

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH

ĐỀ THI CUỐI KỲ
HỌC KỲ 2 - NĂM HỌC 2021-2022

Môn thi: Nhập môn lập trình

Mã lớp: IT001.M21.CLC

Thời gian làm bài: 75 phút

Sinh viên không được sử dụng tài liệu

Sinh viên làm bài trực tiếp trên đề

gồm 10 câu in trên 06 trang giấy A4.

Chữ ký CBCT1:	STT: MASV: 22520274 Họ tên: NGUYỄN XUÂN VIỆT ĐỨC	Điểm số
Chữ ký CBCT2:		

Câu 1 (1 điểm): Điền vào các vị trí khuyết bên dưới, trình bày đặc điểm của biến tĩnh và biến cấp phát động.

- **Biến tĩnh** được khai báo tường minh và có tên gọi rõ ràng.
- **Biến cấp phát động** không được khai báo tường minh, không có tên gọi.
- **Biến cấp phát động** có thể giải phóng khi cần.
- Bản thân biến con trở là **biến tĩnh**
- Dùng biến **biến tĩnh** để lưu giữ địa chỉ của biến động.

Câu 2 (1 điểm): Cho đoạn chương trình sau:

```
#include <iostream>
using namespace std;

void input(int &n){
    cin >> n;
}

void myfunction(int n, int &s){
    if(n==0) return;
    s += n%10;
    myfunction(n/10, s);
}

int main() {
    int n, s=0;
    input(n);
    myfunction(n, s);
    cout << s;
    return 0;
}
```

Nhập n=6789. Chương trình xuất ra kết quả: **30**

Câu 3 (1 điểm): Viết chương trình nhập vào số nguyên dương n ($0 < n < 100$), tính tổng $S = 1^2 + 2^2 + 3^2 + \dots + n^2$ sử dụng hàm đệ quy và sau đó xuất kết quả S ra màn hình.

```
#include <iostream>
using namespace std;
void tongDeQuy(long long N, long long &S) {
    if(N<=0) return;
    else {
        S += (N*N);
        tongDeQuy(N-1, S);
    }
}
int main() {
    int N, S=0;
    cin >> N;
    tongDeQuy(N, S);
    cout << S;
}
```

Câu 4 (1 điểm): Hoàn thành đoạn chương trình sau để thực hiện hoán vị 2 giá trị x và y . Ví dụ nhập vào $x=7, y=8$, xuất ra $x=8, y=7$.

```
#include <iostream>
using namespace std;
void Nhap(int *&a) {
    a = new int;
    cin >> *a;
}
int* Nhap() {
    int *new_ptr = new int;
    cin >> *new_ptr;
    return new_ptr;
}
void Swap(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
int main() {
    int *x, *y;
    Nhap(x);
    y=Nhap();
    Swap(x, y);
    cout << "x=" << *x << ", y=" << *y;
}
```

Câu 5 (1 điểm): Hoàn thành hàm nhập và xuất mảng 1 chiều các số nguyên sử dụng con trỏ.

```
#include <iostream>

int* NhapMang(int &N) {
    std::cin >> N;
    int* A = new int[N];
    for(int i=0; i<N; i++)
        std::cin >> A[i];
    return A;
}

void XuatMang(int A[], int N) {
    for(int i=0; i<N; i++)
        std::cout << A[i] << " ";
    std::cout << std::endl;
}

int main() {
    int *a, n;
    a=NhapMang(n);
    XuatMang(a, n);
    return 0;
}
```

Câu 6 (1 điểm): Hoàn thành chương trình sau để duyệt và in ra màn hình đường chéo chính của mảng 2 chiều. Ví dụ cho sẵn mảng a như bên dưới, chương trình sẽ duyệt mảng a và in các giá trị đường chéo chính theo thứ tự “**8 6 2 0**”.

```
#include <iostream>
using namespace std;
int main() {
    int a[4][4] = {{ 0, 1, 2, 3 },
                  { 1, 2, 4, 5 },
                  { 4, 5, 6, 7 },
                  { 5, 6, 7, 8 } };

    int n=4;
    for(int i=n-1; i>=0; i--) {
        cout << a[i][i] << " ";
    }
}
```

Câu 7 (1 điểm): Viết hàm nhập mảng 2 chiều, và hàm in ra các giá trị viền của mảng 2 chiều. Ví dụ:

