ProgTest ► BIE-PA1 (23/24 ZS) ► Homework 03 ► Work days   Logout

Work days

Submission deadline:	**2023-11-13 11:59:59**	1169458.376 se
Late submission with malus:	**2023-12-31 23:59:59** (Late submission malus: 100.0000 %)	
Evaluation:	**6.8750**	
Max. assessment:	**5.0000** (Without bonus points)	
Submissions:	7 / 20 Free retries + 10 Penalized retries (-10 % penalty each retry)	
Advices:	1 / 2 Advices for free + 2 Advices with a penalty (-10 % penalty each advice)	

The task is to implement two functions (not a whole program, just two functions) which handle working day computations. The required functions are:

`bool isWorkDay ( int y, int m, int d )`
> The function decides whether the date (given in the parameters) is a working day, or not. The parameters are year, month, and day to test. Return value is `true` if the date is a working day, or `false` otherwise (the date is Sunday, Saturday, a national holiday, or the date is invalid).

`TResult countDays ( int y1, int m1, int d1, int y2, int m2, int d2 )`
> The function shall count total days and working days in the given interval of days. The interval is given by the start and end dates. Start date is formed by year, month, and day `y1, m1,` and `d1`, end date is formed by `y2, m2,` and `d2`. The interval is considered closed, i.e., both boundary days are included into the computation. Return value is a structure with two fields: `m_TotalDays` (the total number of days elapsed) and `m_WorkDays` (the number of workdays in the interval). If the parameters are invalid (see below), the return value must set both fields to −1.

`TResult`
> a structure declared in the testing environment. Your implementation will use the structure, however, you are not allowed to change the declaration. The structure contains two fields:
> - `m_TotalDays` - the total number of days and
> - `m_WorkDays` - the number of work days.

The input parameters must define a valid date. To be valid, the input must satisfy the following restrictions:

- year must be greater than or equal to 2000 (i.e., all dates before 2000-01-01 are invalid),
- month is valid (1 to 12),
- day is valid (1 to the number of days in the month),
- in `countDays`: the end date must not precede the start date.

Submit a source file with the implementation of both required functions (`isWorkDay` and `countDays`). Further, the source file must include your auxiliary functions which are called from the required functions. The function will be called from the testing environment, thus, it is important to adhere to the required interface. Use the attached sample code as a basis for your development, complete the required functions and add your required auxiliary functions. There is an example `main` with some test in the attached code. These values will be used in the basic test. Please note the header files as well as `main` is nested in a conditional compile block (`#ifdef/#endif`). Please keep these conditional compile block in place. They are present to simplify the development. When compiling on your computer, the headers and `main` will be present as usual. On the other hand, the header and `main` will "disappear" when compiled by Progtest. Thus, your testing `main` will not interfere with the testing environment's `main`.

Your function will be executed in a limited environment. There are limits on both time and memory. The exact limits are shown in the test log of the reference. The time limits are set such that a correct implementation of the naive solution passes all mandatory tests. Thus, the solution may be awarded nominal 100%. The algorithm must be improved to pass the bonus tests and gain the extra points. There are long intervals tested in the bonus tests (years exceed 4000 several orders of magnitude).

Advice:

- Monday to Friday are considered working days. Moreover, national holidays are not considered working days. We use the following list of national holidays (actually the national holidays in the Czech Republic):

> Jan 1, May 1, May 8, Jul 5, Jul 6, Sep 28, Oct 28, Nov 17, Dec 24, Dec 25, and Dec 2

> Please note Easter is not included in the list. The computation of Easter is a bit more complicated (Moon phase is required), thus Easter is not used in this homework.

- We assume standard Gregorian calendar when counting days. Thus, there is a fixed number of days (30/31) in a month

with the exception of February. February is either 28 days (non-leap year) or 29 days (a leap year). The leap year rules of Gregorian calendar are:

1. years are not leap years in general,
2. except multiples of 4 which are leap years,
3. except multiples of 100 which are not leap years,
4. except multiples of 400 which are leap years,
5. except multiples of 4000 which are not leap years.

Thus, years 2001, 2002, 2003, 2005, ... are not leap years (rule #1), years 2004, 2008, ..., 2096, 2104, ... are leap year (rule #2), years 2100, 2200, 2300, 2500, ... are not leap years (rule #3), years 2000, 2400, ..., 3600, 4400, ... are leap years (rule #4), and years 4000, 8000, ... are not leap years (rule #5).

- Download the attached sample code and use it as a base for your development.
- The `main` function in your program may be modified (e.g., a new test may be included). The conditional compile block must remain, however.
- It is difficult to handle 3-tuples like (year, month, day). It is a good idea to convert the 3-tuple into some other representation, e.g., into a single integer.
- There will be a lot of computation done twice in the program: for the start and end timestamps. Thus, it is a good idea to prepare auxiliary functions and call them twice.
- The years in the mandatory tests do not exceed 2200.
- There is macro `assert` used in the example `main` function. If the value passed to `assert` is nonzero (true), the macro does nothing. On the other hand, if the parameter is zero, the macro stops the execution and reports line, where the test did not match (and shall be fixed). Thus, the program ends silently when your implementation passes the tests correctly.

