

BÁO CÁO BUỔI THỰC HÀNH SỐ 3

Bộ Môn Kỹ Thuật Lập Trình



Sinh viên: MSSV-Họ và tên

Giáo viên hướng dẫn: ThS. Lê Thị Hoa

HTGD: Đoàn Sỹ Nguyên

Hình 2. 1: Code bài 2.	5
Hình 2. 2: Code bài 2.	6
Hình 2. 3: Code bài 2.	7
Hình 2. 4 Code bài 2.	7
Hình 2. 5. Test case bài 2	8
Hình 3. 1. Code bài 3.	9
Hình 3. 2. Code bài 3.	10
Hình 3. 3. Code bài 3.	11
Hình 3. 4. Test case bài 3.	12
Hình 4. 1. Code bài 4.	13
Hình 4. 2. Test case bài 4.	14
Hình 5. 1. Code bài 5.	15
Hình 5. 2. Test case 1.	16
Hình 5. 3. Test case 2.	Error! Bookmark not defined.
Hình 5. 4. Test Case 3.	17
<i>Hình 6. 1. Code bài 6.</i>	<i>18</i>
<i>Hình 6. 2. Test case bài 6.</i>	<i>19</i>
<i>Hình 7. 1. Code bài 7.</i>	<i>20</i>
<i>Hình 7. 2. Test case bài 7.</i>	<i>21</i>
<i>Hình 8. 1. Test case bài 8.</i>	<i>22</i>
<i>Hình 9. 1. Test case 1 và 2 bài 9.</i>	<i>24</i>
<i>Hình 9. 2. Test case 3 bài 9.</i>	<i>25</i>
<i>Hình 10. 1. Test case bài 10.</i>	<i>26</i>
<i>Hình 11. 1. Test case bài 11.</i>	<i>28</i>
<i>Hình 12. 1 Test case 1 bài 12.</i>	<i>29</i>
<i>Hình 12. 2. Test case 2 bài 12.</i>	<i>30</i>
<i>Hình 12. 3. Test case 3 bài 12.</i>	<i>31</i>
<i>Hình 12. 4. Test case 4 bài 12.</i>	<i>32</i>
<i>Hình 12. 5 Test case 5 bài 12.</i>	<i>33</i>
<i>Hình 12. 6 Test case 6 bài 12.</i>	<i>34</i>
<i>Hình 12. 7. Test case 7 bài 12.</i>	<i>35</i>
<i>Hình 12. 8. Test case 8 bài 12.</i>	<i>36</i>

Bài thực hành số 3 – Tuần 12

Bài tập 1: Tính dãy Lucas

Dãy Lucas được định nghĩa bởi $L_n = L_{n-1} + L_{n-2}$ và bắt đầu bởi $L_0=2, L_1=1$. Viết hàm tính số Lucas thứ n .

Answer: (penalty regime: 10, 20, ... %)

```
1 #include<stdio.h>
2 #include<iostream>
3 using namespace std;
4
5 int lucas(int n) {
6     if(n == 0) return 2;
7     if(n == 1) return 1;
8     return lucas(n-1) + lucas(n-2);
9 }
10 //Ho va ten: Nguyen Viet Anh
11 //MSSV: 20215307
```

Precheck

Check

Hình 1. 1. Code bài 1.

```
10 //Ho va ten: Nguyen Viet Anh
11 //MSSV: 20215307
```

Precheck Check

	Test	Expected	Got	
✓	cout << lucas(5);	11	11	✓
✓	cout << lucas(10);	123	123	✓
✓	cout << lucas(30);	1860498	1860498	✓

Passed all tests! ✓

Hình 1. 2. Test case bài 1.

Answer: (penalty regime: 10, 20, ... %)

```

1 #include <iostream>
2 using namespace std;
3
4 int n;
5 int X[100], Y[100]; // # Lưu vị trí các bước di chuyển cụ
6 int mark[100][100]; // # Đánh dấu vị trí các ô mà quân mã
7
8 // # Mang hx, hy mô tả 8 vị trí quân mã có thể di chuyển
9 const int hx[] = {1, 1, 2, 2, -1, -1, -2, -2};
10 const int hy[] = {2, -2, 1, -1, 2, -2, 1, -1};
11
12 // # In ra dãy các bước di chuyển tìm dc
13 void print_sol(){
14     for (int j = 1; j <= n * n; ++j)
15         printf("(%d %d)\n", X[j], Y[j]);
16     exit(0);
17 }
18 //Ho va ten: Nguyen Viet Anh
19 //MSSV: 20215307
20 bool ifpossible_07(int x, int y){
21     if(x < 1 || x > n) return false;
22

```

Precheck

Check

Hình 2. 1: Code bài 2.

Answer: (penalty regime: 10, 20, ... %)

```

20 bool ifpossible_07(int x, int y){
21     if(x < 1 || x > n) return false;
22     if(y < 1 || y > n) return false;
23     if(mark[x][y] == 1) return false;
24     return true;
25 }
26 //# Thuật toán quay lui
27 void TRY(int k){
28     for(int i = 0; i < 8; i++){
29         int xx = X[k-1] + hx[i];
30         int yy = Y[k-1] + hy[i];
31         if(ifpossible_07(xx,yy)){
32             mark[xx][yy] = 1;
33             X[k] = xx;
34             Y[k] = yy;
35             if(k == n*n){
36                 print_sol();
37             }
38             TRY(k+1);
39             mark[xx][yy] = 0;
40         }
41     }

```

Precheck

Check

Hình 2. 2: Code bài 2.

Answer: (penalty regime: 10, 20, ... %)

```
31  if(ifpossible_07(xx,yy)){
32      mark[xx][yy] = 1;
33      X[k] = xx;
34      Y[k] = yy;
35      if(k == n*n){
36          print_sol();
37      }
38      TRY(k+1);
39      mark[xx][yy] = 0;
40  }
41  }
42  }
43  //Ho va ten: Nguyen Viet Anh
44  //MSSV: 20215307
45  int main(){
46      cin >> n;
47      mark[1][1] = 1;
48      X[1] = Y[1] = 1;
49      TRY(2);
50      return 0;
51  }
```

Precheck

Check

Hình 2. 3: Code bài 2.

