

BÀI TẬP 2

CHUYÊN ĐỀ TỔ CHỨC DỮ LIỆU

Họ và tên: Bùi phúc kiến

MSSV: 20880034

Câu 1. Cho đoạn chương trình

```
char a[11];  
*a = 'Y' - 1; //1  
*(a + 1) = 0x69; //2  
int *p = (int *) (a + 2); //3  
*p = 1751326830; //4  
a[6] = 133; //5  
p = (int *)&a[7]; //6  
*p = 0xA6F; //7
```

Giả sử kiến trúc máy tính là little-endian.

a) Cho biết kết quả in của đoạn chương trình trên.

b) Giải thích các lệnh chi tiết các lệnh từ //1 đến //7 ở trên để thấy kết quả in ra tại sao lại như vậy.

c) Mảng a chiếm bao nhiêu ô nhớ? Ghi giá trị (theo cơ số 16) của tất cả các ô nhớ của mảng a.

Bài làm:

a) Cho biết kết quả in của đoạn chương trình trên.

Đoạn chương trình trên in ra cụm từ “Xin chào”

b) Giải thích các lệnh chi tiết các lệnh từ //1 đến //7 ở trên để thấy kết quả in ra tại sao lại như vậy.

*a = 'Y' - 1; //1	'Y' - 1 = 'X', Gán ký tự 'X' vào a[0]
*(a + 1) = 0x69; //2	0x69 là mã ký tự 'i'. Gán ký tự 'i' vào a[1]
int *p = (int *) (a + 2); //3	(a + 2) là địa chỉ của phần tử a[2] chứa char. Int *p = (int *) (a + 2) ép (a + 2) thành kiểu địa chỉ chứa số nguyên gán vào p.
*p = 1751326830; //4	Đổi 1751326830 sang hệ thập lục phân: 1751326830 = 0x6863206E. *p = 1751326830 = 0x6863206E gán vào các ô nhớ mà p trỏ tới. Vì kiến trúc máy tính là little-endian nên: a[2] = 6E (chứa ký tự 'n') a[3] = 20 (khoảng trắng) a[4] = 63 (chứa ký tự 'c') a[5] = 68 (chứa ký tự 'h').
a[6] = 133; //5	133 là mã ký tự 'à' trong bảng mã ASCII mở rộng. Gán ký tự này vào a[6].
p = (int *)&a[7]; //6	Ép &a[7] thành kiểu địa chỉ chứa số nguyên gán vào p.

*p = 0xA6F; //7	Gán 0xA6F vào các ô nhớ mà p trỏ tới. a[7] = 6F (chứa kí tự ‘o’) a[8] = 0A (xuống dòng) a[9] = 0 (chứa kí tự ‘NUL’) a[10] = 0 (chứa kí tự ‘NUL’).
--------------------------------------	---

c) Mảng a chiếm bao nhiêu ô nhớ? Ghi giá trị (theo cơ số 16) của tất cả các ô nhớ của mảng a.

Mảng a chiếm 11 ô nhớ. Giá trị (theo cơ số 16) của tất cả các ô nhớ của mảng a là.

0x43	0x69	0x6E	0x20	0x63	0x68	0x85	0x6F	0x0A	0x00	0x00
------	------	------	------	------	------	------	------	------	------	------

Câu 2. Cho hàm f như sau:

```
int f(char *s){
    int n = 0;
    while(*s != '\0'){
        if('a' <= *s && *s <= 'f')
            n = n*16 + (*s - 'a' + 10);
        else
            n = n*16 + (*s - '0');
        s++;
    }
    return n;
}
```

a) Hãy cho biết kết quả trả về của f(“7de”). Công việc của hàm f là gì?

b) Hãy viết hàm void toHex(int d, char h[]) để chuyển số nguyên không âm d thành số thập lục phân để trong chuỗi h. Các kí số lớn hơn hoặc bằng 10 sẽ để ở dạng thường (a, b, c, d, e, f). Ví dụ đoạn chương trình sau đây sẽ in ra: ff

```
char h[100]
toHex(255, h)
printf(“%s”, h)
```

Bài làm:

a) Hãy cho biết kết quả trả về của f(“7de”). Công việc của hàm f là gì?