Sample data:		**Download**
Submit:	Browse… **No file selected.**	**Submit**

☑ **Reference**

- **Evaluator: computer**
  - Program compiled
  - Test 'Zakladni test podle ukazky': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 0.000 s (limit: 2.000 s)
    - Mandatory test success, evaluation: 100.00 %
  - Test 'Test meznich hodnot': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.000 s (limit: 10.000 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Test osetreni parametru': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.003 s (limit: 10.000 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Test nahodnymi daty': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.002 s (limit: 9.997 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Test nahodnymi daty (dlouhe intervaly, bonus #1)': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 0.002 s (limit: 10.000 s)
    - Bonus test - success, evaluation: 125.00 %
  - Test 'Test nahodnymi daty (dlouhe intervaly, bonus #2)': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 0.010 s (limit: 9.998 s)
    - Bonus test - success, evaluation: 125.00 %
  - Overall ratio: 156.25 % (= 1.00 * 1.00 * 1.00 * 1.00 * 1.25 * 1.25)
- Total percent: 156.25 %
- Early submission bonus: 0.50
- Total points: 1.56 * (5.00 + 0.50) = 8.59

SW metrics:		Total	Average	Maximum	Function name
	Functions:	**12**	--	--	--
	Lines of code:	**192**	**16.00 ± 10.98**	**37**	fillTables
	Cyclomatic complexity:	**59**	**4.92 ± 2.36**	**9**	fillTables

---

7	2023-10-28 16:19:59	**Download**
Submission status:	Evaluated	
Evaluation:	6.8750	

- **Evaluator: computer**
  - Program compiled
  - Test 'Basic test with sample input data': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 0.001 s (limit: 2.000 s)
    - Mandatory test success, evaluation: 100.00 %
  - Test 'Borderline test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.001 s (limit: 10.000 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Invalid input test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.003 s (limit: 9.999 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.031 s (limit: 9.996 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test (long intervals, bonus #1)': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 1.852 s (limit: 10.000 s)
    - Bonus test - success, evaluation: 125.00 %
  - Test 'Random test (long intervals, bonus #2)': Abnormal program termination (Time limit exceeded)
    - Cumulative test time exceeded, killed after:: 8.157 s (limit: 8.148 s)
    - Bonus test - failed, evaluation: No bonus awarded
  - Overall ratio: 125.00 % (= 1.00 * 1.00 * 1.00 * 1.00 * 1.25)
- Advices used: 1
- Penalty due to advices: None (1 <= 2 limit)
- Total percent: 125.00 %
- Early submission bonus: 0.50
- Total points: 1.25 * (5.00 + 0.50) = 6.88

SW metrics:		Total	Average	Maximum	Function name
	Functions:	**11**	--	--	--
	Lines of code:	**252**	**22.91 ± 21.12**	**66**	is_holiday
	Cyclomatic complexity:	**72**	**6.55 ± 5.33**	**20**	get_working_days

---

6	2023-10-28 12:24:18	**Download**
Submission status:	Evaluated	
Evaluation:	6.8750	

- **Evaluator: computer**
  - Program compiled
  - Test 'Basic test with sample input data': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 0.001 s (limit: 2.000 s)
    - Mandatory test success, evaluation: 100.00 %
  - Test 'Borderline test': success

- result: 100.00 %, required: 25.00 %
- Total run time: 0.003 s (limit: 10.000 s)
- Optional test success, evaluation: 100.00 %
  - Test 'Invalid input test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.003 s (limit: 9.997 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.038 s (limit: 9.994 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test (long intervals, bonus #1)': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 1.776 s (limit: 10.000 s)
    - Bonus test - success, evaluation: 125.00 %
  - Test 'Random test (long intervals, bonus #2)': Abnormal program termination (Time limit exceeded)
    - Cumulative test time exceeded, killed after:: 8.230 s (limit: 8.224 s)
    - Bonus test - failed, evaluation: No bonus awarded
  - Overall ratio: 125.00 % (= 1.00 * 1.00 * 1.00 * 1.00 * 1.25)
- Advices used: 1
- Penalty due to advices: None (1 <= 2 limit)
- Total percent: 125.00 %
- Early submission bonus: 0.50
- Total points: 1.25 * (5.00 + 0.50) = 6.88

		Total	Average	Maximum	Function name
SW metrics:	Functions:	**11**	--	--	--
	Lines of code:	**245**	**22.27 ± 21.51**	**66**	is_holiday
	Cyclomatic complexity:	**70**	**6.36 ± 5.48**	**20**	get_working_days

5	2023-10-28 11:33:28	**Download**
Submission status:	Evaluated	
Evaluation:	5.5000	

