****

**C**



**Goals when writing code:**

**⏵The 3 axes are:**

**1. Correctness**

**2. Design**

**3.Style**

**• ultimately it will look like this:**

**Example of just printing “hello, world”**

**#include <stdio.h>**

**int main(void)**

**{**

**printf(“hello, world\n”);**

**}**

**• We can write in word or other program BUT the problem is that the computers just understand the language Binary (0 and 1) so the computer will not understand it if we write it exactly like this in those programs because it has to be translated into 0 and 1’s**

**and none of those tools are really appropriate for programming because they come with bold features and other fluffy stuff that has no impact on what are you trying to do with your code**

**• The 0’s and 1’s are called machine code**

**moreover they do not have the possibility to convert that code into 0’s and 1’s**

**• Tools that do have this capability might be called Integrated Development Environments or IDE’s or simply text editors**

**• The lines in the IDE will help us to know how long or short the program is**

**• Compiler is just converting one language to another**

**Source code -> compiler -> machine code**

**• IDE’s have a terminal window = old school for a person to interact with the computer with the keyboard, $ sign means type your commands here, this is where we write: compile my code into machine code**

**• You can also go to the terminal and write : make hello -> it will create ./hello (folder in my account on cloud) -> enter and you will have hello, world**

**• Terminal = CLI = command line interface**

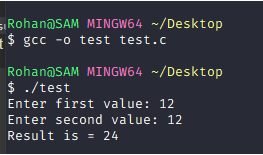
**• Removing using terminal = rm hello the y for yes**

**• So, GUI from the left does what the terminal does, anything that you do with the mouse can be done with the command interface**

**• You have to write make in the terminal to recompile (make is like save)**

**So the top part of the IDE is the text editor, the below part is the terminal window, where $ means type your commands here**

**From terminal after writing in text editor we will do the following:**

** so . means go into my current folder, /hello run the program hello in this current folder**

**Functions, arguments:**

**Functions are actions and verbs like say or ask**

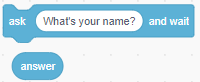
**Arguments are the input for those functions**

**A function in scratch is  and in c is printf(“hello, world\n”); <- take note “” because you use a string and in C every thought is finished with ;**

**Some Functions don’t have side effects but they have return values, variables, something that you can reuse**

**Arguments -> functions -> side effects**

**Arguments -> functions -> return value that we can keep and reuse (like the ask block), so they hand you an output that you can use and reuse**

**Ask blockis equivalent or translated in C to -> string answer = get\_string(“What’s your name? “); <-this takes an input, = is the assignment operator, to assign a value is to store a value into a variable**

**This you will be read it from right to left, so we have a function get\_string that will return to you whatever the person will type and it’s going to be stored on the left in the variable**

**Because of the assignment also in C you have to tell it what type the variable is storing**

**In order to take input from the user you will have to include a library <cs50.h>**

**There is a standard ibrary io (input/output) in C there is a library**

**Bro code:**

**C is a middle level labguages from 1970 and widley used language today**

**C is a middle level language**

**Nearly every language uses or is influenced by “C”**

**Example: Python is written with C**

**Default implementation CPython**

**Important notes:**

**1.C is not OOP**

**2. C++ is an extension of C**

**3.C is difficult**

**You will need an IDE and a GCC compiler to transform into code machine 0 and 1**

**So you can use VS Code then in the extension C/C++ installer and also C runner**

# Check in cmd with g++ --version if you have a gcc installed otherwise you will have to install Mingw-W64 for windows

# #include = the word is a preprocessor command that tells the proccesso rto include the contents of a file wich it ill be in <>

# <stdio.h> will include functions usefeull for input/output tht we will need

# Our main entry point is the main function type int main(){} so anything within the {} is the main function and at the end we use ;

# Use // to write a comment and will be ignored by the compiler

# For multi line comment /\* comments here \*/

# Escape sequence = character combination consisting of a backlash \ followed by a letter or combination of digits.

# \n = newline

# \t = tab

# To print quotes inside the () use printf(“\” I like pizza \” – by lulu”) or for single quotes use ‘ instead of “ = escape of sequences

# Variables = allocated space in memory to store a vlue

# We refer to a variable’s name to access the stored value

# That variable now behaves as if it was the value it contains

# But we need to declare what type of data we are storing

# It is done in 2 steps, declaration then initialization

# int x; // declaration

# x=123;// initialization

# int y = 321;//declaration + initialization

# to create an array: char name[]=”Bro”;//array of characters -> this will act as string!!! So we have to store more than 1 character and store in an array!!emulate a string