Hình 2. 4 Code bài 2.

```

43 //Ho va ten: Nguyen Viet Anh
44 //MSSV: 20215307
45 int main(){
46     cin >> n;
47     mark[1][1] = 1;
48     X[1] = Y[1] = 1;
49     TRY(2);
50     return 0;
51 }

```

Precheck Check

	Input	Expected	Got	
✓	5	(1 1) (2 3) (3 5) (5 4) (4 2) (2 1) (3 3) (1 4) (2 2) (4 1) (5 3) (4 5)	(1 1) (2 3) (3 5) (5 4) (4 2) (2 1) (3 3) (1 4) (2 2) (4 1) (5 3) (4 5)	✓

Hình 2. 5. Test case bài 2

Bài tập 3: Bài toán người du lịch

Một người xuất phát tại thành phố 1, muốn đi thăm tất cả các thành phố khác, mỗi thành phố đúng 1 lần và quay về 1. Chi phí để đi từ thành phố i sang thành phố j là $c_{i,j}$. Hãy tìm tổng chi phí nhỏ nhất có thể

Answer: (penalty regime: 10, 20, ... %)

```

1  #include <bits/stdc++.h>
2  using namespace std;
3  #define MAX 100
4
5  int n, c[MAX][MAX]; //s? thành ph? và ma tr?n chi phí
6  int cmin = INT_MAX; //chi phí đi l?i nh? nh?t gi?a hai
7  int best = INT_MAX; //t?ng chi phí nh? nh?t c?n tìm, b
8  int curr; //t?ng chi phí t?i th?i đi?m hi?n t?i
9  int mark[MAX]; //dánh d?u nh?ng thành ph? đã đi
10 int x[MAX]; //luu gi? các thành ph? đã đi
11
12 //Đọc dữ liệu vào
13 void input(){
14     cin >> n;
15     for (int i = 1; i <= n; ++i)
16         for (int j = 1; j <= n; ++j){
17             cin >> c[i][j];
18             if (c[i][j] > 0) cmin = min(cmin, c[i][j]);
19         }
20     for(int i=0; i<MAX; i++)
21         mark[i] = 0;
22

```

Hình 3. 1.Code bài 3.

Answer: (penalty regime: 10, 20, ... %)

```
17         cin >> c[i][j];
18         if (c[i][j] > 0) cmin = min(cmin, c[i][j]);
19     }
20     for(int i=0; i<MAX; i++)
21         mark[i] = 0;
22     curr = 0;
23 }
24
25 //Ho va ten: Nguyen Viet Anh
26 //MSSV: 20215307
27 bool check_07(int k, int i){
28     if(mark[i] == 1) return false;
29     return true;
30 }
31
32 void solution_07(){
33     best = min(best,curr);
34 }
35
36 //# Thuật toán quay lui
37 void TRY(int k){
38
```

Hình 3. 2. Code bài 3.

Answer: (penalty regime: 10, 20, ... %)

```
46  if(k == n) {
47      curr += c[x[n]][1];
48      solution_07();
49      curr -= c[x[n]][1];
50  }
51  else TRY(k+1);
52
53  mark[i] = 0;
54  curr -= c[x[k-1]][i];
55  }
56  }
57 }
58 //Ho va ten: Nguyen Viet Anh
59 //MSSV: 20215307
60 int main() {
61     input();
62     x[1] = 1;
63     TRY(2);
64     cout << best;
65     return 0;
66 }
```

Precheck

Check

Hình 3. 3. Code bài 3.

```

58 //Ho va ten: Nguyen Viet Anh
59 //MSSV: 20215307
60 int main() {
61     input();
62     x[1] = 1;
63     TRY(2);
64     cout << best;
65     return 0;
66 }

```

Time left 23:12:30

Precheck
Check

	Input	Expected	Got	
✓	4 0 2 1 3 4 0 1 2 2 1 0 3 3 4 2 0	7	7	✓
✓	6 0 2 1 3 7 3 4 0 1 2 8 5 2 1 0 3 6 9 3 4 2 0 2 3 1 7 3 9 0 4 2 1 4 5 6 0	11	11	✓
✓	8 0 2 1 3 7 3 2 7 4 0 9 2 8 5 13 2	14	14	✓

Hình 3. 4. Test case bài 3.

Bài tập 4: LIS

Cho dãy a có n phần tử. Một dãy con của a là dãy thu được bằng cách xóa đi một số phần tử của a và giữ nguyên thứ tự các phần tử còn lại (có thể không xóa phần tử nào). Hãy tìm dãy con tăng dài nhất của a

Answer: (penalty regime: 10, 20, ... %)

```
8 }
9
10 // # Quy hoạch động,
11 // # Hàm lis(i) trả về độ dài dãy con tăng dài nhất kết t
12 int lis(int i) {
13     if(i < 0)
14         return 0;
15     if(mem[i] != -1){
16         return mem[i];
17     }
18     mem[i] = 1;
19     for(int j = 0; j < i; j++){
20         if(a[j] < a[i])
21             mem[i] = max(mem[i], lis(j) + 1);
22     }
23     return mem[i];
24 }
25
26 // # Truy vết lời giải
27 void trace(int i){
28     for(int i = 0; i < i: i++){
```

Hình 4. 1. Code bài 4.

```

36 //Ho va ten Nguyen Viet Anh
37 //MSSV: 20215307
38 int main(){
39     init();
40     cin >> n;
41     for(int i = 0; i < n; i++) cin >> a[i];
42     int res = 1, pos = 0;
43     for(int i = 1; i < n; i++){
44         if (res < lis(i)){
45             res = lis(i);
46             pos = i;
47         }
48     }
49     cout << res << endl;
50     trace(pos);
51     return 0;
52 }

```

Precheck Check

	Input	Expected
✓	6 2 1 5 4 3 6	3 2 5 6
✓	10 2 1 5 4 3 6 3 -6 9 10	5 2 5 6 9 10
✓	20 2 -10 -8 5 4 3 6 3 -6 9 10 12 2 5 25 9 15 30 -100 45	10 -10 -8 5 6 9

Hình 4. 2. Test case bài 4.

Bài tập 5: Tính tổ hợp

Answer: (penalty regime: 10, 20, ... %)

```
1 #include <iostream>
2 using namespace std;
3 int binom(int n, int k) {
4     if (k > n) return 0;
5     if (k == 0) return 1;
6     return binom(n-1, k) + binom(n-1, k-1);
7 }
8 int bio_07[1000][1000] = {0};
9 int binom2(int n, int k){
10
11     //#khu de quy
12     if (k > n){
13         return bio_07[n][k];
14     }
15     if (k == 0){
16         return bio_07[n][k];
17     }
18     bio_07[n][k] = bio_07[n-1][k] + bio_07[n-1][k-1];
19     return bio_07[n][k];
20 }
21 //Ho va ten: Nguyen Viet Anh
22 //MSSV: 20215307
```

Hình 5. 1. Code bài 5.

Answer: (penalty regime: 10, 20, ... %)

```
20 }
21 //Ho va ten: Nguyen Viet Anh
22 //MSSV: 20215307
23 int main() {
24     int m;
25     cin >> m;
26     for (int n = 1; n <= m; ++n){
27         for (int k = 0; k <= n; ++k)
28             printf("%d ", binom(n, k));
29         printf("\n");
30     }
31     for(int i = 1; i <= m; i++){
32         bio_07[i][0] = 1;
33         bio_07[0][i] = 1;
34     }
35     for (int n = 1; n <= m; ++n){
36         for (int k = 0; k <= n; ++k)
37             printf("%d ", binom2(n, k));
38         printf("\n");
39     }
40     return 0;
41 }
```

Precheck

Check

Hình 5. 2.Code bài 5.


```

21 //Ho va ten: Nguyen Viet Anh
22 //MSSV: 20215307
23 int main() {
24     int m;
25     cin >> m;
26     for (int n = 1; n <= m; ++n){
27         for (int k = 0; k <= n; ++k)
28             printf("%d ", binom(n, k));

```

Precheck

Check

	Input	Expected
✓	4	1 1 1 2 1 1 3 3 1 1 4 6 4 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1
✓	10	1 1 1 2 1 1 3 3 1 1 4 6 4 1 1 5 10 10 5 1 1 6 15 20 15 6 1 1 7 21 35 35 21 7 1 1 8 28 56 70 56 28 8 1 1 9 36 84 126 126 84 36 9 1

Hình 5. 3. Test Case bài 5.

