## Домашна работа (поправителна сесия)

Разглеждаме абстрактен базов клас *Множество*, което може да съдържа елементи от произволен тип, но всички елементи в него са от един и същ тип. За едно множество изискваме единствено операция за проверка дали даден елемент принадлежи на множеството.

Наследници на това множество са следните конкретни класове:

* *Крайно множество*, чиито елементи се подават като масив в конструктора. Да се гарантира, че сред тях няма повторения! След създаване на тези елементи, те не се променят.
* *Множество по критерий* – в конструктора се подава предикат (булева функция или еквивалентен обект), който определя дали даден елемент принадлежи на множеството или не.
* *Сечение на множества* – в конструктора се подават едно или повече множества. Създаденият обект (сечение) съдържа даден елемент, когато този елемент се съдържа във всяко от посочените множества. Елементите на всички множества от сечението са от един и същ тип.
* *Обединение на множества* - в конструктора се подават едно или повече множества. Създаденият обект (обединение) съдържа даден елемент, когато този елемент се съдържа в поне едно от посочените множества. Елементите на всички множества от обединението са от един и същ тип.

Реализирайте програма, която прочита от текстов файл множество от символни низове – имена на файлове, всяко от които е разположено на отделен ред. Всеки от тези файлове е двоичен и съдържа две цели, 16 битови числа N и Т, следвани от N цели, 32 битови числа. N няма да надхвърля 32. Стойността на T определя какъв тип множество ще определят тези числа и как ще се добави то към резултата. Възможни стойности са:

0 – числата определят крайно множество. То се добавя с обединение към резултата.

1 – числата определят множество по критерий. Негови елементи са числата, които се делят на всяко от подадените числа. Това множество се добавя с обединение към резултата.

2 – числата определят множество по критерий. Негови елементи са числата, които се делят на някое от подадените числа. Това множество се добавя с обединение към резултата.

8 – числата определят крайно множество. То се добавя със сечение към резултата.

9 – числата определят множество по критерий. Негови елементи са числата, които се делят на всяко от подадените числа. Това множество се добавя със сечение към резултата.

10 – числата определят множество по критерий. Негови елементи са числата, които се делят на някое от подадените числа. Това множество се добавя със сечение към резултата.

От числата във всеки файл създайте съответното множество. Това множество добавете към резултатно множество, съгласно описаното във файла – обединение и съответно сечение с него. Операцията от първия файл се игнорира и винаги започвате с полученото от него множество. Започвате да прилагате операциите от втория файл към първия, след това от третия към резултата и така нататък. Подсигурете коректността на данните.

Въведете от клавиатурата две цели числа, описващи интервал. Изведете броя на числата в този интервал, които са в полученото множество.

Пример:

| Input.txt | съдържание на first.dat | съдържание на second.dat | съдържание на set.dat |
| --- | --- | --- | --- |
| first.dat second.dat set.dat | 5 0  1 3 4 6 2 | 5 0 1 10 12 16 2 | 3 10 2 3 5 |

Интервал: [0, 10]. Резултат: 5 (това са числата 2, 3, 4, 6, 10).

*Забележка* – Съдържанието на двоичните файлове е показано като текст само за удобство на примера. **Файловете трябва да са двоични!**