the numbers in the array
        for (int number : array)
        {   //Print them out, but now in ascending order
            System.out.println("Number: " + number);
        }

    }
}
```

Output:



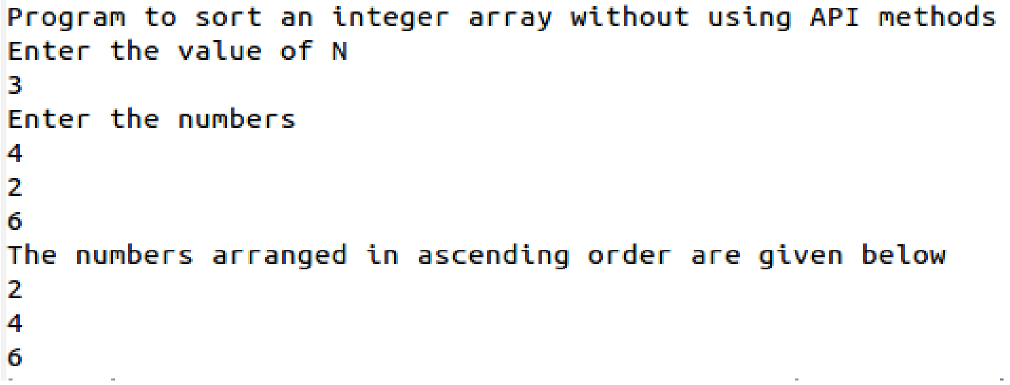
```
Program to sort an integer array without using API methods

Number: 5
Number: 3
Number: 2
Number: 4
Number: 1
Number: 2
Number: 5
Number: 3


In ascending order:

Number: 1
Number: 2
Number: 2
Number: 3
Number: 3
Number: 4
Number: 5
Number: 5
```

C

Code:

```
#include <stdio.h>

int main()
{
 printf("Program to sort an integer array without using API methods\n");
  int i, j, a, n, number[30];

  //User asked how many values they want to enter
  printf("Enter the value of N \n");
  //Input stored in container n
  scanf("%d", &n);

  //Another prompt, user enters values up till the value they stated earlier
  printf("Enter the numbers \n");
  //When i = 0 and when i is less than n, increment by 1
  for (i = 0; i < n; ++i)
  {
   //numbers get stored in the array number
   //if user wants 3 number, count will start from 0 - 1 - 2
    scanf("%d", &number[i]);
  }

  //When i = 0 up to it being less than n(does not exceed n) increment by 1
  for (i = 0; i < n; ++i)
  {
    for (j = i + 1; j < n; ++j)
    {
      if (number[i] > number[j])
      {
        a =  number[i];
        number[i] = number[j];
        number[j] = a;
      }
    }
  }
  //Prints out all the numbers that had been arranged in the last for loop
  printf("The numbers arranged in ascending order are given below \n");
  for (i = 0; i < n; ++i)
    printf("%d\n", number[i]);
}
```

Output:



```
Program to sort an integer array without using API methods
Enter the value of N
3
Enter the numbers
4
2
6
The numbers arranged in ascending order are given below
2
4
6
```

Exercise 6

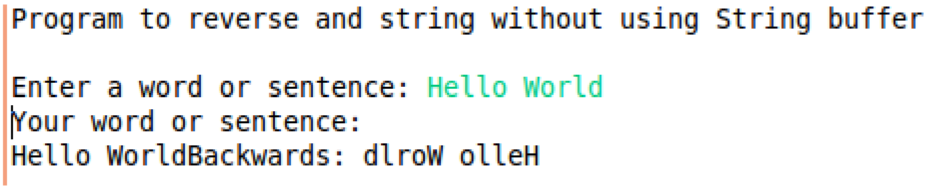
Program to reverse and string without using String buffer.

Java

Code:

```java
package week6;

public class Week6_Ex6
{
   public static void main(String[] args)
   {
    System.out.println("Program to reverse and string without using String buffer\n");

    //Ask for a string input
    String word = Console.askString("Enter a word or sentece: ");

    //Word is revered and then printed (same method used previously in Week6_Ex3)
    String reverse = new StringBuilder(word).reverse().toString();
    System.out.println("Your word or sentence: \n" + word + "Backwards: " + reverse);
   }
}
```

Output:



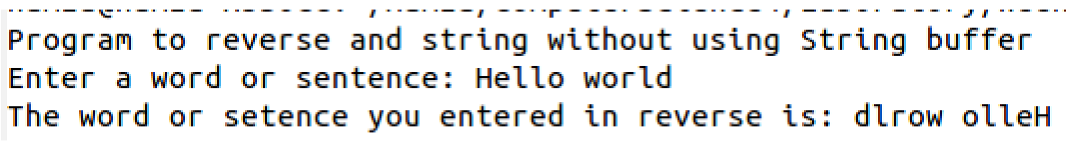
```
Program to reverse and string without using String buffer

Enter a word or sentence: Hello World
Your word or sentence:
Hello WorldBackwards: dlroW olleH
```

C

Code:

```c
#include<stdio.h>
#include<string.h>

int main()
{
  printf("Program to reverse and string without using String buffer\n");

  char string[100], temp;
  int i, j = 0;

  //User enters a string of words and is stored in the array
  printf("Enter a word or sentence: ");
  gets(string);

  //strlen identifies the length of the string -1
  // eg. if word is Hello (there are 5 chars, but the count needs
  // to start at 0, so 1 is taken away
  i = 0;
  j = strlen(string) - 1;

  while(i < j)
   { // i is the begining char and j is the last char
     //Cycling through chars from the begining to end
     temp = string[i];
     string[i] = string[j];
     string[j] = temp;
     i++; //Counting forwards
     j--; //Counting backwards
   }//Loop ends when i and j meet

  //Printing out the reverse
  printf("The word or setence you entered in reverse is: %s\n", string);
  return 0;
}
```

Output:



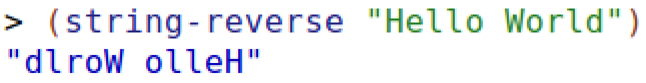
```
Program to reverse and string without using String buffer
Enter a word or sentence: Hello world
The word or setence you entered in reverse is: dlrow olleH
```

Scheme

Code:

```
#lang racket

;;;Program to reverse string
;;WORKS?

