

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_  
CS111 Summer Term 2 - Final 6/26/18

Section I Basic Questions 1,2,3 are required to be correctly answered to obtain a grade higher than C-. Part 1 Questions are worth 10 points each.

1)

// counts number of items in array that are greater than specified value

// {1,2,3,4,5} greaterThan = 5 ==> 0 greaterThan = 3 ==> 2

```
int CountNumbersGreaterThan(int arr[], int size,int greaterThan)
```

```
{  
    int count = 0;  
    for (int i = 0; i < size; i++)  
        if (arr[i] > greaterThan)  
            count++;  
    return(count);  
}
```

2)

// returns the string length

// "ABC" ==> 3 "" ==> 0

```
int StringLength(char arr[])
```

```
{  
    int i = 0;  
    while (arr[i])  
        i++;  
    return(i);  
}
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_

CS111 Summer Term 2 - Final

6/26/18

3)

// appends string2 to string 1

// "ABC" "def" ==&gt; "ABCdef"

void Append(char str1[], char str2[])

```
{
    int i = 0;
    while (str1[i])
        i++;
    int j = 0;
    while (str2[j])
    {
        str1[i] = str2[j];
        i++;
        j++;
    }
    str1[i] = 0;
}
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_

CS111 Summer Term 2 - Final

6/26/18

## SECTION II 7 Points each

```
// NYCClothingTax - add up the purchases and then calculate the tax
```

```
// items greater than 109.99 pay a 8.875% rate
```

```
// so $60 and $80 pay nothing
```

```
// $40 $80 $200 pays 17.75
```

```
//// Q4
```

```
//double NYCClothingTax(double items[], int size)
```

```
//{
```

```
//    double sum = 0;
```

```
//    for (int i = 0; i < size; i++)
```

```
//        if (items[i] > 109.99)
```

```
//            sum += items[i];
```

```
//    return(sum*.08875);
```

```
//}
```

4)

```
// a perfect number is a number where all of its factors besides the
```

```
// number itself add up to the number
```

```
// 6 ==> 1+2+3 == 6 → true
```

```
// 12 1,2,3,4,6 == 16 → false
```

```
// 28 ==> 1,2,4,7,14 == 28 → true
```

```
bool IsPerfectNumber(int n)
```

```
{
```

```
    int sumDivisors = 0;
```

```
    for (int i = 1; i < n; i++)
```

```
        if (n%i == 0)
```

```
            sumDivisors += i;
```

```
    return(sumDivisors == n);
```

```
}
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_  
CS111 Summer Term 2 - Final 6/26/18

5)

// what does "MysteryFunction" function below do?  
// for example what would the output for 12345 be?

ANSWER\_ It reverses the digits 12345 → 54321

```
int MysteryFunction(int num)
{
    int remainder = 0, newNumber = 0;
    while (num)
    {
        remainder = num % 10;
        num = num / 10;
        newNumber = newNumber * 10 + remainder;
    }
    return(newNumber);
}
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_

CS111 Summer Term 2 - Final

6/26/18

6)

```
// a double dimensional array is initialized to
// two rows of 12345
// use these first two rows to calculate 10 new rows of 5 numbers where each new row
// is the sum of the two rows before it e.g. the first three new rows are
// 2,4,6,8,10
// 3,2,9,12,15
// 5,6,15,20,25
// calculate the sets starting at the third row and then
// print out the results separated by a space between each number
// and a new line after each row
void FillDDArray()
{
    // first two rows initialized
    int arr[12][5] = { { 1,2,3,4,5 }, { 1,2,3,4,5 } };
    // first two rows initialized
    int arr[12][5] = { { 1,2,3,4,5 }, { 1,2,3,4,5 } };
    for (int i = 2; i < 12; i++)
        for (int j = 0; j < 5; j++)
            arr[i][j] = arr[i - 1][j] + arr[i - 2][j];

    for (int i = 0; i < 12; i++)
    {
        for (int j = 0; j < 5; j++)
            cout << arr[i][j] << " ";
        cout << "\n";
    }
}
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_

CS111 Summer Term 2 - Final

6/26/18

7)

```
// this function works for all possible values of int
// so {-110,-150} ==> min == -150 max = -110 average -130
//{ 110,150 } == > min == 110 max = 150 average = 130
// you can only use one loop!
void ArrayStatistics(int arr[], int size, int &min, int &max, int &average)
{
    min = arr[0];
    max = arr[0];
    int sum = arr[0];

    for (int i = 1; i < size; i++)
    {
        if (min > arr[i])
            min = arr[i];
        if (max < arr[i])
            max = arr[i];
        sum += arr[i];
    }
    average = sum / size;
}
}
```

8)

