



TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN
BỘ MÔN CÔNG NGHỆ PHẦN MỀM
HỆ CHÍNH QUI
MÔN: **KỸ THUẬT LẬP TRÌNH**

HƯỚNG DẪN THỰC HÀNH

Cấu trúc, con trỏ, cấp phát động

Tuần 4

- ✚ ĐẶNG HOÀNG HẢI
- ✚ TRẦN THANH HẢI
- ✚ ĐỖ NGUYỄN KHA
- ✚ ĐÀO ANH VŨ
- ✚ NGUYỄN LÊ HOÀNG DŨNG
- ✚ NGÔ CHÁNH ĐỨC

TP.HCM, ngày 17 tháng 03 năm 2018

MỤC LỤC

1	Quy định.....	3
2	Bài tập tại lớp	3

1 Quy định

Hoàn thành các bài tập dưới đây, tuân thủ các qui tắc lập trình đã học trong kỳ trước:

1. Qui tắc đặt tên
2. Qui tắc viết code
3. Qui tắc ghi chú (ghi chú mỗi project và ghi chú mỗi hàm)
4. Qui tắc chia file (.h và .cpp)
5. Qui tắc bố trí project (1 solution, n projects: Bai01, Bai02, ...)

2 Bài tập

Bài 1: Cho danh sách các món ăn và phần ăn như trong bảng 1 và bảng 2. Hãy suy nghĩ các tổ chức dữ liệu để vừa tiết kiệm không gian bộ nhớ vừa đảm bảo thực hiện dễ dàng việc chọn một phần ăn, hiển thị danh sách các món ăn trong phần ăn kèm giá tiền, tổng tiền của các món ăn và số tiền cần thanh toán (giảm 10% so với tổng tiền).

Món ăn	Giá tiền
Burger	130
Cheese burger	150
Chicken burger	200
Drink	130
Potato	120
Ice cream	160

Bảng 1: Danh sách món ăn

Phần ăn	Các món ăn
A	Burger, Drink, Potato
B	Cheese burger, Drink, Potato
C	Chicken burger, Drink, Potato
D	Chicken burger, Drink, Potato, Ice cream

Bảng 2: Danh sách phần ăn

Chọn một trong các phần ăn sau

- 1: Phần ăn A (Burger, Drink, Potato)
- 2: Phần ăn B (Cheese burger, Drink, Potato)
- 3: Phần ăn C (Chicken burger, Drink, Potato)
- 4: Phần ăn D (Chicken burger, Drink, Potato, Ice cream)

```
2
***** Phần ăn B *****
      Cheese burger:    150 JPY
              Drink:    130 JPY
              Potato:    120 JPY
-----
Tổng giá trị món :    400 JPY
Giá trị phần ăn :    360 JPY
```

Bảng 3: Giao diện mẫu

Bài 2: Đoạn mã nguồn trong bảng 10 sẽ khởi tạo một mảng động với số lượng phần tử do người dùng cung cấp. Sau đó, người dùng sẽ cung cấp cấp kích thước mới của mảng và chương trình sẽ thay đổi kích thước của mảng tương ứng. Tuy nhiên, phần mã nguồn được cung cấp thiếu một số phần. Các bạn sẽ bổ sung những phần thiếu, đặc biệt là những câu lệnh liên quan tới việc quản lý bộ nhớ. Hãy biên dịch và thực thi chương trình thành công.

```
#include <stdlib.h>
#include <stdio.h>

int main(){
    // Hàm khởi tạo ngẫu nhiên
    srand(time(NULL));

    // Đọc số lượng phần tử của mảng và lưu vào n1
    printf("Enter original array size:");
    int n1=0;
    scanf("%d",&n1);

    // Khởi tạo động mảng a1 có n1 phần tử kiểu int
    int* a1 = /* Bổ sung */
    int i;
    for(i=0; i<n1; i++){
        // Gán các phần tử trong mảng bằng 100
        /* Bổ sung */=100

        // Xuất các phần tử trong mảng ra màn hình
        printf("%d ",/* Bổ sung */ );
    }

    // Đọc số lượng phần tử (mới) của mảng, lưu vào n2
    printf("\nEnter new array size: ");
```

```
int n2=0;
scanf("%d",&n2);

// Cấp phát lại số lượng phần tử của a1
a1 = /* Bỏ sung */

// Nếu có nhiều phần tử hơn ban đầu, gán các phần
// tử mới bằng 0. Lý do, không muốn sử dụng
// các phần tử có giá trị rác.

/* Bỏ
sung
*/

for(i=0; i<n2;i++){
    // Xuất các phần tử của mảng ra màn hình
    printf("%d ",/* Bỏ sung */);
}
printf("\n");

// Kết thúc chương trình

return 0;
}
```

Bảng 4: Phần mã nguồn cần thay đổi

Bài 3: Thực hiện lại bài 1 nhưng sử dụng kiểu vector trong STL của C++

- Gợi ý: <http://www.cplusplus.com/reference/vector/vector/resize/>

Bài 4: Viết chương trình cho phép người dùng nhập n số nguyên và sử dụng mảng cấp phát động để lưu trữ. Xuất kết quả sử dụng hàm qsort được cung cấp sẵn trong thư viện stdlib.h để sắp xếp mảng x tăng hay giảm dần (tùy người dùng chọn).

Sort theo thuật toán nào cũng đc

- Gợi ý: <http://www.cplusplus.com/reference/cstdlib/qsort/>

Bài 5: Sử dụng bài tập kiểu dữ liệu sinh viên ở tuần 1 kết hợp sử dụng mảng cấp phát động để lưu danh sách sinh viên, qsort để sắp xếp lại danh sách theo thứ tự giảm/tăng dần sau đó xuất kết quả.

Kiểu Sinh viên có thông tin (họ tên (char*), mssv (int), điểm toán (float), điểm văn(float))
Sort theo thuật toán nào cũng đc cho THÔNG TIN HỌ TÊN