(define (string-reverse s)
  (list->string (reverse (string->list s))))
```

Output:



```
> (string-reverse "Hello World")
"dlroW olleH"
```

Exercise 7

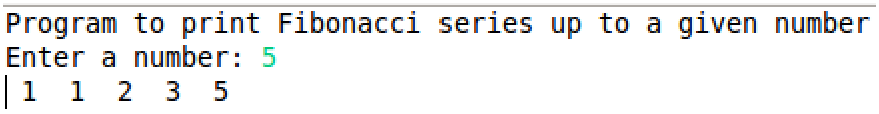
Program to print Fibonacci series up to a given number.

Java

Code:

```
package week6;

public class Week6_Ex7
{
    public static void main(String args[])
    {
        System.out.println("Program to print Fibonacci series up to a given number");

        //Asking the user to input they want the Fibonacci to end at
        int num = Console.askInt("Enter a number: ");
        //Declaring variables
        int f1;
        int f2 = 0;
        int f3 = 1;

        //
        //When i = 1 and until i is less or equal to the number it will end at, add 1
        //i acts as a count
        for(int i=1;i<=num;i++)
        {
            System.out.print(" "+f3+" ");
            f1 = f2; //f1 will start as 0
            f2 = f3; //f2 is now the same as f3
            f3 = f1 + f2; //f3 = 0 + 1
            //Continue until last number matches the input
        }
    }
}
```

Output:



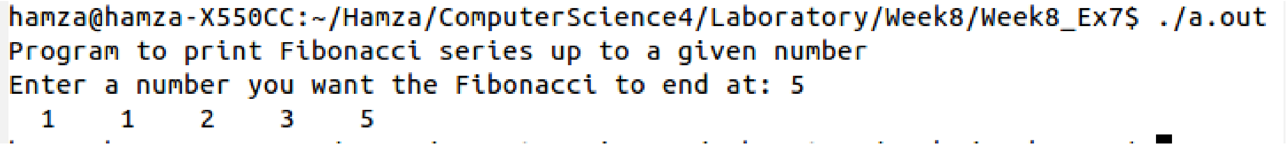
```
Program to print Fibonacci series up to a given number
Enter a number: 5
 1  1  2  3  5
```

C

Code:

```c
#include <stdio.h>

int main()
{
  printf("Program to print Fibonacci series up to a given number\n");

  int number , i , f1 , f2 , f3;

  //User asked for the amount of number he wants the fibonacci to end at
  //Input stored in container number
  printf("Enter a number you want the Fibonacci to end at: ");
  scanf("%d", &number);

  f2 = 0;
  f3 = 1;

  //When i = 1 until i less than or equal to number, increment by 1
  for (i = 1 ; i<= number ; ++i)
   {
     //Start by print f3 = 1
     printf("  %d  ",f3);
     //Starting: f1 = f2 meaning f1 = 0
     f1 = f2;
     //f2 will then  equal 1 as f3 = 1
     f2 = f3;
     // f3 (number to be outputted) will be to start off as 1 + 0 = 1
     f3 = f1 + f2;
     //Loop will increment up till i = number
   }
  printf("\n");
  return 0;
}
```

Output:



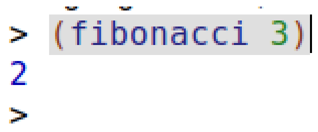
```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex7$ ./a.out
Program to print Fibonacci series up to a given number
Enter a number you want the Fibonacci to end at: 5
  1    1    2    3    5
```

Scheme

Code:

```
#lang racket
;;;Program to print Fibonacci series up to given number
;WORKS sort of, recursive version which prints the nth number

(define (fibonacci num)
  ;If number is less then or = 2, display 1
  (if (<= num 2)
      1
      ;Otherwise fibonacci starts with num - 2
      ; then add that to fibonacci = num - 1
      ; until it is num is reached
      (+ (fibonacci (- num 1)) (fibonacci (- num 2)))))
```

Output:



```
> (fibonacci 3)
2
>
```

Exercise 8

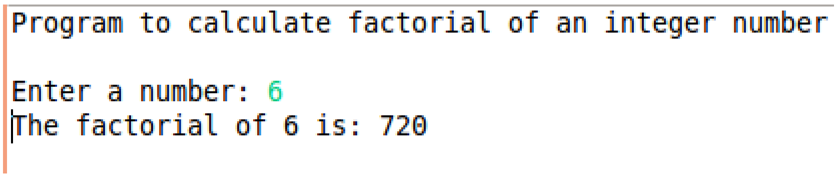
Program to calculate factorial of an integer number.

Java

Code:

```
package week6;

public class Week6_Ex8
{
    public static void main(String[] args)
    {
        System.out.println("Program to calculate factorial of an integer number\n");

        //User enters a number
        int number = Console.askInt("Enter a number: ");

        //Declaring a variable fact for later use
        int fact = 1;

        //When c = 1, until c is less than or equal to the input, add 1 to c
        for (int c = 1 ; c <= number; c++)
        {
            fact = fact*c; // 1* 1, c will increment up 1 every time it loops until c is less or = to input
        }
        System.out.println("The factorial of " + number + " is: " + fact);
    }
}
```

Output:



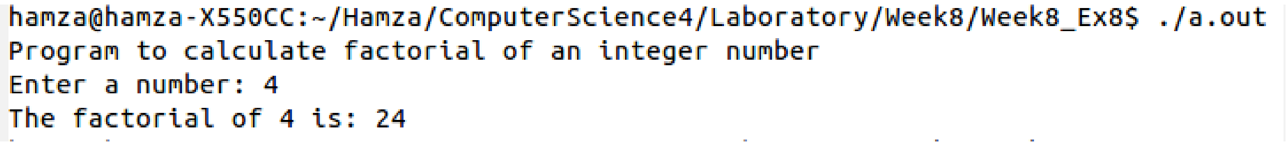
```
Program to calculate factorial of an integer number

Enter a number: 6
The factorial of 6 is: 720
```

C

Code:

```
#include <stdio.h>

int main()
{
 printf("Program to calculate factorial of an integer number\n");

 int number, i , fact = 1;

 //User inputs a number they want the factorial for
 printf("Enter a number: ");
 //number stored in container "number"
 scanf("%d", &number);

 //When i is 1, until i is less than or equal to number, increment by 1
 for (i = 1 ; i<= number; i++)
 { //fact will equal 1*1 (i=1) and will increment
  fact = fact*i;
 }

 //Result printed out
 printf("The factorial of %d is: %d\n" , number , fact);

 return 0;
}
```

Output:



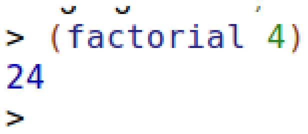
```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex8$ ./a.out
Program to calculate factorial of an integer number
Enter a number: 4
The factorial of 4 is: 24
```

Scheme

Code:

