Code Examples from Lectures on 4-2_C_Conditions

relationalOps.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/relationalOps.c)

A program demonstrating the use of relational operators

```
#include <stdio.h>


int main(void){
    int x = 5;

    printf("If x = %d\n",x);
    printf("x > 4 is %d\n",x > 4);
    printf("x >=4 is %d\n",x >= 4);
    printf("x < 4 is %d\n",x < 4);
    printf("x != 4 is %d\n", x!= 4);
    printf("x == 4 is %d\n",x == 4);

    x = 4;

    printf("\nIf x = %d\n",x);
    printf("x > 4 is %d\n",x > 4);
    printf("x >=4 is %d\n",x >= 4);
    printf("x < 4 is %d\n",x < 4);
    printf("x != 4 is %d\n", x!= 4);
    printf("x == 4 is %d\n",x == 4);

    x = 1;

    printf("\nIf x = %d\n",x);
    printf("x > 4 is %d\n",x > 4);
    printf("x >=4 is %d\n",x >= 4);
    printf("x < 4 is %d\n",x < 4);
    printf("x != 4 is %d\n", x!= 4);
    printf("x == 4 is %d\n",x == 4);

    return 0;
}
```

logicalOps.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/logicalOps.c)

A program demonstrating the use of logical operators

```
#include <stdio.h>

int main(void){
    int x = 7;
    printf("If x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",(x > 0) && (x < 10));

    x = -3;
    printf("\nIf x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",(x > 0) && (x < 10));

    x = 10;
    printf("\nIf x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",(x > 0) && (x < 10));


    return 0;
}
```

shortCircuit.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/shortCircuit.c)

An example of short circuit boolean expression evaluation

```c
#include <stdio.h>

int main(void) {
    int x;
    printf("Enter x: ");
    scanf("%d", &x);
    //1000/x only gets evaluated if x!= 0.
    //This is handy as 1000/x will give us a divide by 0 error if x is 0
    printf("%d\n",(x != 0) && (1000/x < 10));

    return 0;
}
```

notRight.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/notRight.c)

A program demonstrating common mis-use use of relational operators and logical operators

```c
#include <stdio.h>

// If you wanted to check whether a number was between 0 and 10 you would use
// (x > 0) && (x < 10)
// below shows some common mistakes

int main(void){

    // Often beginning students confuse && and ||
    // When we have an expression like (x > 0) || (x < 10)
    // it will be true for all numbers!!
    int x = 7;
    printf("If x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",(x > 0) || (x < 10));

    x = -3;
    printf("\nIf x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",(x > 0) || (x < 10));

    x = 10;
    printf("\nIf x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",(x > 0) || (x < 10));


    // An expression like 0 < x < 10 does not do what you may think
    // It does (0 < x) < 10
    // This will also always be true, as (0 < x) will always be 0 or 1
    // Using dcc we will at least get a warning about this
    x = 7;
    printf("If x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n", 0 < x < 10);

    x = -3;
    printf("\nIf x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",0 < x < 10);

    x = 10;
    printf("\nIf x is %d\n",x);
    printf("(x > 0) && (x < 10) is %d\n",0 < x < 10);
    return 0;
}
```

driving.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/driving.c)

A simple program demonstrating an if statement

```c
#include <stdio.h>

int main(void) {
    int age;

    printf("Hello enter your age: ");
    scanf("%d", &age);

    if(age >=MIN_DRIVING_AGE){
        printf("You are old enough to drive\n");
    } else {
        printf("Sorry you are not old enough to drive\n");
    }
    printf("Goodbye\n");

    return 0;
}
```

larger.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/larger.c)

A simple program demonstrating the use of scanf and if statements

```
#include <stdio.h>

int main(void) {
    int a, b;
    printf("Enter a: ");
    scanf("%d", &a);
    printf("Enter b: ");
    scanf("%d", &b);
    if (a > b) {
        printf("%d is greater than %d\n", a, b);
    } else if (a < b) {
        printf("%d is less than %d\n", a, b);
    } else {
        printf("%d is equal to %d\n", a, b);
    }
    return 0;
}
```

absolute1.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/absolute1.c)

Convert a measurement in feet to metres

A simple program demonstrating the use of scanf, printf and an if statement

```
#include <stdio.h>

int main(void) {
    double x, absoluteValue;

    printf("Enter number: ");
    scanf("%lf", &x);

    if(x >= 0){
        absoluteValue = x;
    } else {
        absoluteValue = -x;
    }
    printf("The absolute value of %lf is %lf\n", x, absoluteValue);
    return 0;
}
```

absolute2.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/absolute2.c)

Convert a measurement in feet to metres

A simple program demonstrating the use of scanf, printf and an if statement

```
#include <stdio.h>

int main(void) {
    double x, absoluteValue;

    printf("Enter number: ");
    scanf("%lf", &x);

    absoluteValue = x;
    if(x < 0){
        absoluteValue = -x;
    }
    printf("The absolute value of %lf is %lf\n", x, absoluteValue);
    return 0;
}
```

digits0.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/digits0.c)

A simple program demonstrating the use of scanf and if statements with complex conditions

```c
#include <stdio.h>

#include <stdio.h>

int main(void) {
    int x;
    printf("Enter x: ");
    scanf("%d", &x);

    printf("%d has ", x);

    //Get absolute value of x
    if(x < 0){
        x = -x;
    }

    if ((x < 10) ) {
        printf("1 digit");
    } else if (x < 100) {
        printf("2 digits");
    } else {
        printf("more than 2 digits");
    }


    printf("\n");

    return 0;
}
```

digits1.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/digits1.c)

