**BTP100 Midterm Review**

1. How many bits are in the following:
   * A byte.
   * A word.
2. What are the following in hexadecimal:
   * 8
   * 16
   * 24
   * 32
3. Name the four most common type of the C language for performing arithmetic calculations.
4. Size specifiers adjust the size of the int and double types. Name three size specifiers.
5. What type qualifier specifies that a variable is unmodifiable.
6. Which variable type has a wider range of values, int or float? Why?
7. What is casting?
8. What is a structured program?
9. Professionals in the field of human-computer interaction confirm that layout and arrangement affects comfort and accessibility.  Poorly laid out code frustrates and promotes misreadings. Name five layout tools at our disposal.
10. What are magic numbers and how do we avoid them?
11. Compare and contrast syntactic errors with semantic errors.
12. Compare and contrast black box tests with white box tests.
13. Compare and contrast a structure with an object of that structure.
14. Compare and contrast a while loop with a do-while loop.
15. Compare and contrast parallel arrays with structures.
16. Convert the following while-loop into a for-loop:

int i = 0;

while (i < 100) {

printf("Hello\n");

++i;

}

1. Convert the following for-loop into a while-loop:

for (i = 100; i > 0; --i) {

printf("Hello\n");

}

1. Fill in the blanks in the following code.

//bmi.c - Midterm review question for filling out the code

// This program asks the user for his/her name, height in meters

// and weight in kgs. It then calculates the body mass index (bmi)

// according to the formula: bmi = height/(weight\*weight)

// If the bmi is higher than 30.0, that person is considered obese.

// To end the program, enter 0 for name. The program then reports

// how many users there are and how many are obese.

A sample run is as follows:

Enter the user's name: Luke

Enter Luke's height in meters: 1.80

Enter Luke's weight in kg: 90

Enter the user's name: Lorelei

Enter Lorelei's height in meters: 1.70

Enter Lorelei's weight in kg: 80

Enter the user's name: Mike

Enter Mike's height in meters: 1.80

Enter Mike's weight in kg: 110

Mike is obese

Enter the user's name: Ming

Enter Ming's height in meters: 1.60

Enter Ming's weight in kg: 70

Enter the user's name: 0

There is/are 4 users and 1 is/are obese

//bmi.c - Midterm review question for filling out the code

// This program asks the user for his/her name, height in meters

// and weight in kgs. It then calculates the body mass index (bmi)

// according to the formula: bmi = height/(weight\*weight)

// If the bmi is higher than 30.0, that person is considered obese.

// To end the program, enter 0 for name. The program then reports

// how many users there are and how many are obese.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#define MAX\_STR 32

int main(void)

{

int numUsers = 0;

int numObeseUsers = 0;

char name[MAX\_STR];

float height, weight, bmi;

int done = 0;

do {

printf("Enter the user's name: ");

scanf("\_\_\_\_\_\_\_\_", \_\_\_\_\_\_\_\_\_\_\_\_\_);

if (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_) { //end the program if '0' is entered

done = 1;

}

else {

//prompt the user for the height and weight

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_", \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);

scanf("%f", \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_", \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);

scanf("%f", \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);

//bmi equals the height divided by the weight squared

bmi = weight / (height \* height);

if (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_) {//test for obesity, bmi greater than 30

printf("%s is obese\n", name);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; //increment the obesity count

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; //increment the user count

printf("\n");

}

} while (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);

printf("There is/are \_\_\_ users and \_\_\_ is/are obese\n", \_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_);

return 0;

}

1. Evaluate the following:

const double e = 2.71828;

printf("The constant e is %d %.2lf\n", (int)e, e);

1. Evaluate the following:

const int x = 16;

const int y = 5;

printf("x/y is %d\n", x / y);

1. Evaluate the following:

const int z = 16;

const int w = 4;

printf("z/w is %.3f\n", (float)z / w);

1. Evaluate the following:

int num1 = 92, num2 = 94;

int result = (num2 < num1) ? (num2 - num1) \* 3 : (num2 - num1) \* 2;

printf("result is: %d\n", result);

1. Evaluate the following:

int num3 = 2, num4 = 4;

int res = (num4 > num3) ? num4 - num3 \* 3 : num4 - num3 \* 2;

printf("res is: %d\n", res);

1. Evaluate the following:

const char college[] = "Seneca College";

printf("%c%c%c\n", college[3], college[4], college[5]);

1. What is the exact output of the following program?
   * Record your work using the walkthrough table provided on the next page for the below code.
   * Put your output answer in the “Screen Output” box on the next page.

1//Walkthrough.c - Midterm review on walk-through tables

2

3//Miguel Watler

4//miguel.watler@senecacollege.ca

5//1234567890

6//Section XXY

7#define \_CRT\_SECURE\_NO\_WARNINGS

8#include <stdio.h>

9#define NUM 5

10#define PASS\_GRADE 50

11

12int main(void)

13{

14 //Midterm marks for five courses

15 int midTerm[NUM] = { 23, 34, 41, 33, 40 };

16 //Final exam marks for five courses

17 int finalExam[NUM] = { 26, 30, 40, 12, 45 };

18 int finalMark[NUM] = { 0 };

19 int numFailed = 0;

20 int numPassed = 0;

21

22 for (int i = 0; i < NUM; ++i) {

23 finalMark[i] = midTerm[i] + finalExam[i];

24 if (finalMark[i] < PASS\_GRADE) {

25 ++numFailed;

26 }

27 else {

28 ++numPassed;

29 }

30 }

31

32 printf("This student has passed %d course(s) and failed %d course(s).\n",

numPassed, numFailed);

33

34 return 0;

35}

**Walkthrough Table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Line # | i | midterm[i] | finalExam[i] | finalMark[i] | numFailed | numPassed |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**Screen Output:**

1. A Complete Program.

Create a structure called PlayerInfo for the Stanley Cup champions St. Louis Blues. The structure should contain the following:

* + Name (up to 31 characters plus the NULL character)
  + Goals (integer)
  + Assists (integer)
  + Points (integer)
  + Games played (integer)
  + Points per game (float)

Prompt the user to enter the last name, the number of goals scored, the number of assists, and the number of games played for four St. Louis Blues. The program will then calculate the number of points for that player according to the formula: **Points = Goals + Assists**, and then calculate the points per game for that player according to the formula: **PointsPerGame = Points/Games**. The statistics for some players from last season are:

// Name Games Goals Assists Points

//O'Reilly 82 28 49 77

//Tarasenko 76 33 35 68

//Schenn 72 17 37 54

//Perron 57 23 23 46

//Pietrangelo 71 13 28 41

//Bozak 72 13 25 38

Once the data has been entered, the program will display each player’s last name, the number of games played, the number of points, and the points per game to three decimal places. Then the program will display the name of the player with the maximum points per game.

Format all your printf’s and scanf’s according to the sample run on the following page.

Enter the player's last name: O'Reilly

Enter the number of goals: 28

Enter the number of assists: 49

Enter the number of games: 82

Enter the player's last name: Tarasenko

Enter the number of goals: 33

Enter the number of assists: 35

Enter the number of games: 76

Enter the player's last name: Schenn

Enter the number of goals: 17

Enter the number of assists: 37

Enter the number of games: 72

Enter the player's last name: Perron

Enter the number of goals: 23

Enter the number of assists: 23

Enter the number of games: 57

O'Reilly played 82 games with 77 points and 0.939 points per game

Tarasenko played 76 games with 68 points and 0.895 points per game

Schenn played 72 games with 54 points and 0.750 points per game

Perron played 57 games with 46 points and 0.807 points per game

O'Reilly has the most points per game with 0.939