Bài tập 6: Tìm ước chung lớn nhất

Answer: (penalty regime: 10, 20, ... %)

```
4  int gcd(int a, int b){
5      if (b == 0) return a;
6      return gcd(b, a % b);
7  }
8
9  int gcd2(int a, int b){
10     // #Khu de quy
11     if(a == 0 || b == 0){
12         return a + b;
13     }
14     // Ho va ten: Nguyen Viet Anh
15     // MSSV: 20215307
16     while(a != b){
17         if(a > b){
18             a -= b;
19         }
20         else{
21             b -= a;
22         }
23     }
24     return a;
25 }
```

Hình 6. 1. Code bài 6.

```

14 //Ho va ten: Nguyen Viet Anh
15 //MSSV: 20215307
16 while(a != b){
17     if(a>b){
18         a -=b;
19     }
20     else{
21         b-=a;
22     }

```

Precheck Check

	Input	Expected	Got	
✓	50 35	5 5	5 5	✓
✓	217 413	7 7	7 7	✓

Passed all tests! ✓

Hình 6. 2. Test case bài 6.

Bài tập 7: Liệt kê xâu nhị phân

Sử dụng phương pháp khử đệ quy bằng stack, hãy liệt kê các xâu nhị phân độ dài n không có k bit 1 nào liên tiếp

Answer: (penalty regime: 10, 20, ... %)

```
28 //Ho va ten: Nguyen Viet Anh
29 //MSSV: 20215307
30 // #Khu de quy
31 if(top.j == 1){
32     L = top.old_L;
33 }
34 if(top.j > 1){
35     s.pop();
36     continue;
37 }
38 if(L+1 < k || top.j == 0){
39     x[top.i] = top.j;
40     top.old_L = L;
41     if(top.j){
42         L += 1;
43     }
44     else L = 0;
45     s.push(state(top.i+1,0));
46 }
47 ++top.j;
48 }
49 return 0;
```

Precheck

Check

Hình 7. 1. Code bài 7.

```

28 //Ho va ten: Nguyen Viet Anh
29 //MSSV: 20215307
30 // #Khu de quy
31 if(top.j == 1){
32     L = top.old L;

```

Precheck Check

	Input	Expected	Got	
✓	4 2	0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 1 0 1 1 0 0 0 1 0 0 1 1 0 1 0	0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 1 0 1 1 0 0 0 1 0 0 1 1 0 1 0	✓
✓	5 3	0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 1 0 0 1 0 0 0 0 1 0 1 0 0 1 1 0 0 1 0 0 0 0 1 0 0 1 0 1 0 1 0 0 1 0 1 1	0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 1 0 0 1 0 0 0 0 1 0 1 0 0 1 1 0 0 1 0 0 0 0 1 0 0 1 0 1 0 1 0 0 1 0 1 1	✓

Hình 7. 2. Test case bài 7.

Bài tập 8: Cân đĩa

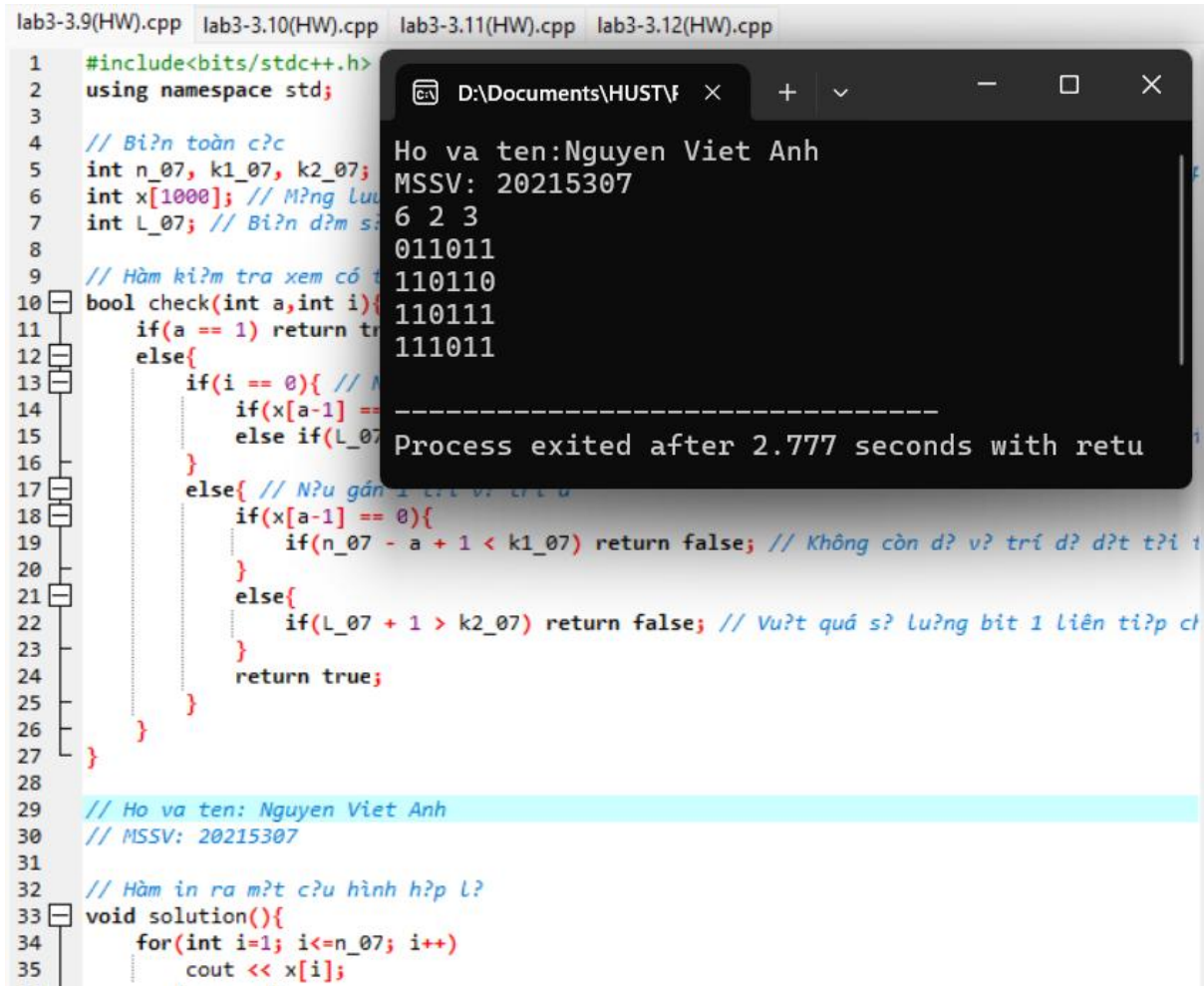
Bạn đang muốn kiểm tra xem một vật cho trước có đúng nặng M như người ta nói hay không. Có một cân thăng bằng và n quả cân. Quả thứ i nặng m_i . Hãy chỉ ra một cách cân thỏa mãn. Quy cách in ra đã được tích hợp trong mã nguồn dưới.

```
[*] lab3-3.8.cpp
9 int main() {
10     cout << "Nguyen Viet Anh\n";
11     cout << "MSSV:20215307\n";
12     int n, M;
13     cin >> n >> M;
14     int m[n+1];
15     for (int i = 1; i <= n; ++i) cin >> m[i];
16     int x[n+1];
17     stack<state> s;
18     /// sum of selected weights
19     int sum = 0;
20     s.push(state(1, -1));
21     while (!s.empty()){
22         state &top = s.top();
23         if (top.i > n){
24             if (sum == M){
25                 for (int i = 1; i <= n; ++i){
26                     if (x[i] == -1) cout << '-' << m[i];
27                     if (x[i] == 1) cout << '+' << m[i];
28                 }
29                 cout << "=" << M;
30                 exit(0);
31             }
32             s.pop();
33             continue;
34         }
35         //Ho va ten: Nguyen Viet Anh
36         //MSSV: 20215307
37         /// Khu de quy
38         s.pop();
39         x[top.i]=top.j;
40         s.push(state(top.i+1, -1, top.s-m[top.i+1]));
41         s.push(state(top.i+1, 0, top.s));
42         s.push(state(top.i+1, 1, top.s+m[top.i+1]));
43     }
44     cout << -1;
```

