Barak Barclay

Leslie Tekamp

ECE-1021

30 September 2015

HWK-4

//

// HWK4(1).cpp

//

// By: Barak Barclay

// Date: 30 Sept 2015

//

// Problem Statement: Write a program that prompts the user to enter an

// integer number one-to-five (numbers represent the first five elements

// in the periodic table) or enter a negative number to quit. Read in the

// integer. Error check to see if the number is with-in bounds

// (one-to-five) – if out of bounds print an error message and start over

// again, else print out the name of the element. If the user enters a

// valid integer, using if-else statements print out the name of the entered

// element. Loop until the user enters a negative number.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variable

// while loop (num >= 0)

// prompt user to enter integer

// read in num

// if statements

// end while

// end main

//

#include <stdio.h>

int main(void)

{

int num = 0;

while (num >= 0)

{

printf\_s("\nEnter a integer between 1 and 5.\n");

scanf\_s("%d", &num);

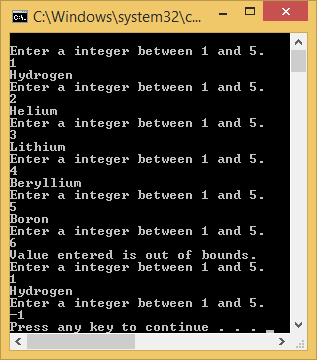
if (num == 0 || num > 5)

{

printf\_s("Value entered is out of bounds.");

}

if (num == 1)

 {

printf\_s("Hydrogen");

}

if (num == 2)

{

printf\_s("Helium");

}

if (num == 3)

{

printf\_s("Lithium");

}

if (num == 4)

{

printf\_s("Beryllium");

}

if (num == 5)

{

printf\_s("Boron");

}

}

}

//

// HWK4(1).cpp

//

// By: Barak Barclay

// Date: 30 Sept 2015

//

// Problem Statement: Rewrite the first problem using a switch statement

// instead of if-else statements.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variable

// do while loop (num >= 0)

// switch statement

// end while

// end main

//

#include<stdio.h>

int main(void)

{

int num = 0;

do{

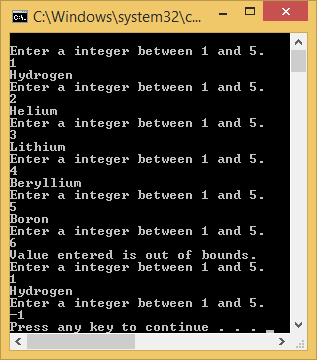
printf\_s("\nEnter a integer between 1 and 5.\n");

scanf\_s("%d", &num);

switch (num)

{

case 1:

 {

printf\_s("Hydrogen");

break;

} //end case

case 2:

{

printf\_s("Helium");

break;

} //end case

case 3:

{

printf\_s("Lithium");

break;

} //end case

case 4:

{

printf\_s("Beryllium");

break;

} //end case

case 5:

{

printf\_s("Boron");

break;

} //end case

default:

{

printf\_s("Value entered is out of bounds.");

break;

} //end case

} //end switch

} while (num >= 0); //end do..while

return 0;

}

//

// HWK4(3).cpp

//

// By: Barak Barclay

// Date: 30 Sept 2015

//

// Problem Statement: Write a program that prompts the user to enter

// in an integer number. Read in the integer number. Using a for loop,

// calculate the sum of all the integers from zero to the entered number

// (including the entered number). Print the result to the screen.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variable

// prompt user tot enter integer

// read in num

// for loop

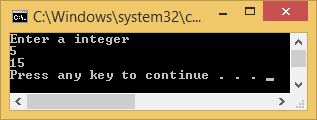
// calculate sum

// end for

// print sum

// end main

//

#include <stdio.h>

int main(void)

{

int num = 0;

int i = 0;

int sum = 0;

printf\_s("Enter a integer\n");

scanf\_s("%d", &num);

for (i = 0; i < num; i++)

{

sum = sum + (num - i);

} //end while

printf\_s("%d\n", sum);

return 0;

}//main

//

// HWK4(4).cpp

//

// By: Barak Barclay

// Date: 30 Sept 2015

//

// Problem Statement: Use a do..while loop. Write a program that prompts

// the user to enter the radius of a circle – type double. Read in the radius.

// Next print a list of menu options for the user to choose from – see below.

// Read in the user option and preform the desired function. Repeat until the

// user enters a number to quit. Print an error message if the user enters an

// integer number outside the allowed values. Repeat until the user enters

// the Quit value.

// 0 – Calculate the diameter of a circle.

// 1 – Calculate the circumference of the circle.

// 2 – Calculate the area of the circle.

// 3 – Calculate the volume of a sphere.

// 4 – Quit program.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variable

// prompt user tot enter radius

// read in r

// do while num != 4

// prompt to enter one of options

// read in num

// if num == 4

// break

// end if

// switch

// cases

// end switch

// end do while

// end main

//

#include <stdio.h>

int main(void)

{

double r = 0;

double d = 0;

double c = 0;

double a = 0;

double v = 0;

int num = 0;

printf\_s("Enter a radius of a circle\n");

scanf\_s("%lf", &r);

do

{

printf\_s("Enter 0 to calculate the diameter of a circle.\nEnter 1 to calculate the circumference of the circle.\nEnter 2 to calculate the area of the circle.\nEnter 3 to calculate the volume of a sphere.\nEnter 4 to quit program.\n");

scanf\_s("%d", &num);

if (num == 4)

{

break;

}

switch (num)

{

case 0:

{

d = 2.0 \* r;

printf\_s("%lf\n", d);

break;