- **Evaluator: computer**
  - Program compiled
  - Test 'Basic test with sample input data': success
    - result: 100.00 %, required: 100.00 %
    - Total run time: 0.001 s (limit: 2.000 s)
    - Mandatory test success, evaluation: 100.00 %
  - Test 'Borderline test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.001 s (limit: 10.000 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Invalid input test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.003 s (limit: 9.999 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.031 s (limit: 9.996 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test (long intervals, bonus #1)': failed
    - result: 96.97 %, required: 100.00 %
    - Total run time: 1.787 s (limit: 10.000 s)
    - Bonus test - failed, evaluation: No bonus awarded
    - Failed (invalid output) **[Unlock advice (70 B)]**
    - Failed (invalid output) **[Unlock advice (71 B)]**
    - Failed (invalid output) **[Unlock advice (71 B)]**
    - Failed (invalid output) **[Unlock advice (71 B)]**
    - Failed (invalid output) **[Unlock advice (69 B)]**
    - Failed (invalid output) **[Unlock advice (71 B)]**
    - Failed (invalid output) **[Unlock advice (69 B)]**

- - - ☐ Failed (invalid output)
  - ○ Test 'Random test (long intervals, bonus #2)': Abnormal program termination (Time limit exceeded)
    - ▪ Cumulative test time exceeded, killed after:: 8.219 s (limit: 8.213 s)
    - ▪ Bonus test - failed, evaluation: No bonus awarded
  - ○ Overall ratio: 100.00 % (= 1.00 * 1.00 * 1.00 * 1.00)
- Total percent: 100.00 %
- Early submission bonus: 0.50
- Total points: 1.00 * (5.00 + 0.50) = 5.50

SW metrics:		Total	Average	Maximum	Function name
	Functions:	**10**	--	--	--
	Lines of code:	**227**	**22.70 ± 21.78**	**66**	is_holiday
	Cyclomatic complexity:	**66**	**6.60 ± 5.70**	**20**	get_working_days

---

4	**2023-10-28 11:32:00**	**Download**
Submission status:	Evaluated	
Evaluation:	0.0000	

- **Evaluator: computer**
  - ○ Compile in 'basic' mode failed. **[Unlock advice (542 B)]**
- Total percent: 0.00 %
- Early submission bonus: 0.50
- Total points: 0.00 * (5.00 + 0.50) = 0.00

3	**2023-10-28 11:30:42**	**Download**
Submission status:	Evaluated	
Evaluation:	0.0000	

- **Evaluator: computer**
  - ○ Compile in 'basic' mode failed. **[Unlock advice (542 B)]**
- Total percent: 0.00 %
- Early submission bonus: 0.50
- Total points: 0.00 * (5.00 + 0.50) = 0.00

2	**2023-10-28 11:30:02**	**Download**
Submission status:	Evaluated	
Evaluation:	0.0000	

- **Evaluator: computer**
  - ○ Compile in 'basic' mode failed. **[Unlock advice (542 B)]**
- Total percent: 0.00 %
- Early submission bonus: 0.50
- Total points: 0.00 * (5.00 + 0.50) = 0.00

1	**2023-10-28 11:26:27**	**Download**
Submission status:	Evaluated	
Evaluation:	5.5000	

- **Evaluator: computer**
  - ○ Program compiled
  - ○ Test 'Basic test with sample input data': success
    - ▪ result: 100.00 %, required: 100.00 %
    - ▪ Total run time: 0.001 s (limit: 2.000 s)
    - ▪ Mandatory test success, evaluation: 100.00 %
  - ○ Test 'Borderline test': success
    - ▪ result: 100.00 %, required: 25.00 %
    - ▪ Total run time: 0.001 s (limit: 10.000 s)

- Optional test success, evaluation: 100.00 %
  - Test 'Invalid input test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.003 s (limit: 9.999 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test': success
    - result: 100.00 %, required: 25.00 %
    - Total run time: 0.030 s (limit: 9.996 s)
    - Optional test success, evaluation: 100.00 %
  - Test 'Random test (long intervals, bonus #1)': failed
    - result: 97.37 %, required: 100.00 %
    - Total run time: 1.851 s (limit: 10.000 s)
    - Bonus test - failed, evaluation: No bonus awarded
    - Failed (invalid output) **[Unlock advice (70 B)]**
    - Failed (invalid output) **[Unlock advice (70 B)]**
    - Failed (invalid output) **[Unlock advice (72 B)]**
    - Failed (invalid output) **[Unlock advice (70 B)]**
    - Failed (invalid output) **[Unlock advice (71 B)]**
    - Failed (invalid output) **[Unlock advice (69 B)]**
    - Failed (invalid output) **[Unlock advice (71 B)]**
    - Failed (invalid output) **[Unlock advice (70 B)]**
  - Test 'Random test (long intervals, bonus #2)': Abnormal program termination (Time limit exceeded)
    - Cumulative test time exceeded, killed after:: 8.157 s (limit: 8.149 s)
    - Bonus test - failed, evaluation: No bonus awarded
  - Overall ratio: 100.00 % (= 1.00 * 1.00 * 1.00 * 1.00)
- Total percent: 100.00 %
- Early submission bonus: 0.50
- Total points: 1.00 * (5.00 + 0.50) = 5.50

SW metrics:		Total	Average	Maximum	Function name
	Functions:	**10**	--	--	--
	Lines of code:	**227**	**22.70 ± 21.78**	**66**	is_holiday
	Cyclomatic complexity:	**66**	**6.60 ± 5.70**	**20**	get_working_days