```
// checks an array for a duplicate value
// {1,2,1} → true {1,2,3} → false
bool IsDuplicate(int arr[], int size=10)
```

```
{
    for (int i = 0; i < size; i++)
        for (int j = 0; j < size; j++)
            if (i != j && arr[i] == arr[j])
                return(true);
    return(false);
}
```

```
// optimized answer (from one of the students)
```

```
for (int i = 0; i < size; i++)
    // we never have to check the element before where "i" currently is
    // because if there was a duplicate we would of already found it
    for (int j = i+1; j < size; j++)
        if (i != j && arr[i] == arr[j])
            return(true);
return(false);
}
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_

CS111 Summer Term 2 - Final

6/26/18

9)

```
// you will need the strlen function which
```

```
// returns the length of string
```

```
// "ABC" → "CBA" "A" → "A"
```

```
void ReverseString(char str[])
```

```
{
```

```
    for (int i = 0, j = strlen(str) - 1; i < strlen(str) / 2; i++, j--)
```

```
    {
```

```
        char temp = str[i];
```

```
        str[i] = str[j];
```

```
        str[j] = temp;
```

```
    }
```

```
// simpler answer (from one of the students)
```

```
for (int i = 0, j = strlen(str) - 1; i < j; i++, j--)
```

```
{
```

```
    char temp = str[i];
```

```
    str[i] = str[j];
```

```
    str[j] = temp;
```

```
}
```

```
}
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_  
 CS111 Summer Term 2 - Final 6/26/18

10)

// returns true if strings are equal false otherwise

bool StringCompare(string str1, string str2)

```
{
    return(str1 == str2);
}
```

// The array below consists of a bunch of names

// return a random name consisting of a random first and last name

// Make sure the First and Last names are not the same

// you will probably need the StringCompare Function above

// good output → newName = "John Doe"

// bad output → newName = "James James"

void RandomName(string str[],int numberOfNames,string &newName)

```
{
    srand(time(0));
    string firstName;
    string lastName;
    do
    {
        firstName = str[rand() % numberOfNames];
        lastName = str[rand() % numberOfNames];

    } while (StringCompare(firstName, lastName));
    newName = firstName + " " + lastName;

    // alternate answer from students
    srand(time(0));
    string firstName= str[rand() % numberOfNames];
    string lastName = firstName;
    while(StringCompare(firstName, lastName))
        lastName = str[rand() % numberOfNames];

    newName = firstName + " " + lastName;

    // alternate answer from students
    srand(time(0));
    int first = rand() % numberOfNames;
    int last = rand() % numberOfNames;
    while (StringCompare(str[first], str[last]))
        last = rand() % numberOfNames;

    newName = str[first] + " " + str[last];
}
```



Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_

CS111 Summer Term 2 - Final

6/26/18

// SECTION III 2 points each WHAT IS THE OUTPUT?

int i = 16;

int f1(int i){ return i / 2; }

double f1(double i){ return i / 2;}

int f3(){return(i /= 2);}

int main()

{

for (int i = 0, j = 3; i &lt; 4; i++, j--)

if (i == j)

cout &lt;&lt; i;

11) None I and j are never equal

int i = 3;

double d = 5;

cout &lt;&lt; f1(i) &lt;&lt; "\n";

12) 1

cout &lt;&lt; f1(d);

13) 2.5

cout &lt;&lt; f1(f3());

14) 4

cout &lt;&lt; f3();

15) 4

{

double d = 10.0;

for (int i = 0; i &lt; 4; i++)

d /= 10;

cout &lt;&lt; d;

}

16) .001

{

int k = 10;

for (int i = 0; i &lt; 4; i++)

k /= 10;

cout &lt;&lt; k;

}

17) 0

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_

CS111 Summer Term 2 - Final

6/26/18

Section IV - Fix the code below so that it works correctly

One Point each there are at least 5 mistakes

18..Q22

```
// make d less than 1
double d = 10.0;
while (d) // while (d>1)
    d /= 10;

char str[4] = "1234"; // char str[5] = "1234";
int num[5] = { 1,2,3,4,5 };
int i; // int i = 0;
//output the elements of the string
while (str[i])
{
    cout << str[i] << "\n";
    i++;
}

// output the 5 elements of the array
for (int j = 1; j <= 5; j++) // (int j = 0; j <5; j++)
    cout << num[j];

int arr[10];
//fill the array from 10 to 1
for (int i = 10; i >0; i++)
    arr[i] = 11-i;

for (int i = 9; i >=0; i--)
    arr[i] = 10-i;
```

Name: \_\_\_\_\_ CUNYfirstID: \_\_\_\_\_  
CS111 Summer Term 2 - Final 6/26/18

THIS PAGE LEFT BLANK FOR EXTRA CODE