Hình 8. 1. Test case bài 8.

Bài tập 9: Lập lịch cho y tá

Một y tá cần lập lịch làm việc trong N ngày, mỗi ngày chỉ có thể là làm việc hay nghỉ ngơi. Một lịch làm việc là tốt nếu không có hai ngày nghỉ nào liên tiếp và mọi chuỗi ngày tối đại làm việc liên tiếp đều có số ngày thuộc đoạn $[K1, K2][1, 2]$. Hãy liệt kê tất cả các cách lập lịch tốt, với mỗi lịch in ra trên một dòng một xâu nhị phân độ dài n với bit 0/1 tương ứng là nghỉ/làm việc. Các xâu phải được in ra theo thứ tự từ điển



The image shows a C++ IDE with a file named lab3-3.12(HW).cpp open. The code defines a namespace std, includes <bits/stdc++.h>, and declares variables n_07, k1_07, k2_07, x[1000], and L_07. It includes a check function and a solution function. The terminal window shows the output of the program, which includes the student's name, ID, and a binary sequence.

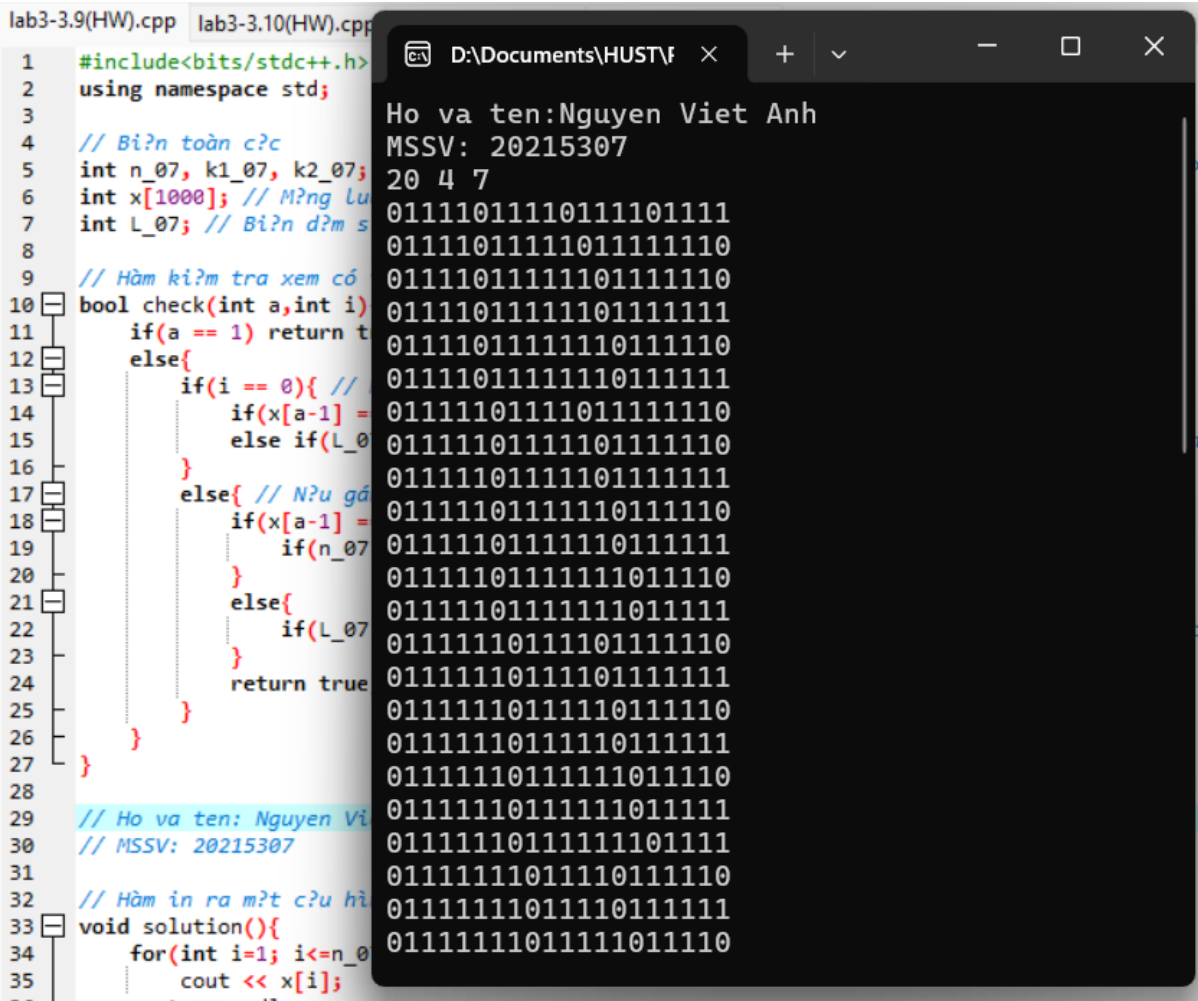
```
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 // Biện toàn c?c
5 int n_07, k1_07, k2_07;
6 int x[1000]; // M?ng lưu
7 int L_07; // Bi?n d?m s?
8
9 // Hàm ki?m tra xem có t
10 bool check(int a,int i){
11     if(a == 1) return tr
12     else{
13         if(i == 0){ // M
14             if(x[a-1] ==
15             else if(L_07
16         }
17     else{ // N?u g?n 1 t?i 0, t?i 1
18         if(x[a-1] == 0){
19             if(n_07 - a + 1 < k1_07) return false; // Không còn d? v? tr? d? d?t t?i t
20         }
21         else{
22             if(L_07 + 1 > k2_07) return false; // Vu?t quá s? l?ng bit 1 liên ti?p ch
23         }
24         return true;
25     }
26 }
27
28
29 // Ho va ten: Nguyen Viet Anh
30 // MSSV: 20215307
31
32 // Hàm in ra m?t c?u hình h?p l?
33 void solution(){
34     for(int i=1; i<=n_07; i++)
35         cout << x[i];
```

