**EEL 2880: Engineering Software : Modified Spring 2019: 01 22 2019 Pr3\_4**

**Subbarao V Wunnava: Chap: 3&4 & 5 problems 4.12 & 4.13 & 5.5 Deitel:**

//01 19 2019: Modified Fall 2016/Spring 2019 Subbarao: Ref Chp 3&4, problem 4.12 & 4.13 Deitel

//demonstration of for loop, even and odd numbers: Modified 01 17 2017

//Calculate sum of even and odd numbers through specified limit

#include <stdio.h>

#include <math.h>

#include <stdlib.h>

int main( void )

{

unsigned int i; // counter to count numbers

unsigned int m; // enter final number

unsigned int esum = 0, osum = 0; //sum of even and odd integers esum, osum initially

printf(" \n 05 19 2015: Summer 2015: Subbarao: Ref Chp 3&4 \n ");

printf(" \t\a demonstration of for loop, even and odd numbers\n ");

printf(" enter a final value for the number m: \a\t");

scanf ("%u", &m);

// loop through even integers up to m

for ( i = 2; i <= m; i += 2 ) //increment by 2 to stay even boundaries

{

(esum += i); // add i to even sum esum

(osum +=(i-1)); // adjusted i for odd numbers

printf ("%u\t", (i-1)); //print odd integer value

printf ("%u\t", i); //print even integer value

} // end for loop

printf( "\n\a Sum of the even integers from 0 to m is: %u\n", esum );

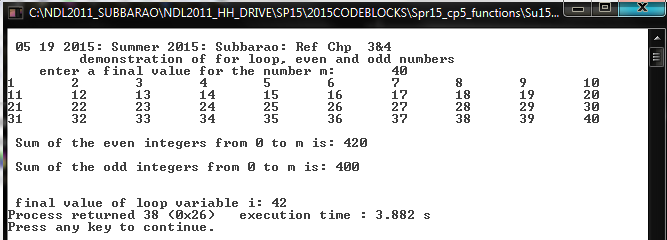
printf( "\n\a Sum of the odd integers from 0 to m is: %u\n\n", osum );

printf( "\n\a final value of loop variable i: %u\t", i );

// system ("PAUSE");// Windows command for keeping the display active

// return 0;

} // end main program for\_while\_loops.c



**EEL 2880: Engg SoftwareWare: Modified Spring 2020 01 28 2020**

**Subbarao Wunnava/Copy right Deitel extended to 2019**

**Control structures and Math Operations Ref Ch3, 4 & 5 & 6**

***Write a functional program code in -C- and which will show nested loops, Math Operations such as factorial, squares, cubes etc.* Nested Loops and Mathematical Functions**

// nested loops and math functions :Original: Summer2015 updated Spring 2017: Subbarao Wunnava

// Exercise 4.14 Solution nested loops and math functions: factorial, square, cube..

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main( void )

{

unsigned int i; // outer counter

unsigned int j; // inner counter

unsigned int factorial,square,cube; // current factorial value

puts( "X\t\ Factorial X\t\t square\t\t cube\n\n" ); // display table headers

// compute the factorial, square, and cube values for X, for X value 1 to 5

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

{

factorial = 1, square = 1, cube = 1;

// calculate factorial of current number

for ( j = 1; j <= i; ++j )

{

factorial \*= j;

square = pow (j,2);

cube = pow (j,3);

} // end inner for

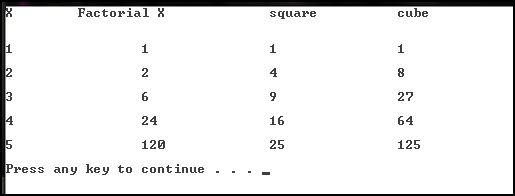
printf( "%u\t\t %u\t\t %u\t\t %u\t\t \n\n", i,factorial,square,cube);

} // end outer for

system ("PAUSE");\\ requirement to hold the display for Microsoft Visual studio

return 0;

} // end main



**EEL 2880: Spring 2020: Switch and While loops: Commercial Applications**

**Subbarao Wunnava Pr6 Chp 3-4-5 : Updated 01 28 2020**

**4.15/5.15 *(Modified Compound Interest Program)*** Modify the compound-interest program of

Section 4.6 & 5.6 to repeat its steps for interest rates of 5%, 6%, 7%, 8%, 9%, and 10%.

**1** // Exercise 4.15 Solution

**2 #include** <stdio.h>

**3 #include** <math.h>

**4**

**5 int** main( **void** )

**6** {

**7 unsigned int** year; // year counter

**8 int** rate; // interest rate

**9 double** amount; // amount on deposit

**10 double** principal = **1000.0**; // starting principal

**11**

**12** // loop through interest rates 5% to 10%

**13 for** ( rate = **5**; rate <= **10**; ++rate ) {

**14**

**15** // display table headers

**16** printf( **"Interest Rate: %f\n"**, rate / **100.0** );

**17** printf( **"%s%21s\n"**, **"Year"**,**"Amount on deposit"** );

**18**

**19** // calculate amount on deposit for each of ten years

**20 for** ( year = **1**; year <= **10**; ++year ) {

**21**

**22** // calculate new amount for specified year

**23** amount = principal \* pow( **1** + ( rate / **100.0** ), year );

**24**

**25** // output one table row

**26** printf( **"%4u%21.2f\n"**, year, amount );

**27** } // end for

**28**

**29** puts( **""** );

**30** } // end for

**31** } // end main [Results only shown for 5% and 8% interest rates, for simplicity]

**Interest Rate: 0.050000**

**Year Amount on deposit**

**1 1050.00**

**2 1102.50**

**3 1157.63**

**4 1215.51**

**5 1276.28**

**6 1340.10**

**7 1407.10**

**8 1477.46**

**9 1551.33**

**10 1628.89**

**...**

**Interest Rate: 0.080000**

**Year Amount on deposit**

**1 1080.00**

**2 1166.40**

**3 1259.71**

**4 1360.49**

**5 1469.33**

**6 1586.87**

**7 1713.82**

**8 1850.93**

**9 1999.00**

**10 2158.92**

4.18/5.18 *(Bar Chart Printing Program)* One interesting application of computers is drawing graphs