Nhập mảng a = { { 0, 1, 2, 3 },
 { 4, 5, 6, 7 },
 { 8, 9, 10, 11 },
 { 12, 13, 14, 15 } };

Hàm sẽ duyệt mảng a và in các giá trị viền của mảng theo thứ tự in đậm lần lượt như sau: **“0 1 2 3 7 11 15 14 13 12 8 4”**.

```
#include <iostream>
using namespace std;
#define MAX 100

void Nhap(int A[MAX][MAX], int &N) {
    cin >> N;
    for(int i=0; i<N; i++){
        for(int j=0; j<N; j++) {
            cin >> A[i][j];
        }
    }
}

void Print(int A[MAX][MAX], int N) {
    for(int i=0; i<N; i++) cout << A[0][i] << " ";
    for(int i=1; i<N-1; i++) cout << A[N-1][i] << " ";
    for(int i=N-1; i>=0; i--) cout << A[N-1][i] << " ";
    for(int i=N-2; i>=1; i--) cout << A[i][0] << " ";
}

int main() {
    int a[MAX][MAX];
    int n;
    Nhap(a, n);
    Print(a, n);
    return 0;
}
```

Câu 8 (1 điểm): Hoàn thành hàm bên dưới để thực hiện đếm số lượng ký tự in hoa trong một chuỗi.

```
int count_UpperCase(char *s1) {
    int i=0, dem=0;
    while(i<strlen(s1)) {
        if(s1[i]>='A' && s1[i]<='Z')
            dem++;
        i++;
    }
    std::cout << dem;
}
```

Câu 9 (1 điểm): Hoàn thành chương trình bên dưới để thực hiện nhập vào 1 vector với 2 điểm đầu và cuối, sau đó xuất ra tọa độ của vector vừa nhập. Ví dụ: Nhập vào vector \overrightarrow{AB} , với điểm A có tọa độ (3, 7), điểm B có tọa độ (5, 6). Vậy tọa độ của vector $\overrightarrow{AB} = (x_B - x_A, y_B - y_A) = (2, -1)$.

```
#include<iostream>
using namespace std;

struct Point { int x, y; };
struct Vecto {
    int dau;
    int cuoi;
};

void Nhap(Point &x) {
    int x, y;
    cin >> x >> y;
    x.x = x;
    x.y = y;
}

void Nhap(Vecto &x) {
    Point A, B;
    Nhap(A); Nhap(B);
    x.dau = B.x - A.x;
    x.cuoi = B.y - A.y;
}

void XuatToaDoVecto(Vecto x) {
    cout << "(" << x.dau << ", " << x.cuoi << ")";
}

int main() {
    Vecto a;
    Nhap(a);
    XuatToaDoVecto(a);
    return 0;
}
```

Câu 10 (1 điểm): Điền vào các vị trí trống để hoàn chỉnh đoạn chương trình nhập Đa thức thừa một biến bên dưới đây. Định nghĩa: Đa thức một biến $f(x)$ có dạng: $f(x)=a_nx^n+a_{n-1}x^{n-1}+...+a_0$. Với a_i ($i= 0..n$) là hệ số, x là biến, n là số mũ lớn nhất trong đa thức, vì là đa thức thừa nên số mũ của các đơn thức có thể không liên tiếp nhau. Ví dụ: Nhập vào đa thức: $6x^{12} + 7x^5 - 6x^3 - 2x + 1$.

```
#include<iostream>
using namespace std;
#define MAXN 1000

struct DONTTHUC {
    float HeSo;
    int SoMu;
};
struct DATHUC{
    DONTTHUC DonThuc[MAXN];
    int SoLuong;
};
void Nhap(DONTTHUC &a){
    cin >> a.HeSo >> a.SoMu;
}
DATHUC* Nhap(){
    DATHUC* new_dathuc = new DATHUC;
    int luongDonThuc;
    cin >> luongDonThuc;
    (*new_dathuc).SoLuong = luongDonThuc;
    for(int i=0; i<luongDonThuc; i++) {
        Nhap((*new_dathuc).DonThuc[i]);
    }
    return new_dathuc;
}
int main() {
    DATHUC *B;
    B = Nhap();
    return 0;
}
```

--- Hết ---