Terminal Output:

```
D:\Documents\HUST\VF x + - □ ×
Ho va ten:Nguyen Viet Anh
MSSV: 20215307
6 2 3
011011
110110
110111
111011
-----
Process exited after 2.777 seconds with retu
```

```
lab3-3.9(HW).cpp lab3-3.11(HW).cpp lab3-3.12(HW).cpp
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 // Bi?n toàn c?c
5 int n_07, k1_07, k2_07;
6 int x[1000]; // M?ng lu?i
7 int L_07; // Bi?n d?m s?
8
9 // Hàm ki?m tra xem có
10 bool check(int a,int i)
11 {
12     if(a == 1) return true;
13     else{
14         if(i == 0){ //
15             if(x[a-1] == 1)
16                 return true;
17             else if(L_07 == 0)
18                 return false;
19         }
20         else{ // N?u g?c
21             if(x[a-1] == 1)
22                 if(n_07 == 0)
23                     return true;
24             else{
25                 if(L_07 == 0)
26                     return false;
27             }
28         }
29     }
30 }
31
32 // Ho va ten: Nguyen Viet Anh
33 // MSSV: 20215307
34
35 // Hàm in ra m?t c?u hình h?p l?
36 void solution(){
37     for(int i=1; i<=n_07; i++)
38         cout << x[i];
39 }
```

Hình 9. 1. Test case 1 và 2 bài 9.



Hình 9. 2. Test case 3 bài 9.

Bài tập 10: Khoảng cách Hamming

Khoảng cách Hamming giữa hai xâu cùng độ dài là số vị trí mà ký tự tại vị trí đó là khác nhau trên hai xâu. Cho S là xâu gồm n ký tự 0. Hãy liệt kê tất cả các xâu nhị phân độ dài n , có khoảng cách Hamming với S bằng H . Các xâu phải được liệt kê theo thứ tự từ điển.

```

lab3-3.10(HW).cpp lab3-3.11(HW).cpp lab3-3.12(HW).cpp
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 int n_07;
5 const int MAX = 100;
6 int N_07[MAX], H_07[MAX];
7 int x[20];
8 int L_07;
9 // Kiểm tra xem có thể gán giá trị i tại
10 bool check(int a, int i, int N_k, int H_k)
11 {
12     if(i == 1){
13         if(L_07 + 1 > H_k) return false;
14         else return true;
15     }
16     else{
17         if(N_k - a + L_07 < H_k) return false;
18         else return true;
19     }
20 }
21 // In ra cấu hình hợp lệ
22 void solution(int N_k){
23     for(int i = 1; i <= N_k; i++){
24         cout << x[i];
25         cout << endl;
26     }
27 }
28 // Quay lui thu các cách chọn bit thỏa mãn
29 void TRY(int a, int N_k, int H_k){
30     for(int i = 0; i <= 1; i++){
31         if(check(a, i, N_k, H_k)){
32             x[a] = i;
33             int pre = L_07;
34             if(i == 1) L_07 += 1;
35             if(a == N_k) solution(N_k);
36             else{
37                 TRY(a + 1, N_k, H_k);
38             }
39         }
40     }
41 }

```

```

D:\Documents\HUSTV
Ho va ten: Nguyen Viet Anh
MSSV: 20215307
7
4 2
1 0
1 1
2 0
2 1
5 2
5 3
0011
0101
0110
1001
1010
1100
0
1
00
01
10

```

Hình 10. 1. Test case bài 10.

Bài tập 11: Lịch trình chụp ảnh

Superior là một hòn đảo tuyệt đẹp với n địa điểm chụp ảnh và các đường một chiều nối các điểm chụp ảnh với nhau. Đoàn khách tham quan có r người với sở thích chụp ảnh khác nhau. Theo đó, mỗi người sẽ đưa ra danh sách các địa điểm mà họ muốn chụp. Bạn cần giúp mỗi người trong đoàn lập lịch di chuyển sao cho đi qua các điểm họ yêu cầu đúng một lần, không đi qua điểm nào khác, bắt đầu tại điểm đầu tiên và kết thúc tại điểm cuối cùng trong danh sách mà họ đưa ra, và có tổng khoảng cách đi lại là nhỏ nhất.

```
lab3-3.11(HW).cpp lab3-3.12(HW).cpp
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 const int MAX = 10000;
5 int n, r;
6 int price[MAX][MAX];
7 int x[MAX];
8 bool visited[MAX];
9 vector<int> vt;
10 int best_07;
11 int sum_price;
12 int start, des, numOfPoint;
13
14 //Ho va ten: Nguyen Viet Anh
15 //MSSV: 20215307
16 void input() { //nhap ma tran
17     cin >> n >> r;
18     for (int i = 0; i < n; i++) {
19         for (int j = 0; j < n; j++) {
20             cin >> price[i][j];
21         }
22     }
23 }
24
25 bool check(int a, int i) {
26     if (visited[vt[i]]) return false;
27     if (price[x[a - 1]][vt[i]] == 0) return true;
28     return true;
29 }
30
31 void solution() {
32     if (price[x[numOfPoint - 2]][des] == 0) return;
33     best_07 = min(best_07, sum_price + price[x[numOfPoint - 2]][des]);
34 }
35
36 void TRY(int k) {
```

```
lab3-3.11(HW).cpp lab3-3.12(HW).cpp
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 const int MAX = 10000;
5 int n, r;
6 int price[MAX][MAX];
7 int x[MAX];
8 bool visited[MAX];
9 vector<int> vt;
10 int best_07;
11 int sum_price;
12 int start, des, numOfPoint;
13
14 //Ho va ten: Nguyen Viet Anh
15 //MSSV: 20215307
16 void input() { //nhap ma tran
17     cin >> n >> r;
18     for (int i = 0; i < n; i++) {
19         for (int j = 0; j < n; j++) {
20             cin >> price[i][j];
21         }
22     }
23 }
24
25 bool check(int a, int i) {
26     if (visited[vt[i]]) return false; //n
27     if (price[x[a - 1]][vt[i]] == 0) return false;
28     return true;
29 }
30
31 void solution() {
32     if (price[x[numOfPoint - 2]][des] == 0) return;
33     best_07 = min(best_07, sum_price + price[x[numOfPoint - 2]][des]);
34 }
35
36 void TRY(int k) {
```

Hình 11. 1. Test case bài 11.