Kết quả in ra của f(“7de”) là 2014. Hàm f chuyển các chuỗi có dạng số thập lục phân thành số nguyên.

b) Hãy viết hàm void toHex(int d, char h[]) để chuyển số nguyên không âm d thành số thập lục phân để trong chuỗi h. Các kí số lớn hơn hoặc bằng 10 sẽ để ở dạng thường (a, b, c, d, e, f). Ví dụ đoạn chương trình sau đây sẽ in ra: ff

```
char h[100]
toHex(255, h)
printf("%s", h)
```

```
void toHex(int n, char h[]){
    int l = sizeof(h)/sizeof(h[0]); //Lấy chiều dài của chuỗi h
    int count = 0; //dùng một biến đếm để đếm số ký tự cần thiết.
    char temp[l]; //Khái báo một chuỗi tạm có kích thước bằng kích thước của chuỗi h.
    while(n != 0){
        int x = n%16; //Lấy số dư khi chia n cho 16
        if(x >= 10) //Nếu x >= 10 thì gán 1 trong các ký tự a, b,
            temp[count] = 'a' + x - 10; //c, d, e, f vào temp[count].
        else
            temp[count] = '0' + x; //Nếu x < 10 thì gán 1 trong các ký tự 0, 1,...,9 vào temp[count]
        n = n/16; //Lấy phần nguyên của n/16 gán lại vào n.
        count++;
    }
    for(int i = 0; i<count; i++){ //Gán các phần tử từ chuỗi tạm vào h theo thứ tự ngược lại.
        h[i] = temp[count-i-1];
    }
}
```

Câu 3. Viết chương trình cho người dùng nhập n số nguyên khác nhau và xuất ra số lớn thứ k trong n số đã nhập. Các số n, k do người dùng nhập với: $1 \leq n \leq 100, 1 \leq k \leq n$. Số lớn thứ 1 là số lớn nhất, số lớn thứ n là số nhỏ nhất.

Bài làm:

```
#include <stdio.h>

//Hàm swap dùng để đổi vị trí hai phần tử của mảng
void swap(int a[], int i, int j){
    int temp = a[i];
    a[i] = a[j];
    a[j] = temp;
}

//Sắp xếp các phần tử của mảng sử dụng selectionSort
void selectionSort(int a[], int l){
    for(int i = 0; i<l; i++){
        for(int j = i; j<l; j++){
            if(a[j] < a[i]){
                swap(a, i, j);
            }
        }
    }
}

int main(){
    int n, k;
    printf("Ban muon nhap bao nhieu so nguyen?\n");
    scanf("%d", &n);
    int a[n];
    for(int i = 0; i < n; i++){
        printf("Nhap so nguyen thu %d\n", i+1);
        scanf("%d", &a[i]);
    }
    printf("Cac so nguyen ban da nhap la: \n");
    for(int i = 0; i < n; i++){
        printf("%d\t", a[i]);
    }
    selectionSort(a, n);
    printf("\nBan muon tim so nguyen lon thu may?\n");
    scanf("%d", &k);
    while( k >= n || k <= 1) //Kiểm tra tính hợp lệ của k.
```

```

{
    printf("\nSo ban nhap khong hop le?\n");
    printf("\nBan muon tim so nguyen lon thu may?\n");
    scanf("%d", &k);
}
printf("So nguyen lon thu %d la %d?\n", k, a[k-1]);
return 0;
}

```

Bài 4. (2.4.6) Thời điểm trong ngày được xác định bởi giờ, phút, giây theo hệ 24 giờ, nghĩa là từ 0 giờ : 0 phút : 0 giây đến 23 giờ : 59 phút : 59 giây. Viết chương trình cho nhập 2 thời điểm, kiểm tra tính hợp lệ, cho biết thời điểm nào trước thời điểm nào (hay trùng nhau) và cho biết tổng số giây trôi qua giữa hai thời điểm đó.

