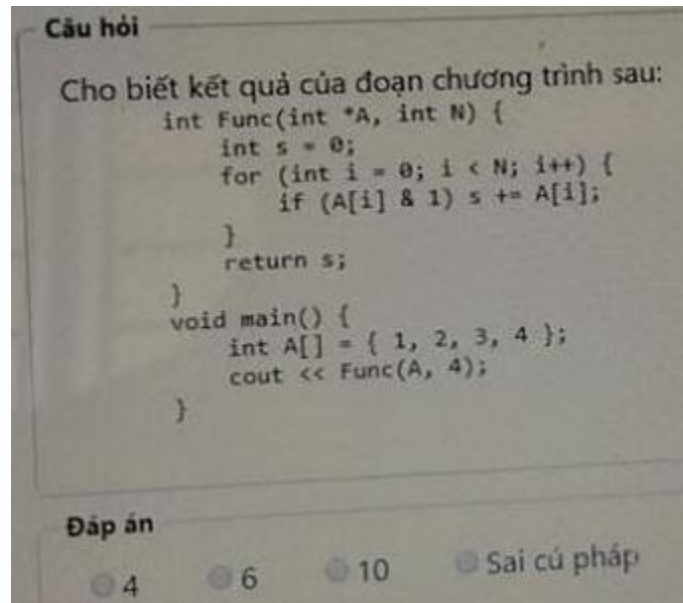


Đề thi

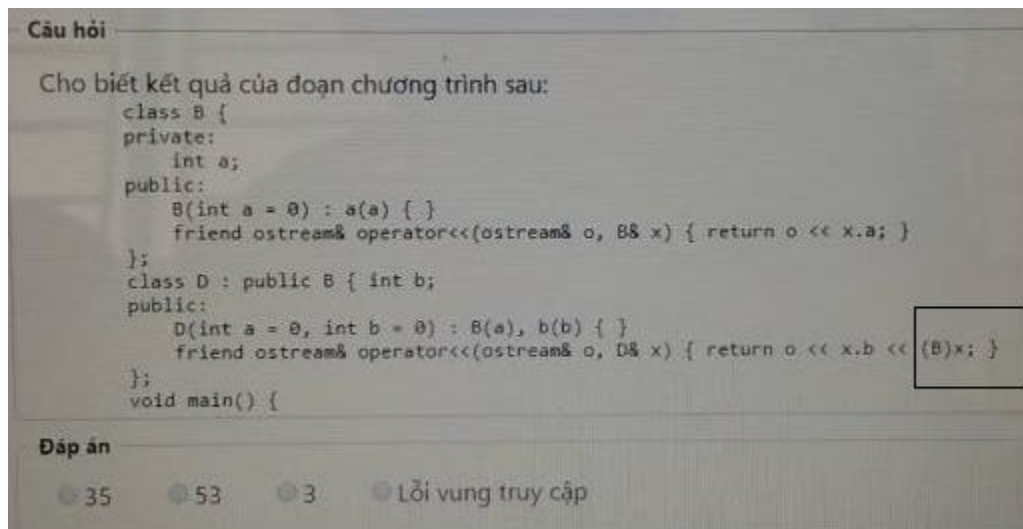


Tính tổng các số lẻ => =4

Prototype của hàm sau đây có lỗi định danh nào?

```
int test_error(int **for_test, int rows, int cols);
```

Không có lỗi nào cả, ** là con trỏ mảng, dùng để biểu diễn mảng 2 chiều (xem slide bài matrix của thầy Tùng)



Câu hỏi

Cho biết kết quả của đoạn chương trình sau:

```
class B {
private:
    int a;
public:
    B(int a = 0) : a(a) { }
    friend ostream& operator<<(ostream& o, B& x) { return o << x.a; }
};
class D : public B { int b;
public:
    D(int a = 0, int b = 0) : B(a), b(b) { }
    friend ostream& operator<<(ostream& o, D& x) { return o << x.a << x.b; }
};
void main() {
    D x(3, 5);
    cout << x;
}
```

Đáp án

- ☐ 35 ☐ 53 ☐ 3 ☐ Lỗi vùng truy cập

Biến a nằm trong vùng private nên D có kết thừa B thì trong class D cũng không thể gọi được biến a => lỗi vùng truy cập

Câu hỏi

Cho biết kết quả của đoạn chương trình viết cho Win32 dưới đây:

```
template <class T> T Abs(T a) { return a < 0 ? -a : a; }
void main() {
    cout << sizeof( Abs(-2) + Abs(1.0) );
}
```

Đáp án

- ☐ 8 ☐ 4 ☐ 12 ☐ 3

[< câu trước](#) | [câu sau >](#)

Sizeof trả về kích cỡ. Code trực tiếp trong DevC cho kết quả là 8. => cả -2 và 1.0 đều được xem như kiểu double

Câu hỏi

Cho biết tác dụng của hàm **Get(char *)**

```
int Get(char * s) {
    for (int i = 0; s[i]; i++)
        if (s[i] == '\n')
            return 1;
    return 0;
}
```

Đáp án

- ☐ Xác định vị trí của ký tự trong xâu ☐ Xóa ký tự trong xâu ☐ Kiểm tra sự tồn tại của ký tự trong xâu ☐ Đổi xâu thành số nguyên

Kiểm tra sự tồn tại : có trả về 1, không trả về 0

Câu hỏi

Những biểu thức nào sai cú pháp khi viết trong vùng các biểu thức của hàm main:

```
class String {
    char *data;
    int len;
public:
    static int StrLen(const char *src) {
        int i = 0; while (src[i]) i++;
        return i;
    }
};
void main() {
    String s;
    // Các biểu thức
}
```

Đáp án

☐ int sLength = s.StrLen("123"); ☐ int
dLength = String::StrLen(s.data); ☐ int
nLength = s.len; ☐ int cLength =
String::StrLen("abc");

Mặc định len và data trong class được khai báo là private nên trong hàm main không .len, .data được

Câu hỏi

Cho biết kết quả của đoạn chương trình sau:

```
class Shape {
protected:
    double a, b;
public:
    Shape(double a, double b) : a(a), b(b) { }
    virtual double Area() = 0;
};
class Rect : public Shape {
public:
    Rect(double a = 0, double b = 0) : Shape(a, b) { }
    double Area() { return a * b; }
};
class Triangle : public Shape { double c;
```

