

CTT003

THI GIỮA KỲ (60')

October 12, 2018

**Quy định thi giữa kỳ**

- KHÔNG SỬ DỤNG INTERNET (trừ trang web Moodle môn học - để download và nộp bài)
- Đây là bài thi cá nhân, tất cả các hình thức gian lận đều được chuyển về Khoa với hình thức xử lý “RỐT MÔN”. Các hình thức gian lận gồm:
  - Trao đổi, thảo luận trong lúc làm bài thi
  - Sao chép bài người khác không phải của mình (của bạn, trên Internet, ...)
  - Cho bạn sao chép bài của mình
  - Sử dụng INTERNET không theo quy định

**Quy định nộp bài**

- Nộp tại Moodle môn học, phần Midterm
- Nộp file MSSV.zip/functions.c – chỉ nộp file FUNCTIONS.C

## Bài 1 (3)

Có bao nhiêu số nguyên trong phạm vi từ trong khoảng  $\{m, n\}$  với  $10 \leq m \leq n \leq 99$ , sao cho tích của 2 chữ số bằng 2 lần tổng của 2 chữ số đó?

- Input:  $m = 35, n = 40$
- $10 \leq m \leq ab < n \leq 99: ab = 2(a+b)$
- Output: 1 số nguyên thỏa mãn điều kiện là 36
- Prototype: `int findab(int m, int n)`
- Statement: `findab(35, 40) = 1`

## Bài 2 (3)

Cho bài toán cổ như sau: trâu đứng ăn năm, trâu nằm ăn ba, ba trâu già ăn một. Cho trước tổng số lượng trâu và cỏ, số lượng trâu đứng, trâu nằm, trâu già (khác 0) thỏa điều kiện lần lượt là (`standing`, `lying`, `oldbulls`). Hỏi có bao nhiêu bộ (`standing`, `lying`, `oldbulls`) thỏa điều kiện khi biết số lượng trâu và cỏ lần lượt là `buffaloes` và `bunches_of_grass`?

- Input: `buffaloes = 100, bunches_of_grass = 120`
  - `standing*5 + lying*3 + oldbulls/3 = bunches_of_grass`
  - `standing + lying + oldbulls = buffaloes`
  - `standing, lying, oldbulls > 0`
- Vét cạn (bruteforce) tất cả các khả năng có thể để chọn được các khả năng thỏa điều kiện
- Output: 5 bộ (`standing`, `lying`, `oldbulls`) thỏa là (2, 29, 69) (6, 22, 72) (10, 15, 75) (14, 8, 78) (18, 1, 81)
- Prototype: `int bruteforce(int buffaloes, int bunches_of_grass)`
- Statement: `bruteforce(100, 120) = 5`

## Bài 3 (4)

Dãy số **Hailstone** bắt đầu từ một số tự nhiên **n** cho trước, các số tiếp theo sẽ được tạo theo quy tắc

- i) Nếu số hiện tại là chẵn thì số tiếp theo sẽ bằng số đó chia cho 2
- ii) Nếu số hiện tại là lẻ thì số tiếp theo sẽ được nhân lên 3 lần rồi cộng 1
- iii) Khi gặp số 1 thì dãy số kết thúc

Tính tổng các số trong dãy **Hailstone** khi biết **n**?

- Input: **n = 3**
- Khi đó, dãy Hailstone tương ứng là 3, 10, 5, 16, 8, 4, 2, 1.
- Output: Tổng = 49
- Prototype: `int hailstone(int n)`
- Statement: `hailstone(3) = 49`