|  |  |  |
| --- | --- | --- |
|  | *Софийски университет „Св. Климент Охридски“ Факултет по математика и информатика* |  |

Домашна 3

*курс Увод в програмирането*

*за специалност Компютърни науки*

*зимен семестър 2019/2020 г.*

Правила

Следните правила описват процеса по реализирането и предаването на домашните по Увод в програмирането.

1. Всеки студент има право да избере да реши различен брой задачи.
2. Срок за предаване на Домашна 3: 12.01.2020г.
3. За отлична оценка трябва да предадете **първите три** задачи.
4. Бонус задачата носи допълнителни точки и може да бъде предадена вместо първа и трета задача.
5. По домашната се работи самостоятелно (т.е. не се допуска работа в екипи)
6. Не губите нищо, ако предадете частично направена домашна! По-добре се опитайте да решите поне една задача, отколкото да си кажете, че не можете!
7. Плагиатство от колеги и от други източници води до анулиране на работата.
8. Предаването става чрез прикачване на ZIP архив към съответното задание в Moodle, който съдържа всички файлове, необходими за компилирането на задачите от домашната. Архивът трябва да бъде в описания в заданието формат.
9. Основният критерий при оценяването на домашните ще бъде доколко решенията успяват да изпълнят поставеното условие.
10. Другите критерии при оценяването са доброто стилизиране и форматиране на кода и използването на показаните добри практики за писане на код.

**Задача 1.**

Аня е много ученолюбива и иска да посети възможно най-много лекции в университета. И тук се появява следният проблем - някои от тях се припокриват, а тя все още не е открила начин да е на две места едновременно.

Аня разполага със списък от началните и крайните часове на N лекции, зададени в минути от началото на деня. Аня познава мястото добре и времето за придвижване между различните зали е пренебрежимо малко.

На стандартния изход да се изведе максималният **брой** лекции, на които Аня може да присъства.

**Пример:**

**Вход:**

11

1 5

6 7

3 6

3 5

2 10

7 10

13 16

11 12

1 8

1 6

8 10

**Изход**:

5

**Обяснение:**

*\* Не са зададени предварителни ограничения за N.   
Използвайте динамично заделяне на памет.*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|  | | | | |  | |  | | |  | |  | | | |
|  |  |  | | |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | | | |  | | | |  |  |  |  |  |  |
|  |  | | | | | | | | |  |  |  |  |  |  |
|  | | | | | | | |  |  |  |  |  |  |  |  |
|  | | | | | |  |  |  |  |  |  |  |  |  |  |

**Задача 2.**

В сезона на празниците Дядо Коледа е закачил коледни лампички из цяла Лапландия. Те светват и изгасват по следния алгоритъм:

* В даден момент само една лампичка променя състоянието си (светва/изгасва)
* Лампичка №1 може винаги да променя състоянието си
* Лампичката, която се намира след първата вече светеща лампичка, може да променя състоянието си
* Ако всички лампички светнат, всички изгасват и алгоритъмът започва отново.

Сега джуджетата се чудят, следвайки този алгоритъм, в какъв ред биха светнали поредица от N на брой лампички.

На стандартния ред да се изведе поредица от <знак><номер>, където <номер> e число между 1 и N и означава номер на лампичка, а <знак> е **‘+’ или ‘-’,** ако поредната лампичка се светва или се изгася.

**Пример:**

**Вход:**

3

**Изход:**

+1

+2

-1

+3

+1

*\* ЗАДАЧА от НОИ 2014. Използвайте рекурсия*

**Задача 3.**

Ванката има да напише проект - текстообработваща програма. Той е почти готов, но в последния момент се оказва, че освен да въвежда и запазва текст, трябва да намери и бърз начин да го редактира. Той трябва да може да премахва всички срещания на дадена дума и/или да я заменя с нова. И понеже времето го притиска, се обръща към вас за помощ.

Да се напише функция remove. Функцията приема низ (*source)* и дума (*word)* и връща нов низ, образуван от *source* чрез премахване на всички срещания на *word*.

Да се напише функция *replace*. Функцията приема низ (*source)* и думи (*oldWord,* *newWord)* и връща нов низ, образуван от *source* чрез замяна на всички срещания на *oldWord* с *newNord*.

**Пример за употребата на *remove*:**

**Параметри:**

I love math cats and programming.

cats

**Резултат:**

I love math and programming.

*\* Между думите трябва да има само един разделител*

**Пример за употребата на *replace*:**

**Параметри:**

I love cats, birds and stones.

stones

dogs

**Резултат**:

I love cats, birds and dogs.

**Бонус задача**

**Grayscale BMP**

Да се реализира функция

bool bmp\_to\_grayscale(const char \* name),

която приема име на картинка и я преобразува в черно-бяла.  
Форматът на картинката е 24-bit true color bmp (bottom - top).

Функцията връща истина, ако превръщането е било успешно, и лъжа при неуспех на някоя входно-изходна операция с файла или при несъответствие на метаданните за размерът на изображението.

**Формат**

Картинката представлява двоичен файл в 2 части (метаданни и изображение).

**Метаданни**

Необходимите метаданни за тази задача са:

* Размер на файла - 4 байтово неотрицателно цяло число, което започва от 2-ра позиция във файла.
* Начало на изображението - 4 байтово неотрицателно цяло число, което започва от 10-та позиция във файла.
* Широчината на изображението в пиксели - 4 байтово неотрицателно цяло число, което започва от 18-та позиция във файла.
* Височината на изображението в пиксели - 4 байтово неотрицателно цяло число, което започва от 22-ра позиция във файла

**Изображение**

Изображението започва от позицията записана на 10-та позиция в метаданните.

Един ред от файла се състои от толкова пиксели, колкото е широчината в метаданните + допълване до число делящо се на 4.

Един пиксел се състои от 3 неотрицателни еднобайтови числа, които определят стойността на синия, зеления и червения цвят (в този ред).

В изображението има толкова реда, колкото е височината записана в метаданните.

**Допълнителни материали**

Прехвърляне на пиксел от цветен до черно-бял: https://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/

Инструмент за превръщане на .bmp изображения в дадения формат: https://online-converting.com/image/convert2bmp/