```
#lang racket

;;; Program to calculate factorial of an integer number
;;WORKS

(define (factorial n)
 ;;If the input is 0, the factorail will be 1, as a rule
   (if (= n 0)
      1
     ;;The inputed number - 1, then multipled by number
     ;; the function is called again with the new value and
     ;; loops until the n = 0
     (* n (factorial (- n 1)))))
```

Output:



```
> (factorial 4)
24
>
```

Exercise 9

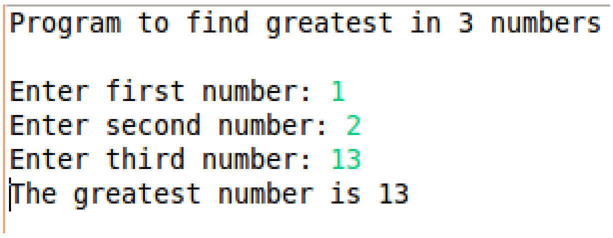
Program to find greatest in 3 numbers.

Java

Code:

```
package week6;

public class Week6_Ex9
{
    public static void main(String[] args)
    {
        System.out.println("Program to find greatest in 3 numbers\n");

        //User enters 3 values
        int num1 = Console.askInt("Enter first number: ");
        int num2 = Console.askInt("Enter second number: ");
        int num3 = Console.askInt("Enter third number: ");

        //If statements to check which is the greatest out of the three
        //If num1 is greater then num2 and greater than num2 and num3
        if (num1 > num2 && num1 > num3)
        {   //Print out num1
            System.out.println("The greatest number is " + num1);
        }
        //Else if num2 is greater than num1 and num2
        else if (num2 > num1 && num2 > num3)
        {   //Print out num2
            System.out.println("The greatest number is " + num2);
        }
        //Otherwise num3 is the greatest out of the three numbers
        else
        {
            System.out.println("The greatest number is " + num3);
        }
    }
}
```

Output:



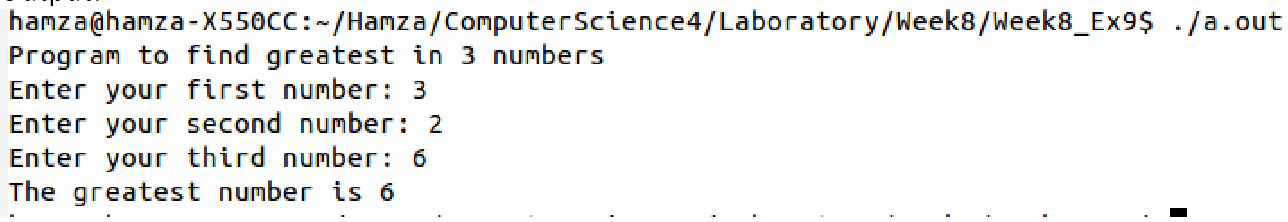
```
Program to find greatest in 3 numbers

Enter first number: 1
Enter second number: 2
Enter third number: 13
The greatest number is 13
```

C

Code:

```
#include <stdio.h>

int main()
{
  printf("Program to find greatest in 3 numbers\n");

  int number1, number2, number3;

  //User enters three number, numbers are then stored in containers
  printf("Enter your first number: ");
  scanf("%d", &number1);
  printf("Enter your second number: ");
  scanf("%d", &number2);
  printf("Enter your third number: ");
  scanf("%d", &number3);

  //If number 1 greater than number2 and 3, print number1
  if(number1 > number2 && number1 > number3)
  {
    printf("The greatest number is %d\n", number1);
  }
  //If number 2 is greater than 1 and 3, print number2
  else if(number2 > number1 && number2 > number3)
  {
    printf("The greatest number is %d\n", number2);
  }
  //Or else 3 is the greatest, and print it
  else
  {
    printf("The greatest number is %d\n", number3);
  }
  return 0;
}
```

Output:



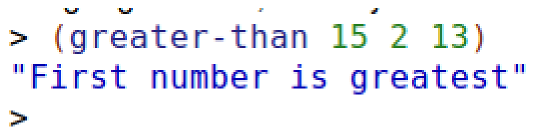
```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_Ex9$ ./a.out
Program to find greatest in 3 numbers
Enter your first number: 3
Enter your second number: 2
Enter your third number: 6
The greatest number is 6
```

Scheme

Code:

```
#lang racket

;;;Program to find the greatest out of 3 numbers
;;WORKS

(define(greater-than a b c)
  (cond
    [ (and(> a b)(> a c)) "First number is greatest"]
    [ (and(> b a)(> b c)) "Second number is greatest"]
    [ else "Third number is greatest"]))
```

Output:



```
> (greater-than 15 2 13)
"First number is greatest"
>
```

Exercise 10

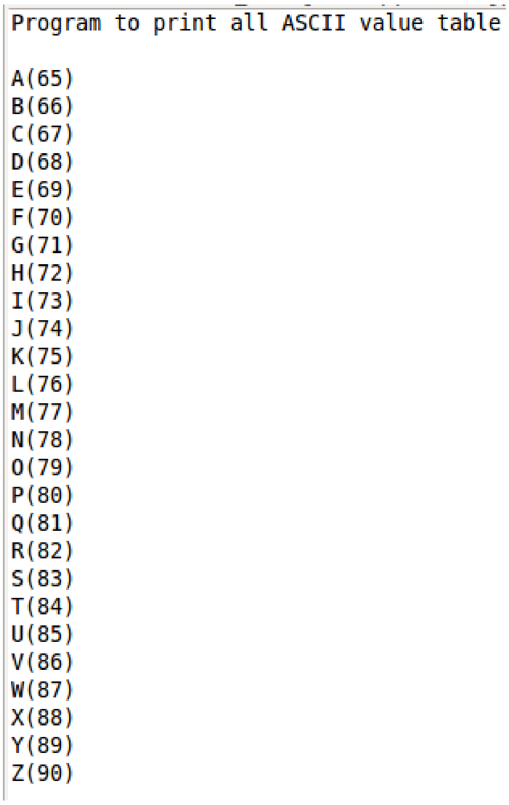
Program to print all ASCII value table.

Java

Code:

```
package week6;

public class Week6_Ex10
{
    public static void main(String[] args)
    {
        System.out.println("Program to print all ASCII value table\n");
        {
            //The first character
            char begin = 'A';
            //The last character
            char end = 'Z';
            //Empty character
            char i = ' ';

            //When the i = the beginning (A), but when until i is greater or equal to end (Z)
            // Add 1 to i
            for (i=begin; i<=end; i++)
            {   //Declaring asciivalue as an integer that converts i(which was a character) into and int val
                int asciiValue = (int)i;

