Hashovací funkce

Termín odevzdání:	**31.08.2024 23:59:59**	5663266.857 sec
Hodnocení:	**6.6000**	
Max. hodnocení:	**6.0000** (bez bonusů)	
Odevzdaná řešení:	6 / 20 Volné pokusy + 10 Penalizované pokusy (-10 % penalizace za každé odevzdání)	
Nápovědy:	1 / 2 Volné nápovědy + 2 Penalizované nápovědy (-10 % penalizace za každou nápovědu)	

Vaším úkolem je realizovat funkci (či sadu funkcí, ne celý program), které naleznou **libovolnou** zprávu, jejíž hash (**SHA-512**) začíná zleva na posloupnost nulových bitů.

Pořadí bitů je big-endian: Bajt 0 od MSB do LSB, Bajt 1 od MSB do LSB, …, poslední bajt od MSB do LSB.

Neboli, dva nulové bity odpovídají například bajtu `0010 0111` (`0x27`).

Funkce je požadována ve dvou variantách:

- základní řešení (funkce `findHash`). Implementace této funkce je povinná.
- vylepšené řešení (funkce `findHashEx`). Implementace této funkce není povinná, bez dodané „dummy“ implementace se ale úloha nepodaří zkompilovat. Funkci implementujte, pokud se rozhodnete usilovat o bonus.

Parametry Vámi implementovaných funkcí jsou:

```
int findHash (int bits, string & message, string & hash)
```

- `bits` - požadovaný počet nulových bitů v hashi zprávy.
- `message` - výstupní parametr. Tento parametr obsahuje data, pro která byl nalezen příslušný hash. Výsledek je uložen jako **hexadecimální** řetězec.
- `hash` - výstupní parametr. Jedná se o hash zprávy `message` z předchozího parametru, opět jde o **hexadecimální** řetězec.
- Návratovou hodnotou funkce je `1` v případě úspěchu, `0` v případě neúspěchu nebo nesprávných parametrů. Těmi je typicky požadovaný počet nulových bitů, který nedává smysl.

```
int findHashEx (int bits, string & message, string & hash, string_view hashFunction)
```

- rozšíření funkce `findHash`. Všechny parametry i návratová hodnota zůstavají stejné jako v případě základní varianty.
- `hashFunction` - nový parametr, který udává, která hashovací funkce má být použita pro nalezení posloupnosti nulových bitů. Zadaný název hashovací funkce je kompatibilní s funkcí `EVP_get_digestbyname`.

Odevzdávejte zdrojový soubor, který obsahuje implementaci požadované funkce `findHash`, resp. `findHashEx`. Do zdrojového souboru si můžete přidat i další Vaše podpůrné funkce, které jsou z `findHash` (resp. `findHashEx`) volané. Funkce bude volána z testovacího prostředí, je proto důležité přesně dodržet zadané rozhraní funkce.

Za základ pro implementaci použijte kód z přiloženého archivu níže. Ukázka obsahuje testovací funkci `main`, uvedené hodnoty jsou použité při základním testu. Všimněte si, že vkládání hlavičkových souborů a funkce `main` jsou zabalené v bloku podmíněného překladu (`#ifdef/#endif`). Prosím, ponechte bloky podmíněného překladu i v odevzdávaném zdrojovém souboru. Podmíněný překlad Vám zjednoduší práci. Při kompilaci na Vašem počítači můžete program normálně spouštět a testovat. Při kompilaci na Progtestu funkce `main` a vkládání hlavičkových souborů „zmizí“, tedy nebude kolidovat s hlavičkovými soubory a funkcí `main` testovacího prostředí.

V ukázce se dále nachází funkce `dumpMatch`, kterou si budete (s nemalou pravděpodobností) muset implementovat pro své lokální testování. Funkce je zabalená v bloku podmíněného překladu (=nebude testována). Přesto je vhodné ji implementovat pro ověření správnosti Vašeho řešení.

Poznámky:

- **POZOR!** Odevzdaná úloha na Progtestu nemusí být zárukou splněné úlohy! Více informací se dozvíte od svého cvičícího.
- Při implementaci můžete využívat prostředky jazyka C i C++. Z knihovny STL je pro tuto úlohu dostupný jen `std::vector` a `std::string`.
- Nepřidávejte si další hlavičkové soubory, aktuální seznam je více než dostačující. Pokud se přesto rozhodnete přidat si další hlavičkové soubory, jejich vložení povede k chybě při překladu.
- Správné řešení není předpočítání si nějakého (= dostatečně dlouhého) hashe a poté jeho předložení jako výsledek. Takovéto řešení nebude uznáno jako validní. Správné řešení by mělo být randomizované s každým spuštěním programu.
- Při kompilaci nezapomeňte přilinkovat openssl crypto library pomocí `-lcrypto`.
- Verze OpenSSL na progtestu je 3.0.11.

Vzorová data: **Download**

Odevzdat:	Choose File No file chosen	**Odevzdat**

☐ **Referenční řešení**

6	**25.04.2024 17:39:22**	**Download**
Stav odevzdání:	Ohodnoceno	
Hodnocení:	6.6000	

- **Hodnotitel: automat**
  - Program zkompilován
  - Test 'Zakladni test podle ukazky': Úspěch
    - Dosaženo: 100.00 %, požadováno: 100.00 %
    - Celková doba běhu: 0.006 s (limit: 15.000 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test osetreni nespravnych vstupu': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 0.000 s (limit: 14.994 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test velkymi daty': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 3.443 s (limit: 14.994 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test nahodnymi daty + mem debugger': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 0.064 s (limit: 2.000 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Bonus: Test osetreni nespravnych vstupu (vice hashovacich funkci)': Úspěch
    - Dosaženo: 100.00 %, požadováno: 100.00 %
    - Celková doba běhu: 0.002 s (limit: 1.000 s)
    - Úspěch v nepovinném testu, hodnocení: 100.00 %
  - Test 'Bonus: Test vice hashovacimi funkcemi': Úspěch
    - Dosaženo: 100.00 %, požadováno: 100.00 %
    - Celková doba běhu: 0.003 s (limit: 1.000 s)
    - Úspěch v bonusovém testu, hodnocení: 110.00 %
  - Celkové hodnocení: 110.00 % (= 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.10)
- Použité nápovědy: 1
- Penalizace za vyčerpané nápovědy: Není (1 <= 2 limit)
- Celkové procentní hodnocení: 110.00 %
- Celkem bodů: 1.10 * 6.00 = 6.60

SW metriky:

	Celkem	Průměr	Maximum	Jméno funkce
Funkce:	**16**	--	--	--
Řádek kódu:	**125**	**7.81 ± 6.15**	**21**	`findHashEx`
Cyklomatická složitost:	**53**	**3.31 ± 2.75**	**10**	`findHashEx`

5	**25.04.2024 17:18:45**	**Download**
Stav odevzdání:	Ohodnoceno	
Hodnocení:	0.0000	

- **Hodnotitel: automat**
  - Program zkompilován
  - Test 'Zakladni test podle ukazky': Úspěch
    - Dosaženo: 100.00 %, požadováno: 100.00 %
    - Celková doba běhu: 0.006 s (limit: 15.000 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test osetreni nespravnych vstupu': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 0.000 s (limit: 14.994 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test velkymi daty': Program překročil přidělenou maximální dobu běhu
    - Vyčerpání limitu na celý test, program násilně ukončen po: 15.021 s (limit: 14.994 s)
    - Neúspěch v závazném testu, hodnocení: 0.00 %
  - Celkové hodnocení: 0.00 % (= 1.00 * 1.00 * 0.00)