Đáp án

☐ 16 ☐ 22 ☐ 32 ☐ Lỗi cú pháp

Tính diện tích 2 hình rồi tổng lại thôi. Đ.a: 16

Cho biết kết quả của đoạn chương trình sau:

```
class A { int value;
public:
    A(int v = 0) { cout << (value = v); }
    ~A() { }
    A operator+(A x) { return A(value + x.value); }
};
void main() {
    A a = 2; a + 3;
}
```

Đáp án

☐ 352 ☐ 523 ☐ 235 ☐ Lỗi toán tử cộng

Lý thuyết là hàm nào gọi sau hủy trước. Theo thứ tự thì 2 3 5 được gọi nên sẽ in ra 532 (đã code trên DevC) nhưng chưa thử trên VS. Cảm tình với đáp án 523 nhất :3

Câu hỏi

Cho biết kết quả của đoạn chương trình sau:

```
int compare(int a, int b) { return a - b; }
void print(int *A, int N) {
    for (int i = 0; i < N; i++) cout << A[i] << ' ';
}
void InsertionSort(int *A, int N) {
    for (int i = 1; i < N; i++) {
        int e = A[i], j = i - 1;
        while (j >= 0 && compare(A[j], e) > 0)
            A[j + 1] = A[j--];
        A[j + 1] = e;
    }
    if (i == 2) print(A, N);
}
void main() {
    int A[] = { 33, 11, 44, 22, 55 };
    InsertionSort(A, 5);
}
```

Đáp án

☐ 11 33 22 44 55 ☐ 11 22 33 44 55
☐ 11 22 44 33 55 ☐ 11 33 44 22 55

Bắt đầu từ i=1 và dừng tại i=2 => mới chạy được 1 vòng = 11 chuyển về đầu => D

Cho biết kết quả của đoạn chương trình sau:

```
int i = 0, s = 100;
while (s > 10) {
    ++i; s >>= i;
}
cout << i;
```

Chú ý ++i. I=1 => s=50 ; I=2=> s=12 ; I=3 => s=1. Đ.a: 3 lần

Khi dịch hoặc chạy chương trình dưới đây sẽ sinh ra lỗi nào?

```
class Array { int *data, len;
public:
    Array(int length = 0) : data(0), len(length) {
        if (len > 0) data = new int[len];
    }
    ~Array() { if (data != 0) delete[] data; }
public:
    int& operator[](int index) { return data[index]; }
};
void main() {
    Array A(10), B = A;
    A[0] = 1;
}
```

Đáp án

- ☐ B = A - không có toán tử gán ☐ A[0] - không thể nằm bên trái toán tử gán ☐ A[0] - truy cập ngoài vùng mảng ☐ A.data bị xóa 2 lần

Sử dụng con trỏ và có hàm hủy thì phải viết riêng toán tử gán, nếu không vùng nhớ bị xóa 2 lần.

Câu 1: Random

```
void Random(int n) {
    for (int i = 0; i < n; i++) {
        int a = rand();
        cout << (i & 1 ? a | 1 : (a >> 1) << 1);
    }
}
```

- ☐ Trong khoảng [1, n]
- ☒ Chẵn lẻ xen kẽ
- ☐ Lẻ chẵn xen kẽ
- ☐ Trong khoảng (1, n)

Giải thích:

$i \& 1 ? a | 1 : (a \gg 1) \ll 1$

$i \& 1$ // kiểm tra $i \% 2 \Rightarrow$ nếu i lẻ trả về 1 và chẵn trả về 0

$a | 1$ // bit cuối của a khi or 1 ra 1 \Rightarrow kq là số lẻ

$(a \gg 1) \ll 1$ // phép $\gg 1$ trả về số nguyên, nên khi $\ll 1 \Rightarrow$ ra số chẵn

```
void Random(int n) {
    for (int i = 0; i < n; i++) {
        int a = rand();
        cout << a % n + 1;
    }
}
```

```
}  
}
```

Đáp án: Trong khoảng [1, n]

Câu 2: Tính giá trị hàm Func(15)

Biểu thức **Func(15)** cho giá trị nào nếu định nghĩa:

```
int Func(int n) {  
    int s = 0;  
    while (n) s += n--;  
    return s << 1;  
}
```

Đáp án

- ☐ 60
- ☐ 120
- ☐ 256
- ☒ 240

Đề có thể bỏ << 1; hoặc chuyển thành return s >> 1;

Giải: $s = 15 + 14 + \dots + 1 = 15 * 16 / 2 = 120$

Câu 2: Hàm Fun() trong String

Cho các định nghĩa sau:

```
typedef struct string_s {  
    char *values;  
    int length;  
} String;  
String * createString(int length = 0) {  
    String *p = (String *)malloc(sizeof(String));  
    p->values = (char *)malloc((p->length = length) + 1);  
    p->values[length] = 0;  
    return p;  
}  
int Func(String *src, String *dst) {  
    String * v = createString(src->length + dst->length);  
    int i = 0, j = 0;  
    while (i < src->length)  
        v->values[j++] = src->values[i++];  
    for (i = 0; i < dst->length; i++)  
        v->values[j++] = dst->values[i];  
    v->values[j] = 0;  
    return v;  
}
```

Hàm Func(String *, String *) có tác dụng nào?

Đáp án

- ☐ Nối src và dst
- ☐ So sánh src và dst
- ☐ Tìm vị trí của dst trong src
- ☐ **Nối dst vào src**

Vấn câu hỏi này đề có thể cho các code khác:

```
typedef struct string_s {
    char *values;
    int length;
} String;
String * createString(int length = 0) {
    String *p = (String *)malloc(sizeof(String));
    p->values = (char *)malloc((p->lenght = length) + 1);
    p->values[length] = 0;
    return s;
}
int Func(String *src, String *dst) {
    for (int i = 0; i < src->length; i++) {
        if (src->values[i] > dst->values[i]) return 1;
        if (src->values[i] < dst->values[i]) return -1;
    }
    return 0;
}
```

Đáp án: So sánh hai xâu.

