

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіти до комп'ютерних практикумів дисципліни

«Мультипарадигмене програмування»

Прийняв
ст. викладач кафедри ІІІ
Очеретяний О.К.

Виконав
Студент групи ІТ-02
Дідур М.А.

Київ – 2022

Лабораторна робота №2

Завдання

Ви напишете 11 функцій SML (і тести для них), пов'язаних з календарними датами. У всіх завданнях, **“дата”** є значенням SML типу `int*int*int`, де перша частина - це рік, друга частина - місяць і третя частина - день. **«Правильна»** дата має позитивний рік, місяць від 1 до 12 і день не більше 31 (або 28, 30 - залежно від місяця). Перевіряти **“правильність”** дати не обов'язково, адже це досить складна задача, тож будьте готові до того, що багато ваших функцій будуть працювати коректно для деяких/всіх **“неправильних”** дат у тому числі. Також, **«День року»** — це число від 1 до 365 де, наприклад, 33 означає 2 лютого. (Ми ігноруємо високосні роки, за винятком однієї задачі.)

1. Напишіть функцію `is_older`, яка приймає дві дати та повертає значення `true` або `false`. Оцінюється як `true`, якщо перший аргумент - це дата, яка раніша за другий аргумент. (Якщо дві дати однакові, результат хибний.)
2. Напишіть функцію `number_in_month`, яка приймає список дат і місяць (тобто `int`) і повертає скільки дат у списку в даному місяці.
3. Напишіть функцію `number_in_months`, яка приймає список дат і список місяців (тобто список `int`) і повертає кількість дат у списку дат, які знаходяться в будь-якому з місяців у списку місяців. **Припустимо, що в списку місяців немає повторюваних номерів.** Підказка: скористайтеся відповіддю до попередньої задачі.
4. Напишіть функцію `dates_in_month`, яка приймає список дат і число місяця (тобто `int`) і повертає список, що містить дати з аргументу **“список дат”**, які знаходяться в переданому місяці. Повернутий список повинен містити дати в тому порядку, в якому вони були надані спочатку.
5. Напишіть функцію `dates_in_months`, яка приймає список дат і список місяців (тобто список `int`) і повертає список, що містить дати зі списку аргументів дат, які знаходяться в будь-якому з місяців у списку місяців. Для простоти, припустимо, що в списку місяців немає повторюваних номерів. Підказка: Використовуйте свою відповідь на попередню задачу та оператор додавання списку SML (`@`).
6. Напишіть функцію `get_nth`, яка приймає список рядків і `int n` та повертає `n`-й елемент списку, де голова списку є першим значенням. Не турбуйтеся якщо в

списку занадто мало елементів: у цьому випадку ваша функція може навіть застосувати `hd` або `tl` до порожнього списку, і це нормально.

7. Напишіть функцію `date_to_string`, яка приймає дату і повертає рядок у вигляді "February 28, 2022". Використовуйте оператор `^` для конкатенації рядків і бібліотечну функцію `Int.toString` для перетворення `int` в рядок. Для створення частини з місяцем не використовуйте купу розгалужень. Замість цього використайте список із 12 рядків і свою відповідь на попередню задачу. Для консистенції пишіть кому після дня та використовуйте назви місяців англійською мовою з великої літери.
8. Напишіть функцію `number_before_reaching_sum`, яка приймає додатний `int` під назвою `sum`, та список `int`, усі числа якої також додатні. Функція повертає `int`. Ви повинні повернути значення `int n` таке, щоб перші `n` елементів списку в сумі будуть менші `sum`, але сума значень від `n + 1` елемента списку до кінця був більше або рівний `sum`.
9. Напишіть функцію `what_month`, яка приймає день року (тобто `int` між 1 і 365) і повертає в якому місяці цей день (1 для січня, 2 для лютого тощо). Використовуйте список, що містить 12 цілих чисел і вашу відповідь на попередню задачу.
10. Напишіть функцію `month_range`, яка приймає два дні року `day1` і `day2` і повертає список `int [m1,m2,...,mn]` де `m1` – місяць `day1`, `m2` – місяць `day1+1`, ..., а `mn` – місяць `day2`. Зверніть увагу, що результат матиме довжину `day2 - day1 + 1` або довжину 0, якщо `day1 > day2`.
11. Напишіть найстарішу функцію, яка бере список дат і оцінює параметр (`int*int*int`). Він має оцінюватися як `NONE`, якщо список не містить дат, і `SOME d`, якщо дата `d` є найстарішою датою у списку.

Код: <https://github.com/lucky-rydar/multiparadigm-lab2>

Результати:

```
smlnj main.sml

Standard ML of New Jersey (64-bit) v110.99.2 [built: Wed Sep 29 16:07:17 2021]
[opening main.sml]
datatype date = date of int * int * int
exception InvalidArg
val dat1 = date (2000,11,1) : date
val dat2 = date (2000,11,1) : date
val is_older = fn : date * date -> bool
[autoloading]
[library $SMLNJ-BASIS/basis.cm is stable]
[library $SMLNJ-BASIS/(basis.cm):basis-common.cm is stable]
[autoloading done]
main.sml:54.5-59.59 Warning: match nonexhaustive
      (0,_) => ...
      (_,nil) => ...
      (sum,first :: 1 :: last) => ...

true
val number_in_month = fn : date list * int -> int
val number_in_months = fn : date list * int list -> int
val dates_in_month = fn : date list * int -> date list
val dates_in_months = fn : date list * int list -> date list
val get_nth = fn : string list * int -> string
val date_to_string = fn : date -> string
val number_before_reaching_sum = fn : int * int list -> int
val what_month = fn : int -> int
val month_range_util = fn : int * int -> int list
val for_loop_util = fn : int * int -> int list
val month_range = fn : int * int -> int list
val oldest_date = fn : date list -> date
val l = [date (2000,11,1),date (2000,12,1),date (2000,11,2)] : date list
2 monthes were found
3 monthes were found
val found_dates = <hidden> : date list
val found_dates = [date (2000,11,1),date (2000,11,2),date (2000,12,1)] :
  date list
val nth_elem = "2" : string
val stringified_date = "February, 11, 2000" : string
val t8_list = [30,30,30,30,30,30] : int list
val t8_sum = 64 : int
val t8_res = 2 : int
val t9_res = 2 : int
val t10_res = [1,2] : int list
val t10_res1 = [1,2,3] : int list
val t11_l =
  [date (2000,11,1),date (2000,12,1),date (2001,1,1),date (2000,11,2)] :
  date list
val t11_res = date (2001,1,1) : date
- |
```