1. (0,5p) Declarați, apoi implementați o funcție care calculează cifra de control pentru un cod ISBN-10. Indicații la <https://en.wikipedia.org/wiki/International_Standard_Book_Number>
2. (0.25p) Scrieti o funcție care testează dacă un an este bisect.  
   *bool isLeapYear(unsigned short year);*
3. (0.5p) Scrieți o funcție care pentru o zi (an luna zi) va returna numărul zilei din săptamană.  
   *unsigned char dayOfTheWeek(unsigned short year,  
   unsigned char month,  
   unsigned char day);*
4. (0.20p) Scrieți o funcție care verifică dacă un număr este palindrom. Un număr este palindrom dacă citit de la dreapta la stanga este egal cu numărul citit de la stânga la dreapta. Ex: 13231 este palindrom dar 12331 nu este palindrom  
   *bool isPalindrom(unsigned long long number);*
5. (0.25p) Scrieți o funcție care, pentru un număr natural, calculeză suma cifrelor reprezentării lui binare.  
   *unsigned char sumBinaryFigure(unsigned long long number);*
6. (0.25p) Scrieti o funcție care pentru un număr *n* returnează al *n*-lea număr din șirul lui Fibonnaci.  
   Ex: fib(3) = fib(2) + fib(1) = fib(1) + fib(0) + 1 = 1 + 0 + 1 = 2  
   *unsigned int fibonnaci(int index);*
7. (0.30p) Spunem că un număr natural este perfect dacă este egal cu suma divizorilor săi strict mai mici decât el. Scrieți o funcție care returnează suma celor mai mari două numere perfecte mai mici decat un număr (number >= 30) dat.  
   *unsigned long perfectNumbers(unsigned int number);*
8. (0.4p) Scrieți o funcție care pentru un interval închis dat [left, right] returnează numărul de numere ce au număr maxim de divizori primi.  
   Ex: Pentru  [30, 45] numerele cu număr maxim de divizori primi = 30,42 (au cate 3 divizori primi, nici un alt număr din interval nu are mai mulți   divizori primi) iar functia va returna 2.  
   *unsigned short primeDivisors(unsigned int left, unsigned int right);*
9. (0.4p) Două numere naturale impare consecutive și prime se numesc numere prime gemene.Scrieți o funcție care determină primele *count* perechi de numere prime gemene mai mari decât un *lowerBound* dat.  
   Ex: Pentru count = 1 și lowerBound = 2 funcția va returna o matrice cu un rând (un rezultat) și 2 coloane (pereche de numere – două numere) 3 5  
   *matrix primeTwins(unsigned int count, unsigned int lowerBound);*
10. (1p) Se consideră un șir de *n* intervale închise întregi. Două intervale consecutive în șir care au intersecția nevidă se reunesc și se înlocuiesc în șir cu intervalul reuniune. Operația se repetă până când nu mai sunt în șir două intervale consecutive cu intersecția nevidă. Să se determine câte intervale există în șir după realizarea acestor operații.