



CTT31– NHẬP MÔN LẬP TRÌNH

BÀI TẬP VỀ NHÀ

BTVN-04: LẬP TRÌNH HÀM VÀ TỔ CHỨC CHƯƠNG TRÌNH

I. Thông tin chung

Mã số bài tập:	BTVN-04
Thời lượng dự kiến:	3 – 5 tiếng
Deadline nộp bài:	
Hình thức:	Bài tập cá nhân
Hình thức nộp bài:	Nộp qua Moodle môn học
GV phụ trách:	Võ Hoài Việt
Thông tin liên lạc với GV:	vhviet@fit.hcmus.edu.vn

II. Chuẩn đầu ra cần đạt

Bài tập này nhằm mục tiêu đạt được các chuẩn đầu ra sau:

- Lập trình hàm và tổ chức chương trình với các thành phần tách biệt trong các tập tin (.h và .cpp)

III. Mô tả bài tập

Bài 1: Viết hàm kiểm tra một số có phải là số chính phương hay không.

Bài 2: Viết hàm kiểm tra một số có phải là số nguyên tố hay không.

Bài 3: Viết hàm kiểm tra một số có phải là số đối xứng hay không.

Bài 4: Viết hàm kiểm tra xem 1 số nguyên dương n có phải là số toàn số lẻ hay không.

Bài 5: Viết hàm kiểm tra xem 1 số nguyên dương n có phải là số hoàn chỉnh hay không.

Bài 6: Viết hàm tính tổng các số nguyên tố từ 1 đến n .

Bài 7: Viết hàm tính tổng các số là số chính phương hoặc số hoàn chỉnh trong khoảng từ 1 đến n .

Bài 8: Viết hàm tính tích các số chính phương hoặc số chẵn trong khoảng từ 1 đến n .

Bài 9: Viết hàm tính tích các số nguyên tố hoặc số toàn lẻ trong khoảng từ 1 đến n .

Bài 10: Viết hàm tìm số chính phương đầu tiên trong khoảng từ 1 đến n . Nếu không có trả về -1.

Bài 10: Viết hàm tìm số hoàn chỉnh cuối cùng trong khoảng từ 1 đến n . Nếu không có trả về -1.

Bài 11: Tính tổng các số từ số hoàn chỉnh đầu tiên đến số hoàn chỉnh cuối cùng trong khoảng từ 1 đến n .

Bài 12: Tính tích các số chẵn từ số hoàn chỉnh đầu tiên đến số chính phương cuối cùng trong khoảng từ 1 đến n .

Bài 13: Tổng hợp tất các bài trên thành một chương trình thống nhất.

IV. Các yêu cầu & quy định chi tiết cho bài nộp





- Bài nộp được nén .RAR hoặc .ZIP và được nộp trên moodle. Với cấu trúc tên tập tin theo thứ tự mã số sinh viên SV1_SV2.RAR hoặc SV1_SV2.ZIP (Ví dụ: 0912496_0912407.RAR)

V. Hướng dẫn chi tiết

Khai báo hàm (thực hiện trong file .h)

Định nghĩa hàm (thực hiện trong file .cpp)

Sử dụng hàm (#include file .h vào chương trình chính hoặc nơi cần sử dụng hàm)

Ví dụ minh họa

Kiemtra.h

```
//Khai báo hàm  
bool KiemTraSoChan(int n);
```

Kiemtra.cpp

```
#include "StdAfx.h"  
#include "KiemTra.h"  
  
//Định nghĩa hàm  
bool KiemTraSoChan(int n)  
{  
    if( n % 2 == 0)  
    {  
        return true;  
    }  
    return false;  
}
```

TinhToan.h

```
#include "KiemTra.h"  
  
//Khai báo hàm  
int TinhTongSoChan(int n);
```

TinhToan.cpp

```
#include "StdAfx.h"  
#include "TinhToan.h"  
  
//Định nghĩa hàm  
int TinhTongSoChan(int n)  
{  
    int s = 0;  
    for(int i = 1; i < n; i++)  
    {  
        if ( KiemTraSoChan(i) == true)  
        {  
            s += i;  
        }  
    }  
    return s;  
}
```

Mai n. cpp

```
#include "stdafx.h"
#include "KiểmTra.h"
#include "TínhToán.h"
int _tmain(int argc, _TCHAR* argv[])
{
    int n;
    printf("--- CHUC NANG ---\n");
    printf("Nhap 1 - Kiểm tra số chan\n");
    printf("Nhap 2 - Tính tổng số chan\n");
    printf("--- ***** ---\n");
    printf("Nhap chuc nang: ");
    scanf("%d", &n);

    switch(n)
    {
        case 1:
            int so;
            printf("Nhap vao mot so: ");
            scanf("%d", &so);
            if( KiểmTraSoChan(so))
            {
                printf("%d là số chan", so);
            }else{
                printf("%d là số lẻ", so);
            }
            break;
        case 2:
            int son;
            printf("Nhap vao mot so: ");
            scanf("%d", &son);
            int s = TínhTongSoChan(son);
            printf("Tổng các số chan là: %d", s);
            break;
    }
    return 0;
}
```

VI. Cách đánh giá

STT	Tên kết quả	Tỉ lệ điểm	Ghi chú
1	Mã nguồn	80%	Cung cấp các thư viện và mã nguồn đầy đủ để biên dịch.
2	Phong cách lập trình	20%	Cấu trúc chương trình rõ ràng, hàm/ biến đặt tên dễ hiểu và gọi nhớ và tuân thủ các qui tắc lập trình.

VII. Tài liệu tham khảo

Slide bài giảng lý thuyết

VIII. Các quy định khác

- Chương trình phải có hướng dẫn sử dụng (Không có hướng dẫn sử dụng sẽ bị trừ 50% số điểm của phần phần chương trình).
- Chương trình nên có giao diện trực quan thể hiện các kết quả tính toán.
- Tất cả các bài làm sai quy định đều bị 0 điểm cho mỗi bài.
- Hai bài giống nhau từ 80% trở lên sẽ bị 0 điểm cho cả hai bất kể ai là tác giả.





- Các trường hợp sử dụng mã nguồn không ghi rõ nguồn tham khảo sẽ bị điểm 0 cho tất các các bài và các tác giả).
- Các bài làm xuất sắc sẽ được điểm cộng.
- Không nhận bài nộp trễ qua mail.