                //And print out the value and loop back
                System.out.println(i + "(" + asciiValue + ")");
            }
        }
    }
}
```

Output:



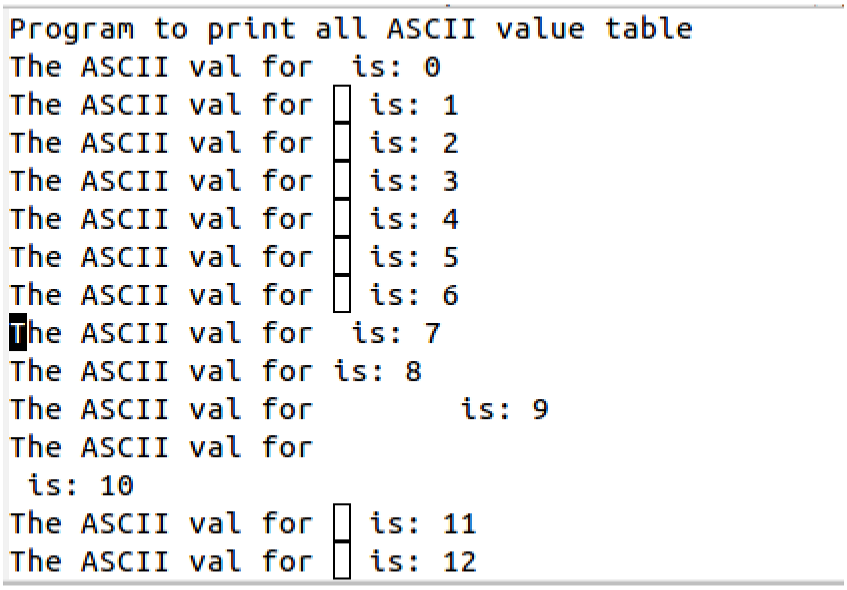
```
Program to print all ASCII value table

A(65)
B(66)
C(67)
D(68)
E(69)
F(70)
G(71)
H(72)
I(73)
J(74)
K(75)
L(76)
M(77)
N(78)
O(79)
P(80)
Q(81)
R(82)
S(83)
T(84)
U(85)
V(86)
W(87)
X(88)
Y(89)
Z(90)
```

C

Code:

```
#include <stdio.h>

int main()
{
  printf("Program to print all ASCII value table\n");

  int i;

  //When i = 0 and until i is less than or equal to 255, increment i by 1
  for(i = 0; i<= 255; i++)
  {
    //Use of %c shows coresponding ascii value for that int
    printf("The ASCII val for %c is: %d\n", i , i);
  }
  return 0;
}
```

Output:



```
Program to print all ASCII value table
The ASCII val for  is: 0
The ASCII val for □ is: 1
The ASCII val for □ is: 2
The ASCII val for □ is: 3
The ASCII val for □ is: 4
The ASCII val for □ is: 5
The ASCII val for □ is: 6
The ASCII val for  is: 7
The ASCII val for is: 8
The ASCII val for        is: 9
The ASCII val for
 is: 10
The ASCII val for □ is: 11
The ASCII val for □ is: 12
```

Language Specific Exercises

The programs in this section where tasks which kept the functionality of the specific languages used in mind.

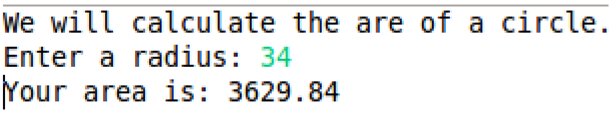
Java

Area of Circle

Code:

```
package week7;

public class Week7_Ex1
{
    public static void main(String[] args)
    {
        System.out.println("We will calculate the are of a circle.");
        int radius = Console.askInt("Enter a radius: ");
        double Pi = 3.14;

        double area = radius * radius * Pi;

        System.out.println("Your area is: " +area );
    }
}
```

Output:



```
We will calculate the are of a circle.
Enter a radius: 34
Your area is: 3629.84
```

Student application

Code:

Default constructor

```
package week7;

public class Student
{
    // Declaring variables
    String firstName;
    String secondName;
    int studentId;
    String course;
    int level;
    int startDate;
    int endDate;

    // Set method
    public void set(String aFirstName , String aSecondName , int aStudentId ,
            String aCourse , int aLevel , int aStartDate , int aEndDate)
    {
        firstName = aFirstName;
        secondName = aSecondName;
        studentId = aStudentId;
        course = aCourse;
        level = aLevel;
        startDate = aStartDate;
        endDate = aEndDate;
    }

    // Using parameters to set data values to Student class
    public Student(String aFirstName , String aSecondName , int aStudentId ,
            String aCourse , int aLevel , int aStartDate , int aEndDate)
    {
        set (aFirstName , aSecondName , aStudentId , aCourse , aLevel , aStartDate , aEndDate);
    }

    // Creating default properties for the student class
    public Student()
    {
        set("Unknown" , "Unknown" , 000000, "Unknown" , 3 , 2014 , 2017 );
    }

    // Set individual data methods
    public void setFirstName(String aFirstName)
    {
        firstName = aFirstName;
    }
    public void setSecondName(String aSecondName)
    {
        secondName = aSecondName;
    }
    public void setStudenId(int aStudentId)
    {
        studentId = aStudentId;
    }
    public void setCourse(String aCourse)
    {
        course = aCourse;
    }
```

```java
    public void setLevel(int aLevel)
    {
        level = aLevel;
    }
    public void setStartDate(int aStartDate)
    {
        startDate = aStartDate;
    }
    public void setEndDate(int aEndDate)
    {
        endDate = aEndDate;
    }

    // Ask method
    public void ask (String prompt)
    {
        System.out.println("Make your own data entries\n");
        setFirstName(Console.askString("Enter a First name: "));
        setSecondName(Console.askString("Enter a Second name: "));
        setStudenId(Console.askInt("Enter the Student ID: "));
        setCourse(Console.askString("Enter the course name: "));
        setLevel(Console.askInt("Enter the course level: "));
        setStartDate(Console.askInt("Enter the start date: "));
        setEndDate(Console.askInt("Enter the end date: "));
    }

    public String toString()
    {
        return "First name: " + firstName + ", Second Name: " + secondName + "Student ID: " + studentId +
                ", Course: " + course + ", Level : " + level + ", Start Date: " + startDate
                + ", End Date: " + endDate;
    }