(3.1.5) Hãy làm lại [Bài tập 2.4.6](#) để nhập/xuất thời điểm theo định dạng HH:MM:SS. (Ví dụ: thời điểm bắt đầu ngày là 00:00:00 và thời điểm cuối cùng trong ngày là 23:59:59)

Bài làm:

```

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

```

```

struct TIME
{
    int hh;
    int mm;
    int ss;
};

```

```

//Hàm tính tổng số giây ở thời điểm t
int totalSeconds(TIME t)
{
    return t.hh*3600 + t.mm*60 + t.ss;
}

```

```

//Hàm đổi từ số nguyên sang string
char* digitsToString(int x)
{
    char *p;
    p = (char*)malloc(3);
    int y = x%10;
    x = x/10;
    *p = '0' + x;
    *(p + 1) = '0' + y;
    *(p + 2) = '\0';
    return p;
}

```

```

//Hàm đổi từ thời gian sang string
char* timeToString(TIME t)
{
    char *p;
    p = (char*)malloc(9);
    char *hh = digitsToString(t.hh);
    char *mm = digitsToString(t.mm);
    char *ss = digitsToString(t.ss);
    *p = *hh;
    *(p + 1) = *(hh + 1);
    *(p + 2) = ':';
    *(p + 3) = *mm;
    *(p + 4) = *(mm + 1);
    *(p + 5) = ':';
    *(p + 6) = *ss;
    *(p + 7) = *(ss + 1);
    *(p + 8) = '\0';
    return p;
}

```

```
//Hàm so sánh hai thời điểm, cho biết thời điểm trước nào trước và tổng số giây trôi qua.
void compareTime(TIME t1, TIME t2)
{
    int s1 = totalSeconds(t1);
    int s2 = totalSeconds(t2);
    int diff = s1 - s2;
    if(diff < 0)
    {
        printf("Thoi diem %s truoc thoi diem %s.", timeToString(t1), timeToString(t2));
    }
    else if (diff == 0)
    {
        printf("Thoi diem %s trung thoi diem %s.", timeToString(t1), timeToString(t2));
    }
    else
    {
        printf("Thoi diem %s truoc thoi diem %s.", timeToString(t2), timeToString(t1));
    }
    printf("\nTong so giay troi qua giua hai thoi diem la %d.", abs(diff));
}
```

```
//Hàm kiểm tra thời gian hợp lệ 0 <= h <= 23, 0 <= m <= 59, 0 <= s <= 59.
bool valid(int h, int m, int s)
{
    if(h<=23 && h>=0 && m<=59 && m>=0 && s<=59 && s>=0)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}
```

```
//Hàm cho phép người dùng nhập thời gian.
TIME inputTime(){
    TIME t;
    char s[8];
    printf("Nhap thoi diem trong ngay theo dinh dang hh:mm:ss vi du 01:05:34 hoac 15:04:02 \n");
    scanf("%s", &s);
    int hh = (s[0] - '0')*10 + (s[1] - '0');
    int mm = (s[3] - '0')*10 + (s[4] - '0');
    int ss = (s[6] - '0')*10 + (s[7] - '0');
    while( !valid(hh, mm, ss))
    {
        printf("Thoi diem ban nhap chua hop le.\nCan nhap theo dinh dang hh:mm:ss vi du 01:05:34 hoac 15:04:02 \n");
        scanf("%s", &s);
        hh = (s[0] - '0')*10 + (s[1] - '0');
        mm = (s[3] - '0')*10 + (s[4] - '0');
        ss = (s[6] - '0')*10 + (s[7] - '0');
    }
    t.hh = hh;
    t.mm = mm;
    t.ss = ss;
    return t;
}
```

```
int main(){
    //Người dùng nhập hai thời điểm.
    TIME t1 = inputTime();
    TIME t2 = inputTime();

    //So sánh hai thời điểm.
    compareTime(t1, t2);

    return 0;
}
```

Chạy chương trình

```
C:\Users\ASUS\OneDrive\Google Drive\IT course\04 chuyen de to chuc du lieu\01\02\question_4.exe
Nhap thoi diem trong ngay theo dinh dang hh:mm:ss vi du 01:05:34 hoac 15:04:02
12:20:30
Nhap thoi diem trong ngay theo dinh dang hh:mm:ss vi du 01:05:34 hoac 15:04:02
25:11:34
Thoi diem ban nhap chua hop le.
Can nhap theo dinh dang hh:mm:ss vi du 01:05:34 hoac 15:04:02
12:29:09
Thoi diem 12:20:30 truoc thoi diem 12:29:09.
Tong so giay troi qua giua hai thoi diem la 519.
-----
Process exited after 27.78 seconds with return value 0
Press any key to continue . . .
```