A simple program demonstrating the use of scanf and if statements with complex conditions

```c
#include <stdio.h>

#include <stdio.h>

int main(void) {
    int x;
    printf("Enter x: ");
    scanf("%d", &x);

    printf("%d has ", x);

    if ((x < 10) && (x > -10)  ) {
        printf("1 digit");

    } else if ((x < 100) && (x > -100) {
        printf("2 digits");
    } else {
        printf("more than 2 digits");
    }


    printf("\n");

    return 0;
}
```

nestedIf.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/nestedIf.c)

A simple program demonstrating the use of scanf and nested if statements

```c
#include <stdio.h>

//Do with nested loops
//-101 big negative number
//-100 small negative number
// 101 big positive number
// 100 small positive number

int main(void) {
    int a;
    printf("Enter a: ");
    scanf("%d", &a);

    printf("%d is a ", a);
    if(a < 0){
        if(a < -100){
            printf("big");
        } else {
            printf("small");
        }
        printf(" negative");
    } else {
        if(a > 100){
            printf("big");
        } else {
            printf("small");
        }
        printf(" positive");
    }
    printf(" number.\n");

    return 0;
}
```

divide.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/divide.c)

Print the result of an integer division checking for divide by 0

```c
#include <stdio.h>

int main(void) {
    int x, y;

    printf("Enter x: ");
    scanf("%d", &x);
    printf("Enter y: ");
    scanf("%d", &y);

    //TODO Print an error instead of doing if we would get a divide by 0 error
    if( y!= 0){
        printf("x/y is %d\n",x/y);
    } else {
        printf("Divide by zero!\n");
    }


    return 0;
}
```

divisibleExercise.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/divisibleExercise.c)

whether an integer is divisible by 2 and whether an integer is divisble by 3

```c
#include <stdio.h>
//test cases

int main(void) {
    int num;
    printf("Enter an integer: ");
    scanf("%d",&num);
    printf("%d\n", num);
    if(num%2 == 0){
        printf("%d is divisible by 2\n",num);
    }

    if(num%3 == 0){
        printf("%d is divisible by 3\n",num);
    }


    return 0;
}
```

debugging.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/debugging.c)

A simple program demonstrating terrible style and errors with if statements

```c
#include <stdio.h>

//Fix the style
//Identify the errors
//Fix the errors
int main(void) {
    int age;
    printf("Enter your age: ");
    scanf("%d", &age);

    printf("Happy Birthday. You are %d \n", age);
    if (age <= 0) {
 printf("You are a liar\n");
    } else {
    if (age == 100) {
    printf("Happy Birthday from the Queen!\n");
    } else
    if(age = 18){
                printf("Now you can vote!\n");
            }

       printf("You are a minor\n");
                                  }


    return 0;
}
```

debuggingFixed.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/debuggingFixed.c)

A simple program demonstrating fixing terrible style and errors with if statements from debugging.c

```c
#include <stdio.h>

//Fix the style
//Identify the errors
//Fix the errors
int main(void) {
    int age;
    printf("Enter your age: ");
    scanf("%d", &age);

    printf("Happy Birthday. You are %d \n", age);
    if (age < 0) {
        printf("You are a liar\n");
    } else {

        if (age == 100) {
            printf("Happy Birthday from the Queen!\n");
        } else if(age >= 18){
            printf("Now you can vote!\n");
        } else {
            printf("You are a minor\n");
        }
     }

     return 0;
}
```

lecture_five2.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/lecture_five2.c)

```c
#include <stdio.h>

// ! means not
int main(void) {
    int a;
    scanf("%d", &a);
    if (a < 0) {
        printf("Less than zero\n");
    }

    if (a >= 0 && a < 10){
        printf("Greater/eq than 0, less than 10\n");
    }

    if (a >= 10 && a < 20) {
       printf("Greater/eq than 10, less than 20\n");
    }

    if (a >= 20) {
       printf("Greater/eq to 20\n");
    }
    return 0;
}
```

lecture_five.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/lecture_five.c)

```c
#include <stdio.h>

// ! means not
int main(void) {
    int a;
    scanf("%d", &a);
    if (a < 0) {
        printf("Less than zero\n");
    } else if (a < 10) {
        printf("Greater/eq than 0, less than 10\n");
    } else if (a < 20) {
       printf("Greater/eq than 10, less than 20\n");
    } else {
       printf("Greater/eq to 20\n");
    }
    return 0;
}
```

lecture_five3.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/lecture_five3.c)

```c
#include <stdio.h>

// ! means not
int main(void) {
    int a;
    scanf("%d", &a);
    if (a < 0) {
        printf("Less than zero\n");
    }
    printf("I promise\n");
    return 0;
}
```

lecture_five4.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/lecture_five4.c)

```c
#include <stdio.h>

// ! means not
int main(void) {
    int a;
    scanf("%d", &a);
    if (a < 0) {
        printf("Less than zero\n");
    } else if (a < 10) {
        printf("Greater/eq than 0, less than 10\n");
    } else if (a < 20) {
       printf("Greater/eq than 10, less than 20\n");
    } else {
       printf("Greater/eq to 20\n");
    }
    return 0;
}
```

lecture_six.c (https://cgi.cse.unsw.edu.au/~cs1911/22T2/lec/4-2_C_Conditions/code/lecture_six.c)

```c
#include <stdio.h>

int main(void) {
    int x;
    int y;
    x = 0;
    y = 100;
    if (y / x > 10) {
        printf("We made it!\n");
    }
    return 0;
}
```