

In Stock I

- ① Acesta rotacioneaza ca si pa vortuta 9 ①

②.

$\frac{f(19,20)}{f(18,21)} \rightarrow \frac{f(18,21)}{f(15,22)} \rightarrow \frac{f(15,22)}{f(8,23)} \rightarrow \frac{f(8,23)}{f(-7,24)}$

return to 24

⑧

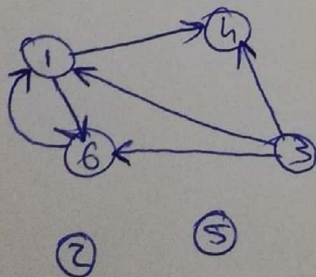
- ③. Se generează toate ~~submulțimile~~ ^{șirurile} de 4 elemente din mulțimea $\{1, 2, 3, 4, 5\}$ cu proprietatea că elementele din mulțimea $\{1, 3, 5\}$ nu se află pe poziții consecutive în șir (nu există 2 numere impare adiacente).
- Determinarea numărului de șiruri de forma $\{2, x, y, 4\}$

Pentru valoarea lui x , se pot lua oricare dintre cele 5 valori din mulțime,
 Pentru valoarea lui y depinde de x , astfel:
 \sum Din cele 5 valori din mulțime

→ dacă x este par, pentru y se pot lua orice valori din mulțime

→ dacă x este impar, pentru y se pot lua doar valorile ~~par~~ pare din mulțime.

Acum numărul de valori pe care mulțimea este 2 , iar cel de valori
impere este 3 , numărul de siruri de forma $\{2, x, y, 4\}$, care respectă
condițiile problemei este egal cu $2 \cdot 5 + 3 \cdot 2 = 10 + 6 = 16$



su sgraffito en ocaseo prop.

ou multiplica \vee

$$\{1, 6, 3, 2, 5\},$$
 $\{1, 6, 3, 2\},$
$$\{1, 0, 3, 5\},$$
$$\{1, 6, 3\}$$

Subiectul II

1. Algoritmul descris în pseudocod producează fiecare pereche de forma (i, j) , $1 \leq j$ cu elementele situate pe poziții egale departe de stânga format din numerele naturale existente în intervalul $[x, y]$, $x \leq y$ și adaugă la sumă ambele numere dacă sunt impare, 0 dacă ambele numere sunt pare sau i , în cazul în care j este impar (și invers)

a. 7 16

$$S = 16 + 8 + 4 + 10 + 12 = 60$$

b. Pt. $(y=20)$

$$S = \underbrace{(11+19)}_{30} + \underbrace{(13+17)}_{30} + \underbrace{(15+15)}_{30} = 90$$

It. $(y=21)$

$$S = \underbrace{(10+20)}_{30} + \underbrace{(12+18)}_{30} + \underbrace{(14+16)}_{30} = 90$$

c. #include <iostream>
using namespace std;
int main()

```
{
    int x, y, i, j, aux, S;
    if (x > y) { aux = x; x = y; y = aux; }
```

```
    i = x; j = y; S = 0;
```

```
    do {
```

```
        S = S + (i%2)*j + (j%2)*i;
```

```
        i++; j--;
```

```
    } while (i <= j);
```

```
    cout << S;
```

```
    return 0;
```

```
}
```

d.

creste x, y

dacă $x > y$ atunci
 $x \leftrightarrow y$

$i \leftarrow x; j \leftarrow y;$

$S \leftarrow (i \% 2) * j + (j \% 2) * i$

$i \leftarrow i + 1; j \leftarrow j - 1$

cat timp $i \leq j$ execută
 $S \leftarrow S + (i \% 2) * j + (j \% 2) * i$
 $i \leftarrow i + 1; j \leftarrow j - 1$

returneaza S

2.

```
struct spectrof{  
    unsigned int cod;  
    unsigned int nrActori;  
    unsigned int vorsta[10];  
} s[20];
```

3.

```
for( i=0; i<4; i++)  
    for( j=0; j<5; j++)  
        a[i][j] = 5*i+5-j;
```