Michael Campbell

X52.9008 – 01

Tues 6:30-9:00PM

July 6, 2010

--Midterm—

1. You are given a **character array** (i.e. **char grade[32]**) of 32 grades spanning the values of **A – F, I**. You need to compute the histogram for each of the six possible grades which will be contained in a **histogram array** (i.e. **int** **hist\_grade[6]**) containing six values. Using the ***switch statement structure***, write a program that will calculate how many grades were given for each grade. Use the following values for the grade array:

char grade[32] = {

‘A’, ‘B’, ‘C’, ‘D’, ‘f’, ‘I’,

‘b’, ‘C’, ‘d’, ‘F’, ‘I’, ‘a’,

‘C’, ‘D’, ‘F’, ‘I’, ‘A’, ‘B’,

‘D’, ‘F’, ‘I’, ‘A’, ‘B’, ‘c’,

‘I’ , ‘D’, ‘F’, ‘F’, ‘A’, ‘B’,

‘a’, ‘b’

};

Hint: Use a ***for loop*** to walk through the grade array and within the loop have

the **switch statement** populate the **hist\_grade** array accordingly.

/\* Program; Histogram.c

Author: Michael Campbell

Date; 7/6/10

Synopsis: Using a character array of 32 grades, compute a histogram for each of the six possible grades (held in histogram array)

using switch statement structure

\*/

#include <stdio.h>

#include <ctype.h>

int main (void)

{

//Initialize Variables

int i;

char grade[32] = { 'A', 'B', 'C', 'D', 'f', 'I', \

'b', 'C', 'd', 'F', 'I', 'a', \

'C', 'D', 'F', 'I', 'A', 'B', \

'D', 'F', 'I', 'A', 'B', 'c', \

'I' , 'D', 'F', 'F', 'A', 'B',\

'a', 'b'};

int hist\_grade[6] = {0};

//For Loop To walk Through the grade array

for (i = 0; i < 32; ++i) {

//If statement to convert lowercase to uppercase

if (islower(grade[i]))

grade[i]=toupper(grade[i]);

//Switch Statements

switch (grade[i]) {

case 'A':

++hist\_grade[0];

break;

case 'B':

++hist\_grade[1];

break;

case 'C':

++hist\_grade[2];

break;

case 'D':

++hist\_grade[3];

break;

case 'F':

++hist\_grade[4];

break;

case 'I':

++hist\_grade[5];

break;

}

}

//Print Histogram

printf("A\tB\tC\tD\tF\tI\t\n");

for (i = 0; i < 6; i++){

hist\_grade[i];

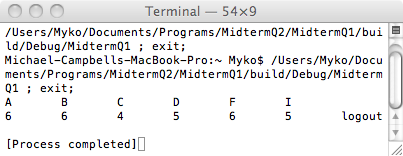
printf("%i\t", hist\_grade[i]);

}

return 0;

}

Output:



1. Write a program that initializes each row of a **10 x 8 matrix, A[10][8],** with the value of the row (e.g. the first row contains all 0’s, the second row contains all 1’s, etc).

* After printing the matrix for the above problem, transpose the matrix into a **second matrix** (**8 x 10**), **B[8][10],** and print out the new transposed matrix.
* This program should demonstrate the use of ***nested for loops***.

/\* Program; Matrix.c

Author: Michael Campbell

Date; 7/6/10

Synopsis: Initializes each row of a 10 x 8 matrix with the value of the row,

prints the matrix,

transpose the matrix into a second matrix (8 x 10), B[8][10], then prints out the new transposed matrix.

\*/

#include <stdio.h>

int main (void)

{

//Initialize Variables

int row;

int column;

int A[10][8] = {0};

int B[8][10] = {0};

//Initialize array A

for (row = 0; row < 10; ++row)

for (column = 0; column < 8; ++column)

A[row][column] = row;

//Print Two dimensional Array 'A'

printf("\n");

printf("Array A\n");

printf("\n");

for (row = 0; row < 10; ++row)

for (column = 0; column < 8; ++column)

printf("%5i", A[row][column]);

printf("\n");

printf("\n");

// Transpose Array 'A' to Array 'B'

for (row = 0; row < 10; ++row)

for (column = 0; column < 8; ++column)

B[column][row] = A[row][column];

// Print Two dimenstional Array 'B'

printf("\n");

printf("Array B\n");

printf("\n");

for (row = 0; row < 8; ++row)

for (column = 0; column < 10; ++column)

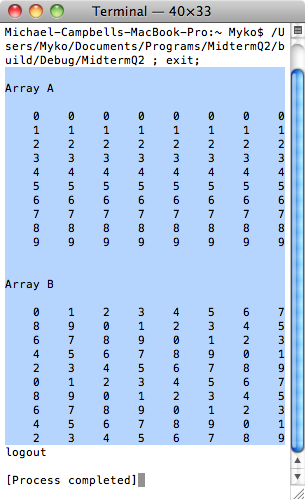
printf("%5i", B[row][column]);

printf("\n");

return 0;

}

Output:



1. Write a program that calculates the sum of the first **20 Fibonacci** numbers. The generated Fibonacci numbers should be stored in an integer array (i.e. **int fib[20]**). Recall from class readings that the Fibonacci series for the first 20 number is as follows:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, … etc… 6765 which can be generated from the following recursion equation

***Fn* = *Fn*−1 + *Fn*−2**

*where* ***F0  = 0, F1 = 1, F2 = 2,…....***

* The program should be organized such that it **calls a *function*****fibonnaci(&array[0], size)** which calculates the Fibonacci series followed by a second ***function*** sum( **int sum(&array[0], size**) which calculates the sum of all 20 Fibonacci numbers.

1. You are given the following string ***as an example*** in a program:

**char prose[ ] = “to be or not to be that is the question”**

* write a program that contains a ***function* Cap(&prose[0])** that will convert the first character of each word to a upper case letter
* Hints:
  + Note that the definition of a string is a **NULL terminated** character array
  + Using a loop (while) search for the NULL value to terminate the search
  + Remember that the numerical difference between a lower case and upper case letter is 32 (a 🡺 97 and A 🡺 65)
  + Recall that each new word is preceded by a blank symbol (32) or a tab symbol (09)