    // Print method
    public void print()
    {
        System.out.println("Student enrolment: \n");
        System.out.println("First name: " + firstName);
        System.out.println("Second Name: " + secondName);
        System.out.println("Student ID: " + studentId );
        System.out.println("Course: " + course );
        System.out.println("Level : " + level);
        System.out.println("Start Date: " + startDate);
        System.out.println("End Date: " + endDate);
    }
}
```

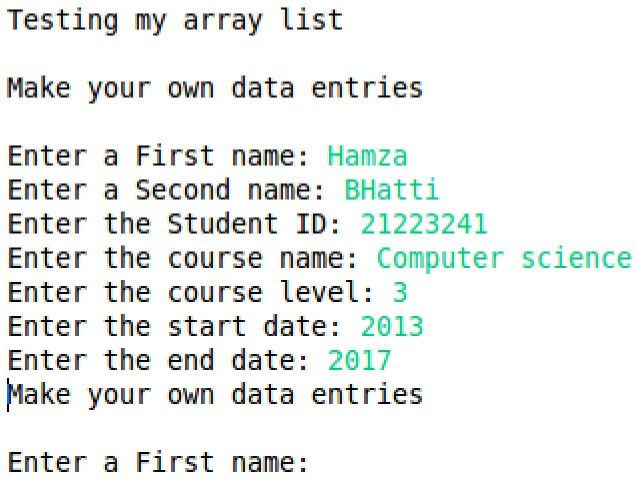
Array

```
package week7;

//importing an array list function
import java.util.ArrayList;

public class Students
{
    // Using an array to store info
    ArrayList<Student>Students;
    // Store all items from student into Students class array

    public Students()
    {
        Students = new ArrayList<Student>();
        //a container to hold contents from the Student class
    }

    // creating a method to check how many students have been enrolled
    public int getSize()
    {
        return Students.size();
    }
    //creating a method to add new item, or students to the array list
    public void add(Student aStudent)
    {
        Students.add(aStudent);
    }

    public void print(String header)
    {
        System.out.println(header);
        for (int i = 0; i < Students.size(); i++)
        System.out.println( Students.get(i)); // prints the numbers of entries in the array list
    }
}
```

Test

```
package week7;

public class Students_Test1
{
    public static void main(String[] args)
    {
        System.out.println("Testing my array list\n");

        Students Students = new Students();

        Student Student1 = new Student(); //create objects
        Student1.ask("Enter details");
        Students.add(Student1);

        Student Student2 = new Student();
        Student2.ask("Enter details");
        Students.add(Student2);

        Students.print("Entries to the array: \n");
        System.out.println(Students.getSize());
    }
}
```

Output



```
Testing my array list

Make your own data entries

Enter a First name: Hamza
Enter a Second name: BHatti
Enter the Student ID: 21223241
Enter the course name: Computer science
Enter the course level: 3
Enter the start date: 2013
Enter the end date: 2017
Make your own data entries

Enter a First name:
```

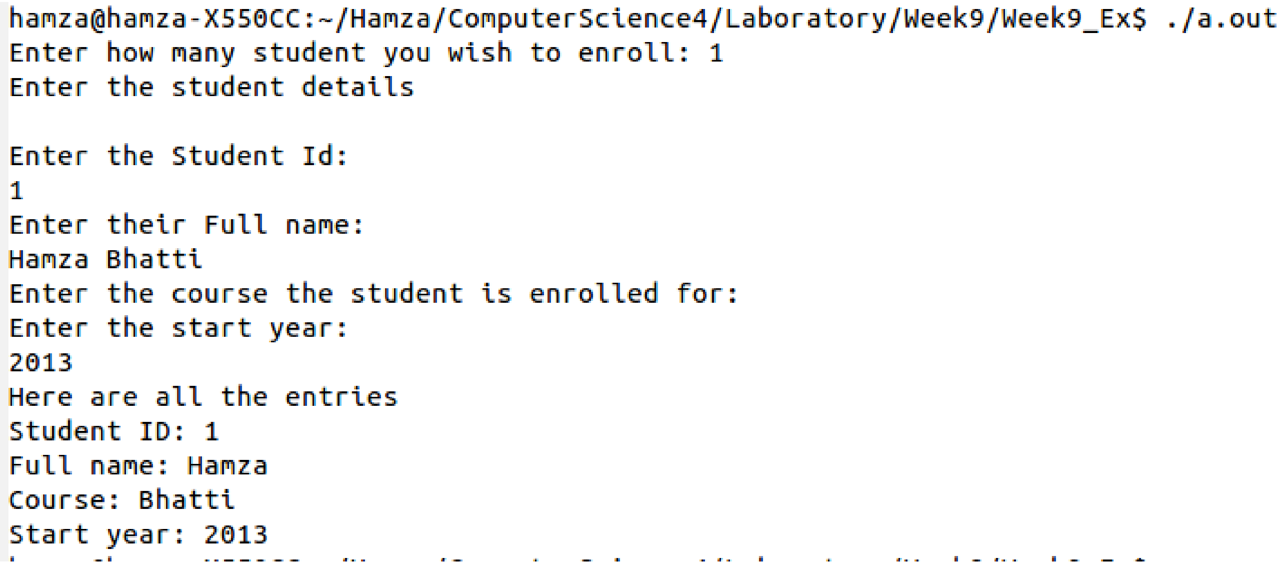
C

Student Program

Code:

```
#include <stdio.h>

//Creating a structure called student

struct Student{
  int studentId;
  char fullName[50];
  char course[100];
  int startYear;
  //int endYear;
};

int main()
{
  int n, i;

  printf("Enter how many student you wish to enroll: ");
  scanf("%d", &n);

  struct Student s[n];

  //Details for the students are entered and stored
  printf("Enter the student details\n");
  for (i = 0; i < n ; ++i)
  {
    printf("   \n");
    printf("Enter the Student Id: \n");
    scanf("%d", &s[i].studentId);

    printf("Enter their Full name: \n");
    scanf("%s", &s[i].fullName);

    printf("Enter the course the student is enrolled for: \n");
    scanf("%s", &s[i].course);

    printf("Enter the start year: \n");
    scanf("%d", &s[i].startYear);

    // s[i].endYear = s[i].startYear + 3;
  }

  //In every array space show that value then increment
  printf("Here are all the entries\n");
  for(i = 0; i < n ; i++)
  {
    printf("Student ID: %d\n", s[i].studentId);
    printf("Full name: %s\n", s[i].fullName);
    printf("Course: %s\n", s[i].course);
    printf("Start year: %d\n", s[i].startYear);
    //printf("End year: %d\n", endYear);
  }

  return 0;
}
```

Output:



```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week9/Week9_Ex$ ./a.out
Enter how many student you wish to enroll: 1
Enter the student details

Enter the Student Id:
1
Enter their Full name:
Hamza Bhatti
Enter the course the student is enrolled for:
Enter the start year:
2013
Here are all the entries
Student ID: 1
Full name: Hamza
Course: Bhatti
Start year: 2013
```

Scheme

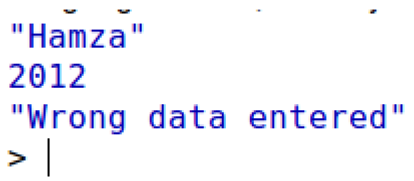
Student Program

Code:

```
#lang racket

;;;Program to enroll students

;Creating a database using a hash table
(define db (make-hash))

;New function taking an id and student number
(define (new-student id snum)
  ;Will insert them into a hash table
  (hash-set! db id snum))

;Creating a student with properties in a list
(define stud1
  '((first-name "Hamza") (second-name "Bhatti")
    (age 21) (course "Computer Science")
    (start-year 2013)))

;Creating second student
(define stud2
  '((first-name "David") (second-name "Brown")
    (age 25) (course "Drama")
    (start-year 2012)))

;Creating third student
(define stud3
  '((first-name "John") (second-name "Doe")
    (age 19) (course "Chemistry")
    (start-year 2010)))

;The new student will have an id of 1 and identified as stud1
(new-student 1 stud1)
(new-student 2 stud2)
(new-student 3 stud3)

;Repetition is removed
(define (find-student-v2 id field)
  ;Added loop. The record function will take the hashref
  ; database and id
  (let ((record (hash-ref db id)))
    ;When a field is entered, find the appropriate field
    (case field
      ((first-name) (cadr (assq field record)))
      ((second-name) (cadr (assq field record)))
      ((age) (cadr (assq field record)))
      ((course) (cadr (assq field record)))
      ((start-year) (cadr (assq field record)))
      ;Error handling
      (else "Wrong data entered"))))

(find-student-v2 1 'first-name)

(find-student-v2 2 'start-year)
;Wrong field is entered giving and error
```

```
(find-student-v2 3 ' star)
```

Output:



```
"Hamza"
2012
"Wrong data entered"
> 
```

HTML

Along with the tasks that involved solving problems with Java, C and Scheme, there was also HTML pages that need to be completed.

HTML LINK: http://soc.uwl.ac.uk/~21223241/

My HTML had 4 pages, Index, Home, Java, C and scheme. The pages consisted of CSS which was standard for all the pages.

The CSS consisted of a title banner which was a light blue colour with a solid blue border. Another feature was a side bar which contained the links to the other pages. This had a grey background colour. And finally a footer which was the same colour as the title banner.

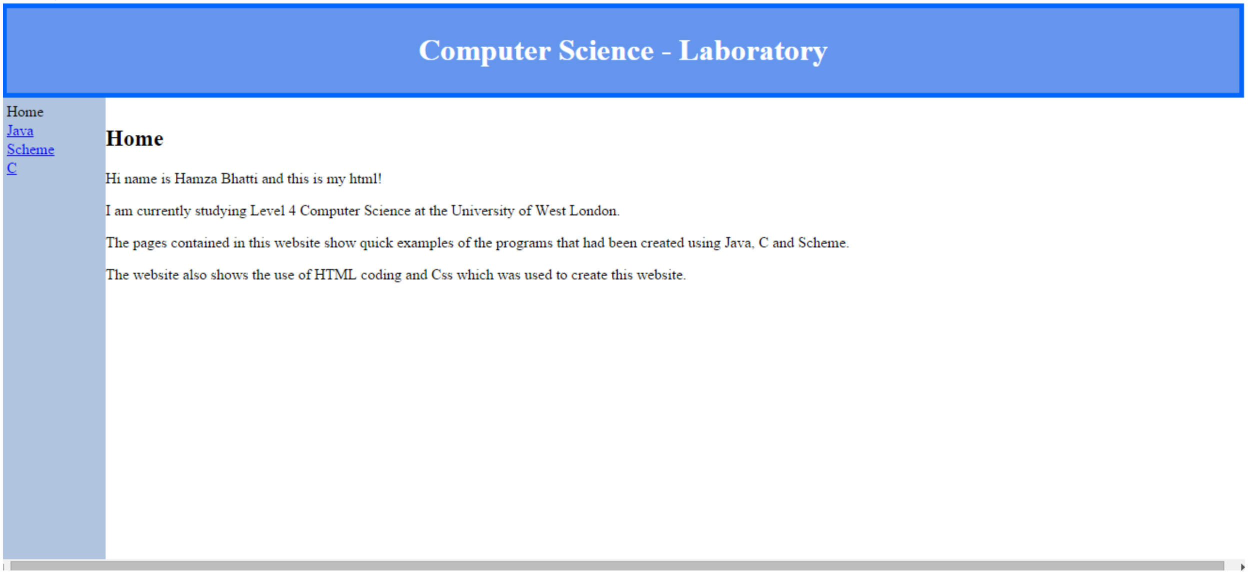
Index/ Home

Code:

```
<!DOCTYPE html>
<html>
<head>
<style>
header {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    text-align:center;
    padding:5px;
}
nav {
    line-height:20px;
    background-color:#b0c4de;
    height:500px;
    width:100px;
    float:left;
    padding:5px;
}
section {
    width: 100% ;
    float:centre;
    padding:10px;
}
footer {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    clear:both;
    text-align:center;
    padding:5px;
}

</style>
</head>

<body>

<header>
<h1>Computer Science - Laboratory</h1>
</header>

<nav>
Home<br>
```

```
<a href = "Hamza_html_Java.html">Java</a><br>
<a href = "Hamza_html_Scheme.html">Scheme</a><br>
<a href = "Hamza_html_C.html">C</a><br>
</nav>

<section>
<h1>Home</h1>
<p>Hi name is Hamza Bhatti and this is my html!</p>

<p>I am currently studying Level 4 Computer Science at the University of West
London.</p>

<p>The pages contained in this website show quick examples of the programs that
had been created using Java, C and Scheme.</p>

<p>The website also shows the use of HTML coding and Css which was used to
create this website.</p>