Câu 3: Lỗi thuật toán A[i]

Chương trình dưới đây có lỗi thuật toán nào?

```
#include <iostream>
using namespace std;
void main() {
    const int n = 4;
    int A[n] = { 1, 15 }, i = 0;
    for (double s = 0; i < n; i++) {
        if (A[i] = 0) continue;
        s += 1.0 / A[i];
    }
}
```

Đáp án

- ☐ Chia cho 0
- ☐ Lặp vô cùng

○ Có A[i] chưa xác định

○ Tổng là số nguyên

Ứng với các đáp án trả lời khác:

```
#include <iostream>
using namespace std;
void main() {
    const int n = 4;
    int A[n] = { 1, 2, 3, 4 }, i = 0;
    for (double s = 0; i < n; i++) {
        if (A[i] = 0) continue;
        s += 1 / A[i]; // chia cho 0
    }
}
```

// Một đoạn khác nữa

Chương trình dưới đây có lỗi thuật toán nào?

```
#include <iostream>
using namespace std;
void main() {
    const int n = 4;
    int A[n] = { 1, 2, 3, 4 }, i = 0;
    for (double s = 0; i < n; i++) {
        if (A[i] == 0) continue;
        s += 1 / A[i]; // số nguyên
    }
}
```

```
#include <iostream>
using namespace std;
void main() {
    const int n = 4;
    int A[n] = { 1, 2, 3, 4 }, i = 0;
    for (double s = 0; n; i++) { // lặp vô cùng
        if (A[i] == 0) continue;
        s += 1 / A[i];
    }
}
```

Câu 4: Hàm GetChar

Cho biết tác dụng của hàm **Get(char *)**

```
int Get(char * s) {
    for (int i = 0; s[i]; i++)
        if (i == '\n') // có cái gì đó sai sai ở đây
            return i;
    return -1;
}
```

Đáp án

○ Xác định vị trí của ký tự trong chuỗi

- ☐ Kiểm tra sự tồn tại của ký tự trong xâu
- ☐ Đổi xâu thành số nguyên
- ☐ Xóa ký tự trong xâu

Một số code khác cho đáp án:

```
int Get(char * s) {
    for (int i = 0; s[i]; i++)
        if (i == '\n')
            return 1; // tồn tại
    return 0;
}
-----
```

```
int Get(char * s) {
    int a = 0, i = 0;
    while (1) {
        char c = s[i++];
        if (c > '9' || c < '0') break;
        a = a * 10 + c - 48; // trong code atoi và atof của thầy
    }
    return a;
}
Tùng
```

```
int Get(char * s) {
    int j = 0;
    for (int i = 0; s[i]; i++)
        if (i != '\n')
            s[j++] = s[i];
    s[j] = 0; // xóa
    return i != j;
}
```

Câu 5: Lỗi cú pháp

Chương trình dưới đây có lỗi cú pháp nào?

```
#include <iostream>
using namespace std;
void main() {
    int s = 0;
    for (i = 0; i < 10; i++) s += i;
}
```

Đáp án

- ☐ Thiếu toán tử return
- ☐ Sai cú pháp vòng for
- ☒ Thiếu khai báo biến
- ☐ Không có lỗi nào

```
#include <iostream>
using namespace std;
void main() {
    int n = 4;
    int A[n] = { 1, 2, 3, 4 };
    for (int i = 0, s = 0; i < n; i++)
        s += sqrt(A[i]);
}
```

1. Các khái niệm cơ bản

1. Cách đặt tên nào sau đây là sai cú pháp?

- ☒ Count-1 // không được có dấu -
 - ☐ _flag
 - ☒ 5th_Element // không được bắt đầu bằng số
 - ☒ struct // trùng tên từ khóa
-

2. Những hằng nào sau đây đúng quy cách?

- ☐ '0x49' // không có ký tự **0x49** trong mã ascii
- ☒ 0x49 // **0xAB** hệ 16 nên chấp nhận số 9
- ☐ 049 // **0AB** hệ 8 nên không được có số 9
- ☒ "049" // đặt giữa “ ” lúc nào cũng đúng

3. Những biểu thức khai báo nào sau đây là đúng?

- ☐ int n = 3, A[n]; // n phải là hằng số, viết const int n=3 thì đúng
 - ☐ short s[]; // phải khai báo số lượng phần tử
 - ☒ int *p = (int *)1000;
 - ☒ double d = 1e-10;
-

4. Cách đặt tên nào sau đây là đúng cú pháp?

- ☒ _flag
- ☐ 5th_Element
- ☐ struct
- ☐ Count-1

5. Hằng nào sau đây không đúng quy cách?

- ☐ 044
- ☒ '0x4F'
- ☐ "04F"

☐ 0x44

6. Những hằng nào sau đây đúng quy cách?

☐ '0x4F'

☐ 049

☒ 0x44

☒ "04F"

1A,2B,3a,4a,5a,6c,7a,8d,9a,10b,11b,12bc,13ac,14d 15ab

Tổng kết:

- Tên không đặt bắt đầu bởi số, chứa dấu – và trùng từ khóa
- Hằng có dạng hệ cơ số 8 là 0AB thì AB phải nhỏ hơn 8; dạng hệ 16 là 0xAB thì thỏa mãn; đặt trong “thì luôn đúng”; kiểu char ‘thì phải có trong mã ASCII’
- Khai báo mảng cần biết số lượng phần tử

2. Biểu thức và toán tử

1. Để nhân đôi một số nguyên dương a , biểu thức nào có tốc độ nhanh nhất?

- ☒ $a \ll 1$ // Các toán tử xử lý bit luôn thực hiện nhanh nhất
- ☐ $a *= 2$
- ☐ $a += a$
- ☐ Bằng nhau

2. Toán tử chia được thực hiện bao nhiêu lần trong đoạn biểu thức sau:

```
int s = 100;
while (s > 10)
    s /= 3;
```

- ☐ 5
- ☒ $3 // 100 - 33 - 11$
- ☐ 4
- ☐ Vô cùng

3. Cho biết kết quả trên màn hình khi chạy đoạn chương trình sau:

```
int a = 10, b = 5;
if (a == 0)
    cout << "a = 0";
```

```
else  
    cout << b / a;
```

- ☒ 0 // 5/10=0
- ☐ a = 0
- ☐ 0.5
- ☐ Báo lỗi chia cho 0

4. Cho biết giá trị của x sau khi thực hiện đoạn biểu thức dưới đây:

```
int x = 12, y = 78;  
while (y != 0) {  
    int z = x % y; x = y; y = z;  
}
```

- ☒ 6 // zxy lần lượt là 12 78 12 – 0 6 0
- ☐ 9
- ☐ 12
- ☐ 1

5. Toán tử dịch phải được thực hiện bao nhiêu lần trong đoạn biểu thức sau:

```
int s = 100;  
while (s > 10)  
    s >>= 1;
```

- ☒ 4 // 100 – 50 – 25 – 12
- ☐ 5
- ☐ 3
- ☐ Vô cùng

6. Nếu a = 0x76 thì biểu thức (a | 1) cho kết quả là bao nhiêu?

- ☐ 0
- ☐ 1
- ☒ 0x77 // toán tử or | _ 76 bit cuối là 0 | 1 lên 1 => tăng 1
- ☐ 0x74