} //end case

case 1:

{

c = 2.0 \* 3.141592 \* r;

printf\_s("%lf\n", c);

break;

} //end case

case 2:

{

a = 3.141592 \* r \* r;

printf\_s("%lf\n", a);

break;

} //end case

case 3:

{

v = (4.0 / 3.0) \* 3.141592 \*r \* r \* r;

printf\_s("%lf\n", v);

break;

} //end case

default:

{

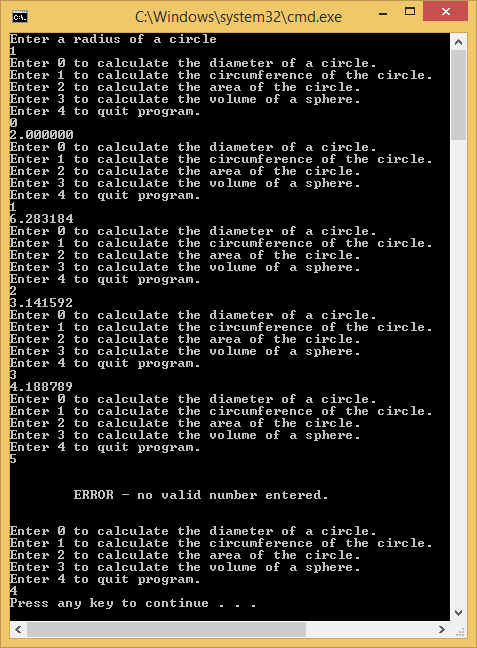
puts("\n\n ERROR - no valid number entered. \n\n");

break;

} //end case

} //end switch

} while (num != 4);

 return 0;

}//main

//

// HWK4(5).cpp

//

// By: Barak Barclay

// Date: 30 Sept 2015

//

// Problem Statement: Use the math library. Write a program to calculate

// the sin function from 0 to pi, in increments of pi/20.

// y = sin(i\*pi / 20) where i is an integer 0 to 20, and y is a float.

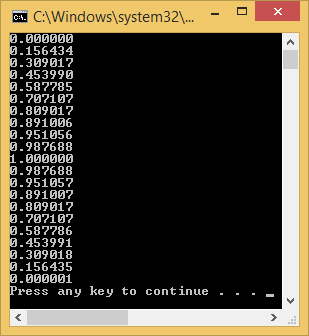
// print out the 21 values to the screen.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variable

// for

// y = sin(i\*pi / 20)

// print y

// end for

// end main

//

#include <stdio.h>

#include <math.h>

int main(void)

{

int i = 0;

double y = 0;

for (i = 0; i <= 20; i++)

{

y = (double)sin((double)i\*3.141592 / 20.0);

printf("%lf\n", y);

} //end while

return 0;

}//main

//

// HWK4(6).cpp

//

// By: Barak Barclay

// Date: 30 Sept 2015

//

// Problem Statement:Rewrite problem 5, but this time you will graph

// the result. After you calculate the result y, multiple it by 15,

// type cast the float to an int and print out that many ‘\*’ to the

// screen. Go to the next line and repeat with the next result for y.

// Repeat for all 21 values. You should see a graph of the sine function,

// down the page, showing the positive sine shape.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

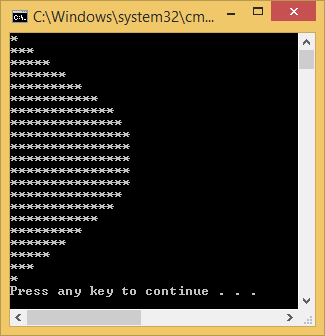
// start main

// init variable

// for

// y = sin(i\*pi / 20)

// y = y \* 15.0;

// for loop

// print \*

// end for

// print new line

// end for

// end main

//

#include <stdio.h>

#include <math.h>

int main(void)

{

int i = 0;

int i1 = 0;

double y = 0;

int y1 = 0;

for (i = 0; i <= 20; i++)

{

y = (double)sin((double)i\*3.141592 / 20.0);

y1 = y \* 15.0;

for (i1 = 0; i1 <= y1; i1++)

{

printf\_s("\*");

}

printf\_s("\n");

} //end while

return 0;

}//main

//

// HWK4(7).cpp

//

// By: Barak Barclay

// Date: 30 Sept 2015

//

// Problem Statement: Write a program to produce the output below using

// looping structures. Zero points for not using looping structures.

// Each new line of ‘#’ is indented one extra space. Line one of the

// #’s starts at column 10.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variable

// for 5 times

// for 9 times

// print values 1-9

// end for

// print 0

// end for

// for 10 times

// while

// print spaces

// end while

// print ##########

// end for

// for 5 times

// for 9 times

// print values 1-9

// end for

// print 0

// end for

// end main

//

#include <stdio.h>

#include <math.h>

#include<stdio.h>

int main(void)

{

int line;

int line1 = 1;

int repeat;

int spaces = 0;

char Header[] = "##########";

for (repeat = 1; repeat <= 5; ++repeat)

{

for (line = 1; line <= 9; ++line)

{

printf\_s("%d", line);

}

printf\_s("0");

}

printf\_s("\n\n");

for (line = 1; line <= 10; ++line)

{

line1 = line;

while (spaces < line1)

{

line1--;

printf\_s(" ");

}

printf\_s("%20s\n", Header);

}

printf\_s("\n");

for (repeat = 1; repeat <= 5; ++repeat)

{

for (line = 1; line <= 9; ++line)

{

printf\_s("%d", line);

}

printf\_s("0");

}

printf\_s("\n\n");

return 0;

}