# to print the above in a statement: printf(“you are % years old”, d) -> so % is a place holder followed by d for decimal, for name use format specifier %s and for float is %f, c% for char

# %lf = long float, so to display 15 digits after I will do %lf15 so double use more memory but more precision

# To work with Boolean you have to include : #include <stdbool.h>

# bool e = true

# format specifier = defines and formats a type of data to be displayed

# %c = character

# %s = string (array of character)

# %f = float

# %lf = double

# %.1 = decimal precision

# %-=left alignso you use % then number = it will put a tab after the number but if you add a . and then a number you will tell it how may numbers you will have: printf(“item 1: $%8.2f\n”, item1);

# So to display output use data format specifier

# Constant = is a fixed value that cannot be altered by the program during its execution

# float pi = 3.14159; -> to make sure nobody will change this data value we will put const in the front and change the var to capital letters-> ex: const float PI = 3.14159;

# operations:

# if yu divide 2 numbers use float

# to cast the int you have to put (float) before

# ex:

# int x = 5;

# int y = 2;

# float z = x / (float)y;

# printf(“the total is %.2f”,z) <- make sure you do not forget about the %f to be float

# so when you divide be carefull that wyou might need to cast it int a float or a double

# modulus gives you the reminader of every division

# if you want to increment you can use ++ or y--

# x++;

# y--;

# augumented assignment operators = used to replace a stattment where an operator takes as one of its arguments

# and then assign the result back to the same varible

# x = x+1

# x+=1

# to read input we use scanf() and you have to write like this:

# int age; -> 1. First you create the variable

# printf(“how old are you?”);fflush(stdout); ->2. Print the message

# scanf(“%d”, &age); -> 3.store the result through scanf() and use & before the var

# printf(“you are %d years old”,age);

#include <stdio.h>

#include <string.h> *// we will add this for string formatting*

int main()

{

    char name[25];*//bytes//size of array is 25*

    int age;

    printf("what's your name? \n");fflush(stdout);

*//scanf("%s", &name);//you will use tis if there is no space in the word*

    fgets(name, 25, stdin);*//3 things(name of variable without address of variable &, size that we will make it to match our array, stdin which stand for standard input)*

*//fgets will include automatically \n after so do below to remove that*

    name[strlen(name)-1]='\0'; *//so we will have no space after*

    printf("what is your age? \n");fflush(stdout);

    scanf("%d", &age);

    printf("%s, nice to meet you\n", name);

    printf("%s you are: %d years old", name, age);

}

# 

# 

# 

# anything inside these braces is part of the puzzle piece main

# 

# Header files: in c you also have to include #include <stdio.h> so you need to tell the compiler to load the library, the code that somebody else wrote so that the compiler knoes what printf means

# So <stdio.h> is called a header file = a menu of available functions so It will prepare the comiler how to implement those functions

# In c there are no technacilay such data type as string by default

# 

# Linux commands:

# 

# touch = create new file

# types of commands in c:

# 

# So in C programming depends on you to tell the computer what is the pattern of 0 and 1, is it a number, letter, a sound, image or a video…

# In the cs50 the functions yu will find:

# 

# 

# 

# Operators

# 

# Variables, syntactic sugar

# 

# 

# 

# 

# In c if you are using printf, first you need to have a string inside the brackets then what you need to do

# gcc -o calculator calculator.c – compile with this so computer will understand what you want it to do

# 1.16 <<< Harvard

# Bro code

# Math function

# You have to include : #include <math.h>

# Make sure you use the appropriate format specifier (lf for double, d for int)

# 

# Example of program to calculate the circumference of a circle and the area also

# The formula to calculate the circumference is :

# 

# The formula for area circle is:

# 

#include <stdio.h>

*//calculate circumference of a circle and the area also*

int main(void)

{

*//first we declare the variables*

    const double PI = 3.14159; *//so nobody will be able to change this*

    double radius;

    double circumference;

    double area;

*//prompt the user to enter a radius*

    printf("\nEnter the radius of a circle: ");fflush(stdout);*//\n is new line*

*//use scanf to accept some user input*

    scanf("%lf", &radius);*//list in the '' the format specifier of double which is ls then the address of the operator*

*//add the formula to our variable circumference*

    circumference = 2 \* PI \* radius;

*//add the formula to calculate the area of the circle*

    area = PI \* radius \* radius;

*//display our circumference with printf*

    printf("\ncircumference: %lf", circumference);fflush(stdout);

    printf("\narea: %lf", area);

    return 0;

}

# 