7. Cho biết kết quả của biểu thức !!5 ?

- ☒ 1 // !5 = 0 nên !!5=1
- ☐ -5
- ☐ 5
- ☐ 0

8. Cho biết kết quả trên màn hình khi chạy đoạn chương trình sau:

```
int a = 10, b = 5;
if (a == 0)
    cout << "a = 0";
else
    cout << b / a;
```

- ☐ 0
- ☐ 0.5
- ☐ a = 0
- ☒ Báo lỗi chia cho 0

9. Nếu $a = 0x76$ thì biểu thức $(a \wedge 2)$ cho kết quả là bao nhiêu?

- ☒ 0x74 // giống bằng 0, khác bằng 1. $10 \wedge 10 = 00 \rightarrow$ giảm 2
- ☐ 0
- ☐ 1
- ☐ 0x77

10. Giá trị của biểu thức nào không phải là kiểu số nguyên nếu khai báo

int a; char c;

- ☐ c + a
- ☒ a + 1.0
- ☐ c + 1
- ☐ (char)(c + 32)

11. Nếu $a = 0x76$ thì biểu thức $((a \& 2) == 0)$ cho kết quả là bao nhiêu?

- ☐ 1
- ☒ 0 // $10 \& 10 = 10 \rightarrow$ sai
- ☐ 0x77
- ☐ 0x74

12. Cho biểu thức:

```
int x = 4, y = 2, z;
```

Những biểu thức nào cho z bằng 4?

- ☐ z = x++ - y--; // dùng trước cộng sau: z=2, x=3, y=1
- ☒ z = x > y ? x : y;
- ☒ z = (y % x) << 1;
- ☐ Cả 3 biểu thức

13. Cho đoạn khai báo sau

```
int a; char c;
```

Những biểu thức nào có kiểu int?

- ☒ `c + 1`
- ☐ `a + 1.0`
- ☒ `c + a` // c sẽ lấy giá trị mã ascii của nó đem cộng
- ☐ `(char)(c + 32)`

14. Toán tử nhân được thực hiện bao nhiêu lần trong đoạn biểu thức sau:

```
int s = 100;
while (s > 10) {
    s >>= 1; s *= 2;
}
```

- ☐ 5
- ☐ 3
- ☐ 4
- ☒ Vô cùng // chia 2 rồi lại nhân 2

15. Cho đoạn khai báo sau:

```
int x = 4, y = 2, z;
```

Những biểu thức nào cho z bằng 4?

- ☒ `z = (x > y) << 2;`
- ☒ `z = (x + y) > 5 ? x : y;`
- ☐ `z = x++ - y--;`
- ☐ Không biểu thức nào

1c2b3d4b5d6c7b8c9c10b

16. Cho biết kết quả trên màn hình sau khi thực hiện đoạn chương trình sau:

```
int i, j, k;
for (i = 0, k = 1; i < 2; i++)
    for (j = 0; j < 3; j++, k++)
        cout << k;
```

- ☐ 012345
- ☐ 0123456
- ☒ 123456 // tăng 6 lần bắt đầu từ 1 và in ra
- ☐ 6

17. Cho biết kết quả trên màn hình sau khi thực hiện đoạn chương trình sau:

```
int i, j, k;
for (i = 0, k = 0; i < 2; i++)
    for (j = 0; j < 3; j++, k++)
```

```
        cout << k;  
    cout << k;
```

- ☐ 123456
- ☒ 0123456 // tăng 6 lần từ 0 rồi in ra. Sau đó in tiếp k++ ra
- ☐ 6
- ☐ 012345

18. Cho biết kết quả trên màn hình sau khi thực hiện đoạn chương trình sau:

```
int i, j, k;  
for (i = 0, k = 0; i < 2; i++)  
    for (j = 0; j < 3; j++, k++)  
        cout << k;
```

- ☐ 0123456
- ☐ 123456
- ☐ 6
- ☒ 012345

19. Cho biết kết quả trên màn hình sau khi thực hiện đoạn chương trình sau:

```
int g = 1, n = 1;  
for (int i = 0; i < 4; i++, n = n * 2 + 1) g += n;  
cout << g;
```

- ☐ 5
- ☒ 27 // $1 + (1 + 3 + 7 + 15) -$ **bội độ thực hiện sau vòng lặp đầu tiên.**
- ☐ 12
- ☐ 58

20. Cho biết kết quả trên màn hình sau khi thực hiện đoạn chương trình sau:

```
int i, j, k;  
for (i = 0, k = 0; i < 2; i++)  
    for (j = 0; j < 3; j++, k++) ;  
cout << k;
```

- ☐ 0123456
- ☐ 123456
- ☐ 012345
- ☒ 6

21. Cho biết giá trị của T sau khi thực hiện đoạn biểu thức dưới đây:

```
int T = 17;  
switch (T * T % 7) {  
    case 1: T = 7; break;  
    case 2: T = 14; break;  
    case 4: T = 28; break;  
}
```


- ☐ 28
- ☐ 17
- ☒ 14
- ☐ 7

22. Cho biết giá trị của T sau khi thực hiện đoạn biểu thức dưới đây:

```
int T = 17;
switch (T * T % 5) {
case 1: T = 7; break;
case 2: T = 14; break;
case 4: T = 28; break;
}
```

- ☐ 14
- ☒ 28
- ☐ 7
- ☐ 17

23. Cho biết kết quả trên màn hình sau khi thực hiện đoạn chương trình sau:

```
int g = 1, n = 1;
for (int i = 0; i < 5; i++, n = n * 2 + 1) g += n;
cout << g;
```

- ☐ 12
- ☐ 5
- ☒ 58
- ☐ 27

24. Cho biết giá trị của T sau khi thực hiện đoạn biểu thức dưới đây:

```
int T = 17;
switch (T * T % 14) {
case 1: T = 7; break;
case 2: T = 14; break;
case 4: T = 28; break;
}
```

- ☐ 14
- ☐ 28
- ☒ 17
- ☐ 7

25. Cho biết kết quả trên màn hình sau khi thực hiện đoạn chương trình sau:

```
int g = 1;
for (int i = 0; i < 3; i++) g *= 2;
cout << g;
```

- ☐ 1
- ☒ 8
- ☐ 4
- ☐ 27

3. Hàm, mảng và con trỏ

1. Giá trị của biểu thức là bao nhiêu nếu định nghĩa:

```
int Calculate(int v) {
    if (v == 1) return v;
    return 1 + (v & 1 ? Calculate(v - 1) : Calculate(v / 2));
}
```

- ☐ 5
- ☒ 7
- ☐ 8
- ☐ 4

Câu này lặp lại rất nhiều. Tính thử cho 28:

28 . 14. 7. 6. 3. 2. 1 => 7 lần 1+ = 7

2. Lập trình hàm có những ưu điểm nào?

- ☐ Tăng tốc độ chương trình
- ☒ Dễ quản lý và chỉnh sửa
- ☒ Giảm dung lượng chương trình
- ☐ Tất cả các điểm trên

3. Cho mảng `int A[] = { 4, 5, 0, 8, 9, 2, 7, 3, 1 };` Tính giá trị của `A[A[2] + A[8]]` .

- ☐ 7
- ☐ 9
- ☒ 5 // `A[0+1]=A[1]=5`
- ☐ 0

4. Biểu thức nào sau đây là đúng?

- ☐ `char s[5]="12345";` // phải là 6 phần tử
- ☐ `int M[];` // kích cỡ mảng
- ☒ `double D[] = { 1, 2, 3 };`
- ☐ Cả 3 biểu thức

5. Cho biết kết quả của chương trình dưới đây:

```
template<class T> T Count(T v) { return v; }  
void main() {  
    cout << sizeof(Count(2.0) + Count(3) + Count('4'));  
}
```

- ☐ 2
- ☒ 8 // 4+2+2=8
- ☐ 4
- ☐ 1

6. Cho mảng int A[]={4,5,0,8,9,2,7,3,1}; nếu &A[2] = 1096 thì (A + 4) là giá trị nào?

- ☐ 1102
- ☒ 1104 // 1096+4*2
- ☐ 7
- ☐ 8

7. Những biểu thức gọi hàm nào sau đây là sai nếu khai báo các hàm:

```
double Power(double, int);  
void ToScreen(int);
```

- ☒ ToScreen(5, 3);
- ☐ Power(5, 3);
- ☒ Power(5.0, 3.0);
- ☐ ToScreen((int)5.25);

8. Biểu thức nào sau đây là đúng?

- ☒ int M[] = { 1, 2, 3 };
- ☐ double D[2] = { 1, 2, 3 };
- ☐ char s[];
- ☐ int n = 10, A[n]; // const int thì đúng

9. Cho biết kiểu của biểu thức nếu khai báo hàm sau:

```
template<class T> T Count(T) { return v; }
```

- ☐ char
- ☐ int
- ☐ T
- ☒ double // ?

10. Bao nhiêu hàm Count được sinh ra trong đoạn chương trình sau:

```
template<class T> T Count(T v) { return v; }
```

```
void main() {
    cout << Count(1.0) + Count(0x01) + Count(1);
}
```

- ☐ 2
- ☐ 3
- ☐ 1
- ☐ 4

11. Cho các hàm sau:

```
double Power(double, int);
void ToScreen(int);
```

Biểu thức gọi hàm nào sau đây là **đúng**?

- ☐ ToScreen(5, 3);
- ☐ **ToScreen((int)5.25);**
- ☐ Power(5.0, 3.0);
- ☐ Cả 3 biểu thức

12. Giá trị của biểu thức là bao nhiêu nếu định nghĩa:

```
int Calculate(int v) {
    if (v == 1) return v;
    return 1 + (v & 1 ? Calculate(v - 1): Calculate(v / 2));
}
```

- ☐ 8
- ☐ 7
- ☒ 5
- ☐ 4

13. Biểu thức Count(2.0) + Count(3) + Count('4') sinh ra những hàm nào nếu khai báo:

```
template<class T> T Count(T);
```

- ☐ char Count(char)
- ☐ double Count(double)
- ☐ int Count(int)
- ☒ Cả 3 hàm trên

14. Giá trị của biểu thức là bao nhiêu nếu định nghĩa:

```
int Calculate(int v) {
    if (v == 1) return v;
```

```
return 1 + (v & 1? Calculate(v - 1): Calculate(v / 2));
```

```
}
```

- ☐ 7
- ☐ 5
- ☒ 4
- ☐ 8

15. Biểu thức nào *sai cú pháp* nếu khai báo:

```
int Analyze(int *, int *);  
int A[], *p, N, &r = N;
```

- ☒ Analyze(*A, *p)
- ☐ Analyze(A, &r)
- ☐ cout << Analyze(A, &N)
- ☐ Analyze(A, p)

16. Những biểu thức nào *đúng cú pháp* nếu khai báo:

```
void Analyze(int *, int *);  
int A[], *p, N, &r = N;
```

- ☐ cout << Analyze(A, &N) // ?
- ☐ Analyze(*A, *p)
- ☒ Analyze(A, &r)
- ☒ Analyze(A, p)

17. Giá trị của biểu thức là bao nhiêu nếu định nghĩa:

```
int Calculate(int v) {  
    if (v == 1 || v == 2) return v;  
    return 1 + (v & 1? Calculate(v - 1): Calculate(v / 2));  
}
```

- ☒ 7
- ☐ 5
- ☐ 4
- ☐ 8

18. Hàm inline có ưu điểm nào?

- ☐ Tăng tốc độ chương trình
- ☒ Dễ quản lý và chỉnh sửa
- ☐ Giảm dung lượng chương trình
- ☐ Tất cả các điểm trên

19. Cho mảng short A[] = { 4, 5, 0, 3, 9, 2, 8, 7, 1 } ; nếu &A[2] = 1096 thì (A + 5) là giá trị nào?

- ☐ 7
- ☐ 8
- ☐ 1104
- ☒ 1102 // 1096 + (5-2)*2

1b2bc3c4c5b6b7ac8a9d10a11b12c13d14c15a16cd17a18b19d

1. Cho biết kết quả của đoạn chương trình sau:

```
void Convert(int &a, int b) { a = ++b; }  
void main() {  
    int x = 3, y = 1;  
    Convert(x, y);  
    cout << x << y;  
}
```

- ☐ 32
- ☒ 21 // truyền & thì kết quả mới thay đổi sau khi ra khỏi hàm
- ☐ 31
- ☐ 22

2. Cho biết kết quả của đoạn chương trình sau:

```
void Convert(int a, int b) { a = ++b; }  
void main() {  
    int x = 3, y = 1;  
    Convert(x, y);  
    cout << x << y;  
}
```

- ☐ 22
- ☐ 21
- ☐ 32
- ☒ 31

3. Cho biết chỉ số của phần tử cuối cùng của mảng s nếu khai báo

```
char s[] = "123456789";
```

- ☐ 11
- ☒ 9 // phần tử cuối có chỉ số là s[8] và giá trị là 9
- ☐ 10
- ☐ 8

4. Chọn biểu thức thay cho /**/ trong đoạn biểu thức sau để khi gọi , độ dài của mảng p tăng lên 1:

```
int *p, n; // n - độ dài của mảng p
void Function() {
    int *q = new int[++n];
    for (int i = 0; i < /**/; i++)
        q[i] = p[i];
    delete[] p;
    p = q;
}
```

☒ n - 1 // độ dài mảng q là n+1 (vì ++n nên n=n+1). Vòng for cần lặp n lần (do mảng p có độ dài bằng n) => cần lặp từ i đến n-1.