</section>

<footer>

</footer>

</body>
```

Output:



Computer Science - Laboratory

Home
Java
Scheme
C

Home

Hi name is Hamza Bhatti and this is my html!

I am currently studying Level 4 Computer Science at the University of West London.

The pages contained in this website show quick examples of the programs that had been created using Java, C and Scheme.

The website also shows the use of HTML coding and Css which was used to create this website.

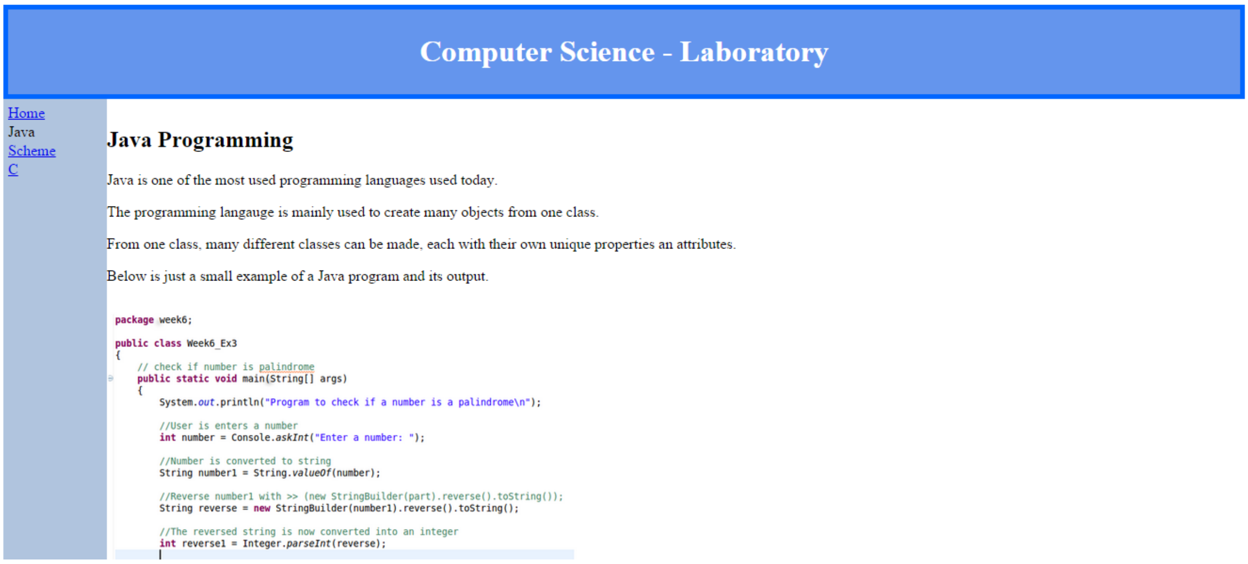
Java

Code:

```
<!DOCTYPE html>
<html>
<head>
<style>
header {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    text-align:center;
    padding:5px;
}
nav {
    line-height:20px;
    background-color:#b0c4de;
    height:500px;
    width:100px;
    float:left;
    padding:5px;
}
section {
    width: 100% ;
    float:centre;
    padding:10px;
}
footer {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    clear:both;
    text-align:center;
    padding:5px;
}

</style>
</head>

<body>

<header>
<h1>Computer Science - Laboratory</h1>
</header>

<nav>
Home<br>
<a href = "Hamza_html_Java.html">Java</a><br>
<a href = "Hamza_html_Scheme.html">Scheme</a><br>
<a href = "Hamza_html_C.html">C</a><br>
</nav>

<section>
<h1>Home</h1>
<p>Hi name is Hamza Bhatti and this is my html!</p>

<p>I am currently studying Level 4 Computer Science at the University of West
London.</p>

<p>The pages contained in this website show quick examples of the programs that
had been created using Java, C and Scheme.</p>

<p>The website also shows the use of HTML coding and Css which was used to
create this website.</p>
```

```
</section>

<footer>

</footer>

</body>
```

Output:



Computer Science - Laboratory

Home
Java
Scheme
C

Java Programming

Java is one of the most used programming languages used today.

The programming langauge is mainly used to create many objects from one class.

From one class, many different classes can be made, each with their own unique properties an attributes.

Below is just a small example of a Java program and its output.

```
package week6;

public class Week6_Ex3
{
    // check if number is palindrome
    public static void main(String[] args)
    {
        System.out.println("Program to check if a number is a palindrome\n");

        //User is enters a number
        int number = Console.askInt("Enter a number: ");

        //Number is converted to string
        String number1 = String.valueOf(number);

        //Reverse number1 with >> (new StringBuilder(part).reverse().toString());
        String reverse = new StringBuilder(number1).reverse().toString();

        //The reversed string is now converted into an integer
        int reverse1 = Integer.parseInt(reverse);
```

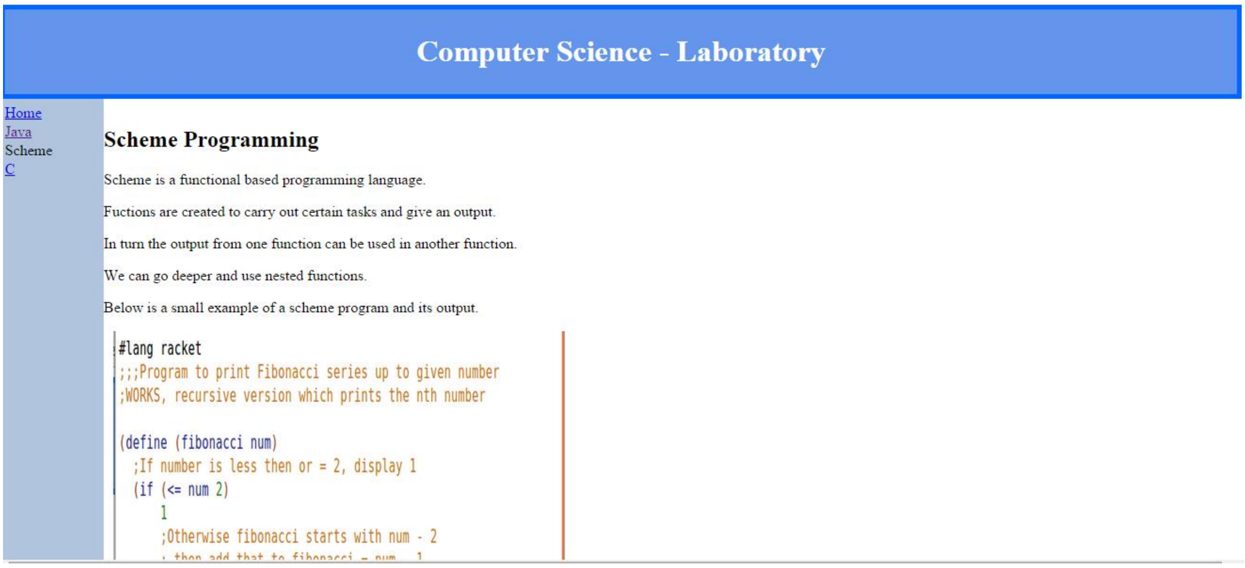
Scheme

Code:

```
<!DOCTYPE html>
<html>
<head>
<style>
header {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    text-align:center;
    padding:5px;
}
nav {
    line-height:20px;
    background-color:#b0c4de;
    height:800px;
    width:100px;
    float:left;
    padding:5px;
}
section {
    width: 100% ;
    float:centre;
    padding:10px;
}
footer {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    clear:both;
    text-align:center;
    padding:5px;
}

</style>
</head>

<body>

<header>
<h1>Computer Science - Laboratory</h1>
</header>

<nav>
<a href = "Hamza_html_Home.html">Home</a><br>
<a href = "Hamza_html_Java.html">Java</a><br>
Scheme<br>
<a href = "Hamza_html_C.html">C</a><br>
</nav>

<section>
<h1>Scheme Programming</h1>
<p>Scheme is a functional based programming language.</p>
<p>Fuctions are created to carry out certain tasks and give an output.</p>
<p>In turn the output from one function can be used in another function.</p>
<p>We can go deeper and use nested functions.</p>
<p>Below is a small example of a scheme program and its output.</p>

<img src="Images/SchemeEx.png" alt="Scheme Example"
style="width:500px;height:500px">
</section>
```

```
<footer>

</footer>

</body>
```

Output:



Scheme Programming

Scheme is a functional based programming language.

Fuctions are created to carry out certain tasks and give an output.

In turn the output from one function can be used in another function.

We can go deeper and use nested functions.

Below is a small example of a scheme program and its output.

```
#lang racket
;;;Program to print Fibonacci series up to given number
;WORKS, recursive version which prints the nth number

(define (fibonacci num)
  ;If number is less then or = 2, display 1
  (if (<= num 2)
      1
      ;Otherwise fibonacci starts with num - 2
      ; then add that to fibonacci - num - 1
```

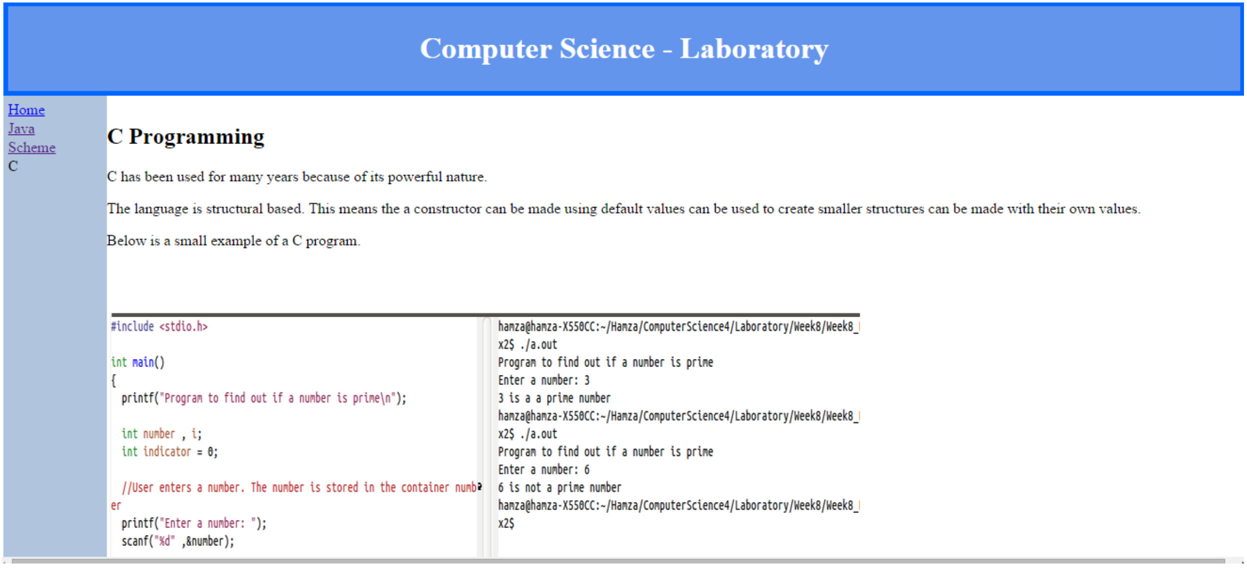
C

Code:

```
<!DOCTYPE html>
<html>
<head>
<style>
header {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    text-align:center;
    padding:5px;
}
nav {
    line-height:20px;
    background-color:#b0c4de;
    height:1000px;
    width:100px;
    float:left;
    padding:5px;
}
section {
    width: 100% ;
    float:centre;
    padding:10px;
}
footer {
    border: 5px solid #0066FF;
    background-color:#6495ed;
    color:white;
    clear:both;
    text-align:center;
    padding:5px;
}

</style>
</head>

<body>

<header>
<h1>Computer Science - Laboratory</h1>
</header>

<nav>
<a href = "Hamza_html_Home.html">Home</a><br>
<a href = "Hamza_html_Java.html">Java</a><br>
<a href = "Hamza_html_Scheme.html">Scheme</a><br>
C<br>
</nav>

<section>
<h1>C Programming</h1>
<p>C has been used for many years because of its powerful nature.</p>
<p>The language is structural based. This means the a constructor can be made
using default values can be used to create smaller structures can be made with
their own values.</p>
<p>Below is a small example of a C program.</p>
<img src="Images/CEx.png" alt="C Example" style="width:800px;height:800px">
</section>

<footer>
```

```
</footer>

</body>
```

Output:



Computer Science - Laboratory

Home
Java
Scheme
C

C Programming

C has been used for many years because of its powerful nature.

The language is structural based. This means the a constructor can be made using default values can be used to create smaller structures can be made with their own values.

Below is a small example of a C program.

```
#include <stdio.h>

int main()
{
  printf("Program to find out if a number is prime\n");

  int number , i;
  int indicator = 0;

  //User enters a number. The number is stored in the container numb
er
  printf("Enter a number: ");
  scanf("%d" ,&number);
```

```
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_
x2$ ./a.out
Program to find out if a number is prime
Enter a number: 3
3 is a a prime number
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_
x2$ ./a.out
Program to find out if a number is prime
Enter a number: 6
6 is not a prime number
hamza@hamza-X550CC:~/Hamza/ComputerScience4/Laboratory/Week8/Week8_
x2$
```