Bài tập 12: Đếm đường đi

Cho đồ thị vô hướng G , hãy đếm số đường đi đi qua k cạnh và không đi qua đỉnh nào quá một lần.

lab3-3.12(HW).cpp

```
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 const int SO_DINH_TOI_DA = 100;
5 int so_dinh_07, so_dinh_can_chon_07;
6 int so_canh_07;
7 vector<vector<int>> danh_sach_ke_07;
8 int da_chon_07[SO_DINH_TOI_DA];
9 bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet Anh
13 //MSSV: 20215307
14 //Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai bu
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;
```

D:\Documents\HUST\I

Ho va ten: Nguyen Viet Anh
MSSV: 20215307
10 4
20
6 8
1 2
9 7
5 9
2 3
7 4
5 8
1 8
9 6
3 5
3 9
4 2
2 9
1 7
6 3
10 3
2 10
6 10
7 8
3 1
468

Process exited after 1.198 seconds with return code 0

Hình 12. 1 Test case 1 bài 12.

The image shows a C++ program in a text editor and its execution output in a terminal window.

Code (lab3-3.12(HW).cpp):

```

1  #include<bits/stdc++.h>
2  using namespace std;
3
4  const int SO_DINH_TOI_DA = 100;
5  int so_dinh_07, so_dinh_can_chon_07;
6  int so_canh_07;
7  vector<vector<int>> danh_sach_ke_07;
8  int da_chon_07[SO_DINH_TOI_DA];
9  bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet Anh
13 //MSSV: 20215307
14 // Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai bu
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;
37     for(int j = 0; j < danh_sach_ke_07[i].size(); j++){
38         int k = danh_sach_ke_07[i][j];
39         if(k == a) return false;
40     }
41     return true;
42 }
43
44 int main(){
45     nhap_du_lieu();
46     // Kiem tra xem co the chon dinh i tai bu
47     for(int i = 0; i < so_dinh_07; i++){
48         if(hop_le(0, i)){
49             dem_cach_chon_07++;
50         }
51     }
52     cout << dem_cach_chon_07 << endl;
53     return 0;
54 }

```

Execution Output:

```

14 13
4 15
20 14
20 17
8 13
14 2
14 8
9 6
20 5
4 8
3 7
1 20
4 11
1 8
14 10
16 10
12 6
11 2
8 6
14 9
15 6
16 14
16 20
16825
-----
Process exited after 1.711 seconds with return code 0

```

Hình 12. 2. Test case 2 bài 12.

```

1 #include<bits/stdc++.h>
2 using namespace std;
3
4 const int SO_DINH_TOI_DA = 100;
5 int so_dinh_07, so_dinh_can_chon_07;
6 int so_canh_07;
7 vector<vector<int>> danh_sach_ke_07;
8 int da_chon_07[SO_DINH_TOI_DA];
9 bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet Anh
13 //MSSV: 20215307
14 //Nhap thong tin do thi va khai tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai bu
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;
37     for(int j = 0; j < danh_sach_ke_07[i].size(); j++){
38         int k = danh_sach_ke_07[i][j];
39         if(k == 0) continue;
40         if(!hop_le(a + danh_sach_ke_07[i][j], k)) return false;
41     }
42     return true;
43 }
44
45 int main(){
46     nhap_du_lieu();
47     //Kiem tra xem co the chon dinh i tai bu
48     for(int i = 0; i < so_dinh_07; i++){
49         if(hop_le(0, i)) dem_cach_chon_07++;
50     }
51     cout << dem_cach_chon_07 << endl;
52     return 0;
53 }

```

```

1 3
2 1
14 15
1 13
4 15
8 12
14 18
19 16
19 6
16 11
1 8
4 5
1 12
4 2
13 11
14 11
7 10
2 14
1 18
10 17
13 4
3 4
11 6
9 19
16565
-----
Process exited after 1.503 seconds with retu

```

Hình 12. 3. Test case 3 bài 12.

The image shows a C++ IDE with two windows. The left window displays the source code for a program, and the right window shows the output of the program.

Source Code (lab3-3.12(HW).cpp):

```

1  #include<bits/stdc++.h>
2  using namespace std;
3
4  const int SO_DINH_TOI_DA = 100;
5  int so_dinh_07, so_dinh_can_chon_07;
6  int so_canh_07;
7  vector<vector<int>> danh_sach_ke_07;
8  int da_chon_07[SO_DINH_TOI_DA];
9  bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet An
13 //MSSV: 20215307
14 //Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai bu
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;
37     for(int j = 0; j < danh_sach_ke_07[a].size(); j++){

```

Output:

```

12 11
3 15
10 7
1 17
6 16
2 10
17 8
19 15
8 13
3 10
19 2
9 15
1 12
13 5
11 17
2 1
19 11
11 7
20 9
6 10
1 6
10 9
16 11
11 20
43781
-----
Process exited after 4.515 seconds with return value 0

```

Hình 12. 4.Test case 4 bài 12.

The image shows a C++ IDE with a code editor on the left and a terminal window on the right. The code editor displays a C++ program for a scheduling problem. The terminal window shows the output of the program, which consists of a list of tasks and their completion times, followed by a message indicating the process has exited.

```

1  #include<bits/stdc++.h>
2  using namespace std;
3
4  const int SO_DINH_TOI_DA = 100;
5  int so_dinh_07, so_dinh_can_chon_07;
6  int so_canh_07;
7  vector<vector<int>> danh_sach_ke_07;
8  int da_chon_07[SO_DINH_TOI_DA];
9  bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet An
13 //MSSV: 20215307
14 //Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai bu
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;

```

```

4  19
7  6
2  19
1  16
1  14
2  18
16  5
6  1
19  10
9  11
15  17
14  17
1  9
15  3
3  16
4  16
3  2
13  14
18  11
6  19
1  3
20  1
5  12
12  20
475911
-----
Process exited after 2.127 seconds with retu

```

Hình 12. 5 Test case 5 bài 12.

The image shows a C++ IDE with a source file named 'lab3-3.12(HW).cpp' and a terminal window displaying the execution output.