- ☐ n
- ☐ n + 1
- ☐ Biểu thức khác

5. Cho biết chỉ số của phần tử cuối cùng của mảng s nếu khai báo

```
char s[] = "0123456789";
```

- ☐ 9
- ☐ 11
- ☒ 10
- ☐ 8

6. Chọn kết quả hiện trên màn hình sau khi thực hiện đoạn chương trình sau:

```
char s[] = "NgON ngu LAP trinh C++";
for (int i = 0; s[i]; i++) {
    if (s[i] >= 'A' && s[i] <= 'Z') s[i] += 32;
}
cout << s;
```

- ☐ NGON NGU LAP TRINH C++
- ☒ ngon ngu lap trinh c++
- ☐ Ngon Ngu Lap Trinh C++
- ☐ Không có đáp án đúng

7. Chọn kết quả hiện trên màn hình sau khi thực hiện đoạn chương trình sau:

```
char s[] = "NgON ngu LAP trinh C++";
for (int i = 0; s[i]; i++) {
    if (s[i] >= 'a' && s[i] <= 'z') s[i] -= 32;
}
cout << s;
```

- ☐ Ngon Ngu Lap Trinh C++
- ☐ ngon ngu lap trinh c++
- ☒ NGON NGU LAP TRINH C++
- ☐ Không có đáp án đúng

8. Chọn biểu thức thay cho // trong đoạn biểu thức sau để khi gọi , độ dài của mảng *p* bằng *len*:**

```
int *p, n; // n - độ dài của mảng p
void Function(int len) {
    int L = /**/, *q = new int[n = len];
    for (int i = 0; i < L; i++) q[i] = p[i];
    delete[] p;
    p = q;
}
```

- ☐ n
- ☐ len
- ☒ n < len? n: len // **L tối đa là len**
- ☐ n > len? n: len

9. Trong biểu thức nào, toán tử & là toán tử tham chiếu?

- ☒ int Find(int *, int &);
- ☐ a == b && b == c
- ☐ flag & 2
- ☐ int *p = &a; // lấy địa chỉ

10. Chọn kết quả hiện trên màn hình sau khi thực hiện đoạn chương trình sau:

```
char s[] = "NgON ngu LAP trINH C++", last = 32;
for (int i = 0; s[i]; i++) {
    char ch = s[i];
    if (ch >= 'a' && ch <= 'z' && last == 32) s[i] = ch - 32;
    if (ch >= 'A' && ch <= 'Z' && last != 32) s[i] = ch + 32;
}
cout << s;
```

- ☐ Ngon Ngu Lap Trinh C++
- ☒ NGON NGU LAP TRINH C++
- ☐ ngon ngu lap trINH c++
- ☐ Không có đáp án đúng

11. Cho biết kết quả của đoạn chương trình sau:

```
void Convert(int &a, int &b) { a = b++; }
void main() {
    int x = 3, y = 1;
    Convert(x, y);
    cout << x << y;
}
```

- ☐ 32
- ☐ 31
- ☐ 11

☒ 12

12. Vì sao đoạn chương trình dưới đây sinh ra lỗi?

```
int *p = new int;  
p = new int[10];  
delete[] p;  
delete p;
```

- ☒ Xóa con trỏ khi chưa cấp phát
- ☐ Cấp phát động không dùng dùng cho biến đơn
- ☐ Sai cú pháp
- ☐ Cấp phát mảng khi chưa xóa

13. Chọn biểu thức thay cho // trong đoạn biểu thức sau để khi gọi , độ dài của mảng p giảm đi 1:**

```
int *p, n; // n - độ dài của mảng p  
void Function() {  
    int *q = new int[/**/];  
    for (int i = 0; i < n; i++) q[i] = p[i];  
    delete[] p;  
    p = q;  
}
```

- ☐ $n - 1$
- ☒ $--n$
- ☐ $n--$
- ☐ 1 trong 3 biểu thức

14. Cho biết kết quả của đoạn chương trình sau:

```
void Convert(int a, int &b) { a = ++b; }  
void main() {  
    int x = 3, y = 1;  
    Convert(x, y);  
    cout << x << y;  
}
```

- ☐ 22
- ☐ 21
- ☐ 31
- ☒ 32

15. Cho biết kết quả của đoạn chương trình sau:

```
void Convert(int &a, int b) { a = b++; }  
void main() {  
    int x = 3, y = 1;  
    Convert(x, y);  
    cout << x << y;  
}
```

- ☐ 31
- ☒ 11
- ☐ 32
- ☐ 12

16. Chọn kết quả hiện trên màn hình sau khi thực hiện đoạn chương trình sau

```
char s[] = "NgON ngu LAP trinh C++", last = 32, ch;
for (int i = 0; ch = s[i]; i++) {
    if (ch >= 'a' && ch <= 'z' && last == 32) s[i] -= 32;
    if (ch >= 'A' && ch <= 'Z' && last != 32) s[i] += 32;
    last = ch;
}
cout << s;
```

- ☐ NGON NGU LAP TRINH C++
- ☒ Ngon Ngu Lap Trinh C++
- ☐ ngon ngu lap trinh c++
- ☐ Không có đáp án đúng

17. Tìm kết quả của đoạn chương trình sau:

```
int A[10] = {1, 2, 3, 4};
int i = A[0]; // =1
int *p = &A[1]; // p trỏ vào a1
p++; // p trỏ vào a2
(*p)--; // giá trị a2 trừ đi 1 -> còn 2
*(p + 2) = 3; // giá trị a4 =3
int k = A[0] + A[2] + A[4]; // 1 + 2 + 3
int j = A[2];
cout << i << j << k;
```

- ☐ 135
- ☐ 124
- ☐ 137
- ☒ 126

18. Biểu thức nào khai báo mảng A chứa được n phần tử?

- ☐ int A[n];
- ☐ int A = new int[n];
- ☒ int* A = new int[n];
- ☐ Cả 3 biểu thức

19. Những biểu thức nào khai báo mảng A chứa được 10 phần tử?

- ☒ int A[10];

