|  |  |  |
| --- | --- | --- |
|  | MAKE-UP LAB #3  Functions |  |

This make-up lab exercise is **open book/open notes** and an **individual effort**; however, collaboration is permitted with the four 2/c TOOP Assistants. This make-up lab will allow you to earn up to half the points you missed on Lab #3 (I will take the average of both scores). Follow the comments and write the structure of the described program. You must complete this make-up lab in Code::Blocks. Submit via GitHub and e-mail me that you attempted it by 0800 on Thursday, 30 September.

|  |
| --- |
| #include <iostream>  // Define a function called isEvenlyDivisible.  // It will take a divisor and dividend  // It will return true if the divisor can evenly divide the dividend, false otherwise.  bool isEvenlyDivisible(int divisor, int dividend) {  }  // Define a function called isPrime that receives a single positive number.  // It will return true if the given number is prime, false otherwise.  // Consider that a prime number divided by any natural number will have a nonzero remainder.  // NOTE: All you need to do is call isEvenlyDivisible in this function and pass in the number being  // tested and the current value being tested against (j).  bool isPrime(int num) {  bool numIsPrime = true;  for (int j = 2; j < num && numIsPrime; j++) {  numIsPrime = !\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_, \_\_);  }  return numIsPrime;  }  // Define a function that takes a positive integer "a" and returns the number of primes between 1 and a.  // Call isPrime in this method.  // return -1 on invalid input (negative number, for example)  int numPrimes(int a) {  }  int main() {  } |