Source Code (lab3-3.12(HW).cpp):

```

1  #include<bits/stdc++.h>
2  using namespace std;
3
4  const int SO_DINH_TOI_DA = 100;
5  int so_dinh_07, so_dinh_can_chon_07;
6  int so_canh_07;
7  vector<vector<int>> danh_sach_ke_07;
8  int da_chon_07[SO_DINH_TOI_DA];
9  bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet An
13 //MSSV: 20215307
14 //Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai buoc a hay khong
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;
37     for(int j = 0; j < danh_sach_ke_07[i].size(); j++){
38         int k = danh_sach_ke_07[i][j];
39         if(hop_le(a + 1, k)) dem_cach_chon_07++;
40     }
41 }
42
43 int main(){
44     nhap_du_lieu();
45     hop_le(0, 0);
46     return 0;
47 }

```

Terminal Output:

```

21 27
17 18
1 15
7 26
24 22
9 7
3 11
18 4
7 3
18 2
22 7
10 1
6 11
5 10
21 8
10 4
3 5
15 28
26 21
14 27
14 22
8 11
19 26
12 17
802
-----
Process exited after 4.293 seconds with return code 0

```

Hình 12. 6 Test case 6 bài 12.

```

1  #include<bits/stdc++.h>
2  using namespace std;
3
4  const int SO_DINH_TOI_DA = 100;
5  int so_dinh_07, so_dinh_can_chon_07;
6  int so_canh_07;
7  vector<vector<int>> danh_sach_ke_07;
8  int da_chon_07[SO_DINH_TOI_DA];
9  bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet An
13 //MSSV: 20215307
14 //Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_b
23         danh_sach_ke_07[dinh2 - 1].push_b
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai bu
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;

```

```

D:\Documents\HUST\F
1 21
23 6
14 21
5 10
6 26
12 1
21 5
20 14
10 13
20 30
14 30
12 10
26 22
13 6
9 18
6 28
9 4
21 16
21 19
14 26
6 11
5 25
27 2
13 20
1105855
-----
Process exited after 3.159 seconds with retu

```

Hình 12. 7. Test case 7 bài 12

The image shows a C++ IDE with a source code window on the left and a console window on the right. The source code is for a program named 'lab3-3.12(HW).cpp'. It includes `<bits/stdc++.h>` and uses the `std` namespace. Constants and variables are defined, including `SO_DINH_TOI_DA = 100`, `so_dinh_07`, `so_dinh_can_chon_07`, `so_canh_07`, `danh_sach_ke_07`, `da_chon_07`, `da_tham_07`, and `dem_cach_chon_07`. A function `nhap_du_lieu()` is defined, which takes input for `so_dinh_07`, `so_dinh_can_chon_07`, and `so_canh_07`, and initializes `danh_sach_ke_07`, `da_tham_07`, and `dem_cach_chon_07`. A function `hop_le(int a, int i)` is also defined. The console window shows the output of the program, which is a list of numbers: 5 18, 29 16, 9 6, 18 9, 26 14, 18 22, 3 16, 12 28, 6 11, 18 24, 29 28, 13 28, 29 21, 5 19, 20 13, 22 2, 4 28, 29 2, 29 30, 13 24, 1 4, 23 5, 15 1, 6 12, 1865192. The console also shows the message 'Process exited after 5.601 seconds with retu'.

```

1  #include<bits/stdc++.h>
2  using namespace std;
3
4  const int SO_DINH_TOI_DA = 100;
5  int so_dinh_07, so_dinh_can_chon_07;
6  int so_canh_07;
7  vector<vector<int>> danh_sach_ke_07;
8  int da_chon_07[SO_DINH_TOI_DA];
9  bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet An
13 //MSSV: 20215307
14 //Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai bu
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;

```

```

5 18
29 16
9 6
18 9
26 14
18 22
3 16
12 28
6 11
18 24
29 28
13 28
29 21
5 19
20 13
22 2
4 28
29 2
29 30
13 24
1 4
23 5
15 1
6 12
1865192
-----
Process exited after 5.601 seconds with retu