- ☐ `int A = new int[10];`
- ☒ `int* A = new int[10];`
- ☐ Cả 3 biểu thức

2|4|2|1|3|2|3|3|1|2|4|1|2|4|2|2|4|3|13

1b2d3b4a5c6b7c8c9a10b11d12a13b14d15b16b17d18c19ac

20. Cho biết kết quả của chương trình sau:

```
int Get(int a, int b) {
    if (a == 2) return (b % 4 == 0? 29: 28);
    switch (a) {
        case 4: case 6:
        case 9: case 11: return 30;
    }
    return 31;
}
void main() { cout << Get(2000, 2); }
```

- ☐ 30
- ☒ 29
- ☐ 31
- ☐ 28

21. Cho biết kết quả của chương trình sau:

```
int Get(double a, double b, double c, double x[]) {
    double d = b * b - 4*a*c;
    if (d < 0) return 0;
    a *= 2;
    if (d == 0) {
        x[0] = x[1] = -b / a;
        return 1;
    }
    d = sqrt(d);
    x[0] = (-b - d) / a; x[1] = (-b + d) / a;
    return 2;
}
void main()
{
    double a = 1, b = 3, c = 2;
    double x[2];
    cout << Get(a, b, c, x); cout << x[1];
}
```

- ☐ -2-1
- ☐ 1-1
- ☒ 2-1
- ☐ 2-2

4. Kiểu struct

1. Đây là vùng truy cập mặc định của class ?

- ☐ public
- ☐ protected
- ☒ **private**
- ☐ Không xác định

2. Đây là vùng truy cập mặc định của struct ?

- ☒ **public**
- ☐ protected
- ☐ private
- ☐ Không xác định

5. Các thuật toán

Phần này rất khó giải thích, các bạn vào <http://nnlt.vst.edu.vn/> và xem hoạt động của thuật toán. Hoặc xem ở đây <https://www.youtube.com/channel/UCXdzRQcok-Hg5hHwvBEVEAA>

Theo mình tổng hợp thì như thế này:

- **Hậu tố:** Cứ gặp số thì ném ra, gặp dấu thì phải xem xét (nếu là +- thì được ném ra hết các dấu trong stack rồi đưa chính nó vào; nếu là */ thì chỉ được ném */ ra thôi), gặp) thì ném tất cả ra cho đến khi gặp (dừng lại.
- **Insertsort:** Duyệt từ phần tử thứ 2 trở đi, đem nó so sánh với các phần tử trước nó, đến khi nào chèn được vào giữa thì chèn (như chơi tú)
- **Selection:** Tìm thằng nhỏ nhất rồi đưa nó lên đầu
- **Bubble:** Duyệt từ cuối, cứ nhỏ hơn thì đổi chỗ, đến khi không đổi được thì xét thằng đó
- **Quick:** đem 1 thằng lên làm mốc, duyệt từ 2 bên về, trái dừng ở số lớn hơn mốc, phải dừng ở số nhỏ hơn mốc. Đổi chỗ. Nếu duyệt qua nhau thì đưa mốc xuống :3
- **Cây tìm kiếm nhị phân:** cha > con trái và nhỏ hơn con phải

- Cây heap: Cha là to nhất, còn phải > con trái

1. Chọn biểu thức hậu tố của biểu thức số học $(2 * (5 + 4) / 3 - 1)$

- ☒ 2 5 4 + * 3 / 1 -
- ☐ 2 5 * 4 3 / + 1 -
- ☐ 2 5 * 4 3 1 - / +
- ☐ 2 5 * 4 + 3 1 - /

2. Cho dãy $K = \{ 44, 55, 33, 11, 22, 66 \}$. Thuật toán InsertionSort(K, 6) phải thực hiện bao nhiêu lần so sánh 2 phần tử?

- ☐ 10
- ☐ 9
- ☐ 12
- ☒ 11

3. Thuật toán dưới đây xác định tính chất nào của *value* trong dãy A?

```
int Search(int A[], int N, int value) {  
    for (int i = N - 1; i >= 0; i--)  
        if (A[i] == value)  
            return i;  
    return -1;  
}
```

- ☐ Số lần xuất hiện
- ☐ Có tồn tại
- ☒ Vị trí cuối cùng
- ☐ Vị trí đầu tiên

4. Thuật toán dưới đây xác định tính chất nào của *value* trong dãy A?

```
int Search(int A[], int N, int value) {  
    for (int i = 0; i < N; i++)  
        if (A[i] == value)  
            return 1;  
    return 0;  
}
```

- ☒ Có tồn tại
- ☐ Vị trí cuối cùng
- ☐ Vị trí đầu tiên
- ☐ Số lần xuất hiện

5. Thuật toán tìm kiếm chia đôi phải lặp bao nhiêu lần khi tìm giá trị 5 trong dãy $\{ 1, 2, 3, 4, 5, 6 \}$

- ☐ 3
- ☒ 2
- ☐ 4
- ☐ 1

6. Dãy nào sau đây có tính chất của cây nhị phân tìm kiếm?

- ☐ {55, 33, 88, 22, 44, 66, 11, 99}
- ☐ {55, 33, 88, 44, 77, 66, 99, 11}
- ☒ {55, 33, 88, 22, 44, 66, 99, 11}
- ☐ {55, 88, 33, 22, 44, 66, 99, 11}

7. Chọn biểu thức hậu tố của biểu thức số học $(2 * 5 + 4 / 3 - 1)$

- ☐ 2 5 * 4 + 3 1 - /
- ☐ 2 5 4 + * 3 / 1 -
- ☐ 2 5 * 4 3 1 - / +
- ☒ 2 5 * 4 3 / + 1 -

8. Thuật toán tìm kiếm chia đôi phải lặp bao nhiêu lần khi tìm giá trị 5 trong dãy { 1,2,3,4,5,6,7 }

- ☐ 4
- ☐ 1
- ☐ 2
- ☒ 3

9. Cho dãy $K = \{ 44, 22, 66, 33, 11, 55 \}$. Thuật toán SelectionSort(K,6) phải thực hiện bao nhiêu lần đổi chỗ hai phần tử?

