# Code Writing

1. Declaring and initializing a struct
   1. Declare a struct data-type called “country” that contains 3 data members:
      1. double: population
      2. int: numStates
      3. string: continent
   2. use the “typedef” declaration on the struct to create an alias called “cty”
   3. create a new variable of the type “cty” name “unitedStates”
   4. Use the dot operator to initialize the data members:
      1. 329260000.00 for population
      2. 50 for numStates
      3. “North America” for continent
2. Using the struct pointer short-hand (i.e. (\*). Is the same as ->), remember that you only use this when you try to access data members through pointers
   1. Create a struct pointer called “australia” that is of type cty
   2. Use the arrow operator to initialize the data members:
      1. 25390000.00 for population
      2. 6 for numStates
      3. “Australia” for continent
3. Function pointers
   1. Create a string variable called news
   2. Create a function pointer called Kobe with:
      1. A string return type
      2. Takes three string parameters
   3. Write a function that can be pointed to by the Kobe function pointer
   4. Set news equal to the function pointer with the passed values “Kobe,” “GOAT,” and “RIP”
   5. Print what is returned by the function pointer
4. Working with switch statements
   1. Write a program that prompts a user for an integer
   2. The logic in the program is a switch statement that prints the month when a number is entered (e.g. 1 is January, 2 is February, etc)
   3. Make sure that the program only prints one letter
   4. Make a default statement where it says “Invalid entry”
5. Write the code to compile and run main.cpp, functionality.cpp, things.h files in the command line
6. Which is more efficient way to pass parameters? Pass by reference or pass by value? Why?
7. What does the “typedef” keyword mean? Give an example with a struct.

# Code Writing Answers

1.

#include <string>

using namespace std;

//this would go in a header file

typedef struct country {

int population;

int numStates;

string continent;

} cty;

//this would go in the main function

//one way to do it

cty unitedStates;

unitedStates.population = 329260000.00;

unitedStates.numStates = 50;

unitedStates.continent = “North America”;

//another way to do it

cty unitedStates = {329260000.00,50,”North America”};

2.

//this would go in the main function

cty\* australia = (cty\*)malloc(sizeof(cty));

//C++: cty\* australia = new cty;

australia->population = 25390000.00;

australia->numStates = 6;

australia->continent = “Australia”;

//C++: delete cty;

free(australia);

3.

//must use these includes and libraries for it to work

#include <stdio.h>

#include <string>

using namespace std;

//tribute to one of the greats

string basketballPlayer (string a, string b, string c){

printf(“%s is a %s! May %s %s! \n”,a.c\_str(),b.c\_str(),a.c\_str(),c.c\_str());

return “Live life to the fullest!“;

}

//instead of using the C function to convert to strings (.c\_str()), you are better off using cout and #include <iostream> from c++

int main(){

string news;

string (\*Kobe)(string,string,string) = basketballPlayer;

news = Kobe(“Kobe”,“GOAT”,”RIP”);

printf(“%s”,news.c\_str());

}

4.

//taking an integer

int a;

//C++: you can use cout, and cin

printf(“Please enter a month number : “);

scanf(“%d”,&a);

switch (a){

case 1:

printf(“January”);

break;

case 2:

printf(“February”);

break;

case 3:

printf(“March”);

break;

//and so fourth until 12

//.

//.

//.

default:

printf(“You entered an invalid number”);

}

5.

g++ -c main.cpp functionality.cpp -Wall –o main.o functionality.o

g++ main.o functionality.o –o out

./a.out

6.

Pass by reference means the address location of a variable in memory is passed to a function; there is only ever one copy. Pass by value actually creates a copy in memory and now you have 2 copies, In a big program, this can cause major over head; especially when copying linkedlist or calling a function multiple times. Therefore, pass by reference is more efficient when passing parameters to a function. (Answers may vary)

7.

It creates an alias, or in other words, another name that a data type can be called.

//using the typedef keyword

typedef struct Coding {

string language;

int\* intSize;

string\* stringSize;

} NewName;

# Multiple Choice

1. Which answer will print out “Hello World” to the console?
   1. cout << “Hello World” << endl;
   2. cout << “Hello World” <<;
   3. cout << Hello World << endl;
   4. cin >> “Hello World”;
2. What library allows a program to print to the console (cout, cin)?
   1. <iomanip>
   2. <fstream>
   3. <cstdlib>
   4. <iostream>
3. Which is **NOT** a preprocessor directive?
   1. #include
   2. #endif
   3. #ifndef
   4. #define
4. Linkedlists are created through what?
   1. An array
   2. A vector
   3. A struct
   4. A string
5. The keyword “const” does what?
   1. Allows values to be changed after declaration
   2. Lets a variable to be used within a function
   3. Restricts a value from being changed for the rest of the program
   4. Permits a variable to be global
6. Which for loop goes through 30 iterations?
   1. for(int i = 0; i < 30; i++){}
   2. for(int i = 0; i <=30; i++){}
   3. for(int i = 0; i < 30; i = i+2){}
   4. for(int i = 0; i < 29; i++){}
7. The “%d” in the printf() function is a string specifier?
   1. True
   2. False
8. In a switch statement, the case can be a conditional statement.
   1. True
   2. False
9. Which is the correct syntax for an if statement
   1. else – if
   2. if else – else
   3. if – if else – else
   4. if – else if – else
10. a void \* can be assigned to a pointer to any type and a pointer to any type can be assigned to a void \*
    1. True
    2. False
11. Is the following code snippet valid?

int arr[3] = {1,2,3};

int\* ptr = arr;

* 1. Yes
  2. No

1. Which pass-by type allows the programmer to change the **actual value** of a variable when passing it through a function?
   1. pass by value
   2. pass by number
   3. pass by reference
   4. pass by string
2. If this were a command line prompt, what is argv[3]?

./a.out Luke I am your father

* 1. Luke
  2. father
  3. ./a.out
  4. am

1. Which is a correct function declaration that returns void and takes in an integer function pointer with two integer parameters?
   1. string randomFunc( int (\*name)(int,int));
   2. void randomFunc( int (name)(int,int));
   3. void randomFunc( int (\*name)(int,int));
   4. int randomFunc( int name(int\*,int\*));
2. Which are properties of a linkedlist? (Select all that apply)
   1. Contiguous in memory
   2. Linked together by pointers
   3. Items can be inserted or deleted at any point in the linkedlist chain
   4. Are dynamically allocated data structures
3. The end of a LinkedList is found when the last pointer points to \_\_\_\_\_\_\_\_\_\_\_\_\_.

# Multiple Choice Answers

1. a 2. d 3. a 4. c 5. c 6. a 7. b 8. b 9. d 10. a 11. a 12. c

13. d 14. c 15. (b,c,d) 16. nullptr or NULL