```

Hình 12. 8. Test case 8 bài 12.

```
[*] lab3-3.12(HW).cpp
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 const int SO_DINH_TOI_DA = 100;
5 int so_dinh_07, so_dinh_can_chon_07;
6 int so_canh_07;
7 vector<vector<int>> danh_sach_ke_07;
8 int da_chon_07[SO_DINH_TOI_DA];
9 bool da_tham_07[SO_DINH_TOI_DA];
10 int dem_cach_chon_07;
11
12 //Ho va ten: Nguyen Viet An
13 //MSSV: 20215307
14 // Nhap thong tin do thi va khoi tao bien
15 void nhap_du_lieu(){
16     cin >> so_dinh_07 >> so_dinh_can_chon_07;
17     cin >> so_canh_07;
18     danh_sach_ke_07.resize(so_dinh_07);
19     for(int i = 0; i < so_canh_07; i++){
20         int dinh1, dinh2;
21         cin >> dinh1 >> dinh2;
22         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
23         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
24     }
25     for(int i = 0; i < so_dinh_07; i++){
26         da_tham_07[i] = false;
27     }
28     dem_cach_chon_07 = 0;
29 }
30
31 // Kiem tra xem co the chon dinh i tai buoc a hay khong
32 bool hop_le(int a, int i){
33     if(a == 0) return true;
34     if(da_tham_07[i]) return false;
35
36     int index = 0;
37     for(int j = 0; j < danh_sach_ke_07[i].size(); j++){
38         if(danh_sach_ke_07[i][j] == a) return false;
39     }
40     return true;
41 }
42
43 int main(){
44     nhap_du_lieu();
45     if(so_dinh_07 < 1 || so_dinh_07 > 100 || so_canh_07 < 1 || so_canh_07 > 100) return 0;
46     if(so_dinh_07 < so_canh_07) return 0;
47     danh_sach_ke_07.resize(so_dinh_07);
48     for(int i = 0; i < so_canh_07; i++){
49         int dinh1, dinh2;
50         cin >> dinh1 >> dinh2;
51         danh_sach_ke_07[dinh1 - 1].push_back(dinh2);
52         danh_sach_ke_07[dinh2 - 1].push_back(dinh1);
53     }
54     for(int i = 0; i < so_dinh_07; i++){
55         da_tham_07[i] = false;
56     }
57     dem_cach_chon_07 = 0;
58     do{
59         int a = 0;
60         do{
61             a++;
62             if(a > 100) break;
63             if(hop_le(a, 0)) continue;
64             da_tham_07[0] = true;
65             dem_cach_chon_07++;
66             danh_sach_ke_07[0].clear();
67             danh_sach_ke_07[0].resize(so_dinh_07);
68             for(int i = 1; i < so_dinh_07; i++){
69                 danh_sach_ke_07[i].clear();
70                 danh_sach_ke_07[i].resize(so_dinh_07);
71             }
72             danh_sach_ke_07[0][0] = a;
73             danh_sach_ke_07[0][1] = 0;
74             danh_sach_ke_07[0][2] = 0;
75             danh_sach_ke_07[0][3] = 0;
76             danh_sach_ke_07[0][4] = 0;
77             danh_sach_ke_07[0][5] = 0;
78             danh_sach_ke_07[0][6] = 0;
79             danh_sach_ke_07[0][7] = 0;
80             danh_sach_ke_07[0][8] = 0;
81             danh_sach_ke_07[0][9] = 0;
82             danh_sach_ke_07[0][10] = 0;
83             danh_sach_ke_07[0][11] = 0;
84             danh_sach_ke_07[0][12] = 0;
85             danh_sach_ke_07[0][13] = 0;
86             danh_sach_ke_07[0][14] = 0;
87             danh_sach_ke_07[0][15] = 0;
88             danh_sach_ke_07[0][16] = 0;
89             danh_sach_ke_07[0][17] = 0;
90             danh_sach_ke_07[0][18] = 0;
91             danh_sach_ke_07[0][19] = 0;
92             danh_sach_ke_07[0][20] = 0;
93             danh_sach_ke_07[0][21] = 0;
94             danh_sach_ke_07[0][22] = 0;
95             danh_sach_ke_07[0][23] = 0;
96             danh_sach_ke_07[0][24] = 0;
97             danh_sach_ke_07[0][25] = 0;
98             danh_sach_ke_07[0][26] = 0;
99             danh_sach_ke_07[0][27] = 0;
100            danh_sach_ke_07[0][28] = 0;
101            danh_sach_ke_07[0][29] = 0;
102            danh_sach_ke_07[0][30] = 0;
103            danh_sach_ke_07[0][31] = 0;
104            danh_sach_ke_07[0][32] = 0;
105            danh_sach_ke_07[0][33] = 0;
106            danh_sach_ke_07[0][34] = 0;
107            danh_sach_ke_07[0][35] = 0;
108            danh_sach_ke_07[0][36] = 0;
109            danh_sach_ke_07[0][37] = 0;
110            danh_sach_ke_07[0][38] = 0;
111            danh_sach_ke_07[0][39] = 0;
112            danh_sach_ke_07[0][40] = 0;
113            danh_sach_ke_07[0][41] = 0;
114            danh_sach_ke_07[0][42] = 0;
115            danh_sach_ke_07[0][43] = 0;
116            danh_sach_ke_07[0][44] = 0;
117            danh_sach_ke_07[0][45] = 0;
118            danh_sach_ke_07[0][46] = 0;
119            danh_sach_ke_07[0][47] = 0;
120            danh_sach_ke_07[0][48] = 0;
121            danh_sach_ke_07[0][49] = 0;
122            danh_sach_ke_07[0][50] = 0;
123            danh_sach_ke_07[0][51] = 0;
124            danh_sach_ke_07[0][52] = 0;
125            danh_sach_ke_07[0][53] = 0;
126            danh_sach_ke_07[0][54] = 0;
127            danh_sach_ke_07[0][55] = 0;
128            danh_sach_ke_07[0][56] = 0;
129            danh_sach_ke_07[0][57] = 0;
130            danh_sach_ke_07[0][58] = 0;
131            danh_sach_ke_07[0][59] = 0;
132            danh_sach_ke_07[0][60] = 0;
133            danh_sach_ke_07[0][61] = 0;
134            danh_sach_ke_07[0][62] = 0;
135            danh_sach_ke_07[0][63] = 0;
136            danh_sach_ke_07[0][64] = 0;
137            danh_sach_ke_07[0][65] = 0;
138            danh_sach_ke_07[0][66] = 0;
139            danh_sach_ke_07[0][67] = 0;
140            danh_sach_ke_07[0][68] = 0;
141            danh_sach_ke_07[0][69] = 0;
142            danh_sach_ke_07[0][70] = 0;
143            danh_sach_ke_07[0][71] = 0;
144            danh_sach_ke_07[0][72] = 0;
145            danh_sach_ke_07[0][73] = 0;
146            danh_sach_ke_07[0][74] = 0;
147            danh_sach_ke_07[0][75] = 0;
148            danh_sach_ke_07[0][76] = 0;
149            danh_sach_ke_07[0][77] = 0;
150            danh_sach_ke_07[0][78] = 0;
151            danh_sach_ke_07[0][79] = 0;
152            danh_sach_ke_07[0][80] = 0;
153            danh_sach_ke_07[0][81] = 0;
154            danh_sach_ke_07[0][82] = 0;
155            danh_sach_ke_07[0][83] = 0;
156            danh_sach_ke_07[0][84] = 0;
157            danh_sach_ke_07[0][85] = 0;
158            danh_sach_ke_07[0][86] = 0;
159            danh_sach_ke_07[0][87] = 0;
160            danh_sach_ke_07[0][88] = 0;
161            danh_sach_ke_07[0][89] = 0;
162            danh_sach_ke_07[0][90] = 0;
163            danh_sach_ke_07[0][91] = 0;
164            danh_sach_ke_07[0][92] = 0;
165            danh_sach_ke_07[0][93] = 0;
166            danh_sach_ke_07[0][94] = 0;
167            danh_sach_ke_07[0][95] = 0;
168            danh_sach_ke_07[0][96] = 0;
169            danh_sach_ke_07[0][97] = 0;
170            danh_sach_ke_07[0][98] = 0;
171            danh_sach_ke_07[0][99] = 0;
172        }while(a < 100);
173        if(a > 100) break;
174        danh_sach_ke_07[0].clear();
175        danh_sach_ke_07[0].resize(so_dinh_07);
176        for(int i = 1; i < so_dinh_07; i++){
177            danh_sach_ke_07[i].clear();
178            danh_sach_ke_07[i].resize(so_dinh_07);
179        }
180        danh_sach_ke_07[0][0] = a;
181        danh_sach_ke_07[0][1] = 0;
182        danh_sach_ke_07[0][2] = 0;
183        danh_sach_ke_07[0][3] = 0;
184        danh_sach_ke_07[0][4] = 0;
185        danh_sach_ke_07[0][5] = 0;
186        danh_sach_ke_07[0][6] = 0;
187        danh_sach_ke_07[0][7] = 0;
188        danh_sach_ke_07[0][8] = 0;
189        danh_sach_ke_07[0][9] = 0;
190        danh_sach_ke_07[0][10] = 0;
191        danh_sach_ke_07[0][11] = 0;
192        danh_sach_ke_07[0][12] = 0;
193        danh_sach_ke_07[0][13] = 0;
194        danh_sach_ke_07[0][14] = 0;
195        danh_sach_ke_07[0][15] = 0;
196        danh_sach_ke_07[0][16] = 0;
197        danh_sach_ke_07[0][17] = 0;
198        danh_sach_ke_07[0][18] = 0;
199        danh_sach_ke_07[0][19] = 0;
200        danh_sach_ke_07[0][20] = 0;
201        danh_sach_ke_07[0][21] = 0;
202        danh_sach_ke_07[0][22] = 0;
203        danh_sach_ke_07[0][23] = 0;
204        danh_sach_ke_07[0][24] = 0;
205        danh_sach_ke_07[0][25] = 0;
206        danh_sach_ke_07[0][26] = 0;
207        danh_sach_ke_07[0][27] = 0;
208        danh_sach_ke_07[0][28] = 0;
209        danh_sach_ke_07[0][29] = 0;
210        danh_sach_ke_07[0][30] = 0;
211        danh_sach_ke_07[0][31] =
```