- ☒ 4
- ☐ 3
- ☐ 2
- ☐ 5

10. Thuật toán dưới đây xác định tính chất nào của *value* trong dãy *A*?

```
int Search(int A[], int N, int value) {
    int s = 0;
    for (int i = 0; i < N; i++)
        if (A[i] == value) s++;
    return s;
}
```

- ☐ Vị trí đầu tiên
- ☐ Vị trí cuối cùng
- ☐ Có tồn tại
- ☒ Số lần xuất hiện

11. Thuật toán dưới đây xác định tính chất nào của *value* trong dãy *A*?

```
int Search(int A[], int N, int value) {
    for (int i = 0; i < N; i++)
        if (A[i] == value)
            return i;
    return -1;
}
```

- ☒ Vị trí đầu tiên
- ☐ Vị trí cuối cùng
- ☐ Số lần xuất hiện
- ☐ Có tồn tại

12. Chọn biểu thức hậu tố của biểu thức số học $(2 * 5 + 4 / (3 - 1))$

- ☐ 2 5 4 + * 3 / 1 -
- ☐ 2 5 * 4 + 3 1 - /
- ☐ 2 5 * 4 3 / + 1 -
- ☒ 2 5 * 4 3 1 - / +

13. Cho dãy $K[] = \{ 5, 7, 3, 9, 1, 4, 8, 2, 6 \}$. Dãy K sẽ có trạng thái nào sau khi thực hiện thuật toán phân đoạn $\text{Part}(K, 0, 8)$ với phần tử chốt là phần tử $K[0]$?

- ☐ { 1, 7, 3, 4, 5, 2, 8, 9, 6 }
- ☐ { 1, 4, 3, 2, 5, 9, 8, 7, 6 }
- ☐ { 4, 2, 3, 1, 5, 9, 8, 7, 6 }
- ☒ { 1, 2, 3, 4, 5, 9, 8, 7, 6 }

14. Cho dãy $K = \{ 44, 11, 33, 55, 22, 66 \}$. Những trạng thái nào của K có thể xuất hiện trong thuật toán $\text{BubbleSort}(K, 6)$ sau mỗi lần lặp i ?

- ☒ { 11, 44, 22, 33, 55, 66 }
- ☐ { 11, 22, 44, 55, 33, 66 }
- ☐ { 11, 44, 33, 55, 22, 66 }
- ☒ { 11, 22, 44, 33, 55, 66 }

15. Cho dãy $K = \{ 55, 22, 66, 33, 11, 44 \}$. Thuật toán $\text{SelectionSort}(K, 6)$ phải thực

hiện bao nhiêu lần đổi chỗ hai phần tử?

- ☐ 4
- ☒ 3
- ☐ 5
- ☐ 2

16. Những dãy nào sau đây có tính chất heap (đồng)

- ☐ { 88, 77, 44, 11, 22, 66, 55 }
- ☒ { 88, 44, 77, 11, 22, 66, 55 }
- ☐ { 88, 44, 77, 11, 55, 66, 22 }
- ☒ { 88, 77, 55, 44, 66, 22, 11 }

17. Cho dãy $K = \{ 44, 33, 55, 11, 22, 66 \}$. Thuật toán InsertionSort(K, 6) phải thực hiện bao nhiêu lần so sánh 2 phần tử?

- ☐ 9
- ☒ 10
- ☐ 11
- ☐ 12

18. Chọn biểu thức hậu tố của biểu thức số học $((2 * 5 + 4) / (3 - 1))$

- ☐ 2 5 4 + * 3 / 1 -
- ☒ 2 5 * 4 + 3 1 - /
- ☐ 2 5 * 4 3 / + 1 -
- ☐ 2 5 * 4 3 1 - / +

1a2d3c4a5b6c7d8d9a10d11a12d13d14ad15b16bd17b18b

1. Đoạn mã dưới đây được dùng trong thuật toán nào?

```
for (int i = N / 2; i >= 0; i--)  
    Build(A, N, i);
```

- ☐ InsertionSort
- ☐ BubbleSort
- ☒ HeapSort
- ☐ QuickSort

2. Đoạn mã dưới đây được dùng trong thuật toán nào?

```
while (j >= 0 && A[j] < e)  
    A[j + 1] = A[j--];
```



```
A[j + 1] = e;
```

- ☐ BubbleSort
- ☐ QuickSort
- ☒ InsertionSort
- ☐ HeapSort

3. Đoạn mã dưới đây được dùng trong thuật toán nào?

```
if (lb >= ub) return;  
int j = Get(A, lb, ub);  
Sort(A, lb, j - 1); Sort(A, j + 1, ub);
```

- ☒ QuickSort
- ☐ InsertionSort
- ☐ BubbleSort
- ☐ HeapSort

6. Class

Đoạn chương trình dưới đây cho kết quả nào?

```
class Value {  
    int _v;  
public:  
    Value(int v = 0) : _v(v) { }  
    ~Value() { cout << _v; }  
};  
void main() {  
    Value a(5), b; // đầu tiên nó gọi hàm Value(5) sau đó gọi hàm  
    Value(0). Gõ sau thì hủy trước nên sẽ in ra 05.  
}
```

Đáp án

- ☐ 55
- ☒ 05
- ☐ 50
- ☐ 00

Cho class sau:

```
class String {
char * data;
public:
    String() { data = new char[1]; }
    String(char c, int n) {
        data = new char[n+1];
        for (int i = 0; i < n; i++) data[i] = c;
    }

    static void Copy(char *dst, char *src) {
        int i = 0; while (src[i]) dst[i] = src[i++];
        dst[i] = 0; }
};

void main() {
    char s1[10], s2[]="Kiem tra \Ngon ngu lap trinh\";
    // Các biểu thức
}
```

Chương trình sẽ sinh ra lỗi khi chạy đến biểu thức nào?

Đáp án

- ☐ **String::Copy(s1, s2);** // độ dài của s1 không đủ chứa s2
- ☐ String b('0', 8); ☐ String a; ☐ Cả 3 biểu thức trên

7. Cấu trúc dữ liệu

Cho biết ý nghĩa của hàm `Func(List *, Node*)` với các định nghĩa sau:

```
typedef struct node_s {
    node_s * next;
} Node;
typedef struct list_s {
    node_s *first, *last;
} List;
void Func(List *list, Node *p) {
    list->last == 0 ? list->first = p : list->last->next = p;
    p->next = 0;
    list->last = p;
}
```

Đáp án

☐ Xóa toàn bộ danh sách ☐ Tạo danh sách liên kết 1 chiều ☒ **Thêm một Node vào cuối danh sách** ☐ Thêm một Node vào đầu danh sách

// Trong code phần tính biểu thức số học của thầy có ghép vào đầu, chèn vào giữa, xóa,... nên đọc thêm.

Stack có thể được sử dụng trong những thao tác xử lý nào?

Đáp án

☐ Gọi hàm ☐ Khử đệ quy ☐ Tính toán biểu thức số học ☒ **Tất cả các đáp án trên**

8. Kế thừa