and bar charts (sometimes called “histograms”). Write a program that reads five numbers (each between

1 and 30). For each number read, your program should print a line containing that number

of adjacent asterisks. For example, if your program reads the number seven, it should print \*\*\*\*\*\*\*.

Solution: Courtesy: Deitel Group: Modified SubbaraoWunnava [FIU] with permission from the group 01 21 2020

1 // Exercise 4.18/5.18 Solution

2 #include <stdio.h>

3

4

int main( void )

5 {

6 unsigned int i; // outer counter

7 unsigned int j; // inner counter

8 unsigned int number; // current number

9

10 printf( "%s", "Enter 5 numbers between 1 and 30: " );

11

12 // loop 5 times

13 for ( i = 1; i <= 5; ++i )

{

14 scanf( "%u", &number );

15

16 // print asterisks corresponding to current input

17 for ( j = 1; j <= number; ++j )

{

18 printf( "%s", "\*" );

19 } // end for inner loop

20

21 puts( "" );

22 } // end for outer loop

23 } // end main

Enter 5 numbers between 1 and 30: 28 5 13 24 7

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

**4.19/5.19 *(Calculating Sales)*** An online retailer sells five different products whose retail prices are

shown in the following table:

Write a program that reads a series of pairs of numbers as follows:

a) Product number b) Quantity sold for one day

Your program should use a switch statement to help determine the retail price for each product.

Your program should calculate and display the total retail value of all products sold last week.

[Product: Retail price]

1: $ 2.98 2: $ 4.50 3: $ 9.98 4: $ 4.49 5: $ 6.87

**1** // Exercise 4.19/5.19 Solution: Dietel/Subbarao 01 28 2020

2 #include<stdio.h>

**3**#include<math.h>

**4**

**int** main( **void** )

**5** {

**6 int** product; // current product number

**7 int** quantity; // quantity of current product sold

**8 double** total = **0.0**; // current total retail value

9

**10** // prompt for input

**11** puts( **"Enter pairs of item numbers and quantities."**);

**12** puts( **"Enter -1 for the item number to end input."** );

**13** scanf( **"%d"**, &product );

**14**

**15** // loop while sentinel value not read from user

**16 while** ( product != **-1** ) {

**17** scanf( **"%d"**, &quantity );

**18**

**19** // determine product then perform calculation

**20 switch** ( product ) {

**21**

**22 case 1**:

**23** total += quantity \* **2.98**; // update total

**24 break**;

**25**

**26 case 2**:

**27** total += quantity \* **4.50**; // update total

**28 break**;

**29**

**30 case 3**:

**31** total += quantity \* **9.98**; // update total

**32 break**;

**33**

**34 case 4**:

**35** total += quantity \* **4.49**; // update total

**36 break**;

**37**

**38 case 5**:

**39** total += quantity \* **6.87**; // update total

**40 break**;

**41**

**42 default**:

**43** printf( **"Invalid product code: %d\n"**, product );

**44** printf( **" Quantity: %d\n"**, quantity );

**45** } // end switch

**46**

**47** scanf( **"%d"**, &product ); // get next input

**48** } // end while

**49**

**50** // display total retail value

**51** printf( **"The total retail value was: %.2f\n"**, total );

**52** } // end main

Enter pairs of item numbers and quantities.

Enter -1 for the item number to end input.

1 1

2 1

3 1

4 1

5 1

6 1

Invalid product code: 6

Quantity: 1

1 1

-1

The total retail value was: 31.80

**ASCII and Hex Decimal coded Numbers for Computer Applications: Courtesy: IBM/Bell Laboratories**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 | 0 1 2 3 4 5 6 7 8 9 A B C D E F | NUL SOH STX ETX EOT ENQ ACK BEL BS TAB LF VT FF CR SO SI | | |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 | 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F | DLE DC1 DC2 DC3 DC4 NAK SYN ETB CAN EM SUB ESC FS GS RS US | | |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 | 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F | (space) ! " # $ % & ' ( ) \* + , - . / | | |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 | 30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F | 0 1 2 3 4 5 6 7 8 9 : ; < = > ? | |
| |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 | 40 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F | @ A B C D E F G H I J K L M N O | | |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 | 50 51 52 53 54 55 56 57 58 59 5A 5B 5C 5D 5E 5F | P Q R S T U V W X Y Z [ \ ] ^ \_ | | |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 | 60 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F | ` a b c d e f g h i j k l m n o | | |  |  |  | | --- | --- | --- | | **ASCII** | **Hex** | **Symbol** | |  | | | | 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 | 70 71 72 73 74 75 76 77 78 79 7A 7B 7C 7D 7E 7F | p q r s t u v w x y z { | } ~  | |