

Задача Lilies

Обяснение върху примера:

Колко най-малко луковици трябва да засади Цветан, за да е сигурен, че в градината му ще има поне по един зюмбюл от всеки цвят

Вход

3

9 6 8

Изход

18

$$18 = (9+6+8) - 6 + 1$$

Задача Lilies

Трябва да се намери сумата S на броя на всички луковици и броя M на тези луковици, който е минимален. Тогава отговорът е $S-M+1$.

Задачата се свежда до намиране на **сума** и на **минимум** на редица от въведени числа.

Понеже не използваме масиви, това трябва да стане едновременно с четенето на данните.

Задача Lilies

```
int n; cin>>n;
int m; cin>>m;
int s=m;
for(int i=2;i<=n;i++)
{int a; cin>>a;
  if(m>a) m=a;
  s=s+a;
}
cout<<s-m+1<<endl;
```

Задача Sequence

Програмата трябва да изведе на стандартния изход числото от най-дългата последователност равни числа, която се среща в редицата

Пример

Вход:

9

15 2 2 2 4 4 4 19 2

Изход:

2

Задача Sequence

Четейки числата от входа, трябва да запазваме в променлива текущо прочететно и да имаме в друга променлива предишно прочетеното

(не използваме масив).

Задача Sequence

Ако текущо прочетеното е равно на предишно прочетеното, увеличаваме с единица брояч, който показва текущата дължина на последователността от равни числа.

Ако текущо прочетеното не е равно на предишно прочетеното, пресмятаме текущата максимална стойност на брояча, зареждаме брояча с единица и полагаме предишната текуща стойност от редицата да е равна на текущата.

Задача Sequence

```
long long t, v, maxv;
int n, c, maxc=1;
cin>>n;
cin>>t; c=1; maxv=t;
for(int i=1;i<n;i++)
{ cin>>v;
  if(v==t) c++;
  else
  { if (c>maxc) {maxc=c;maxv=t;}
    t=v;c=1;
  }
}
if (c>maxc) {maxc=c;maxv=t;}
cout<<maxv<<endl;
```

Двоен цикъл

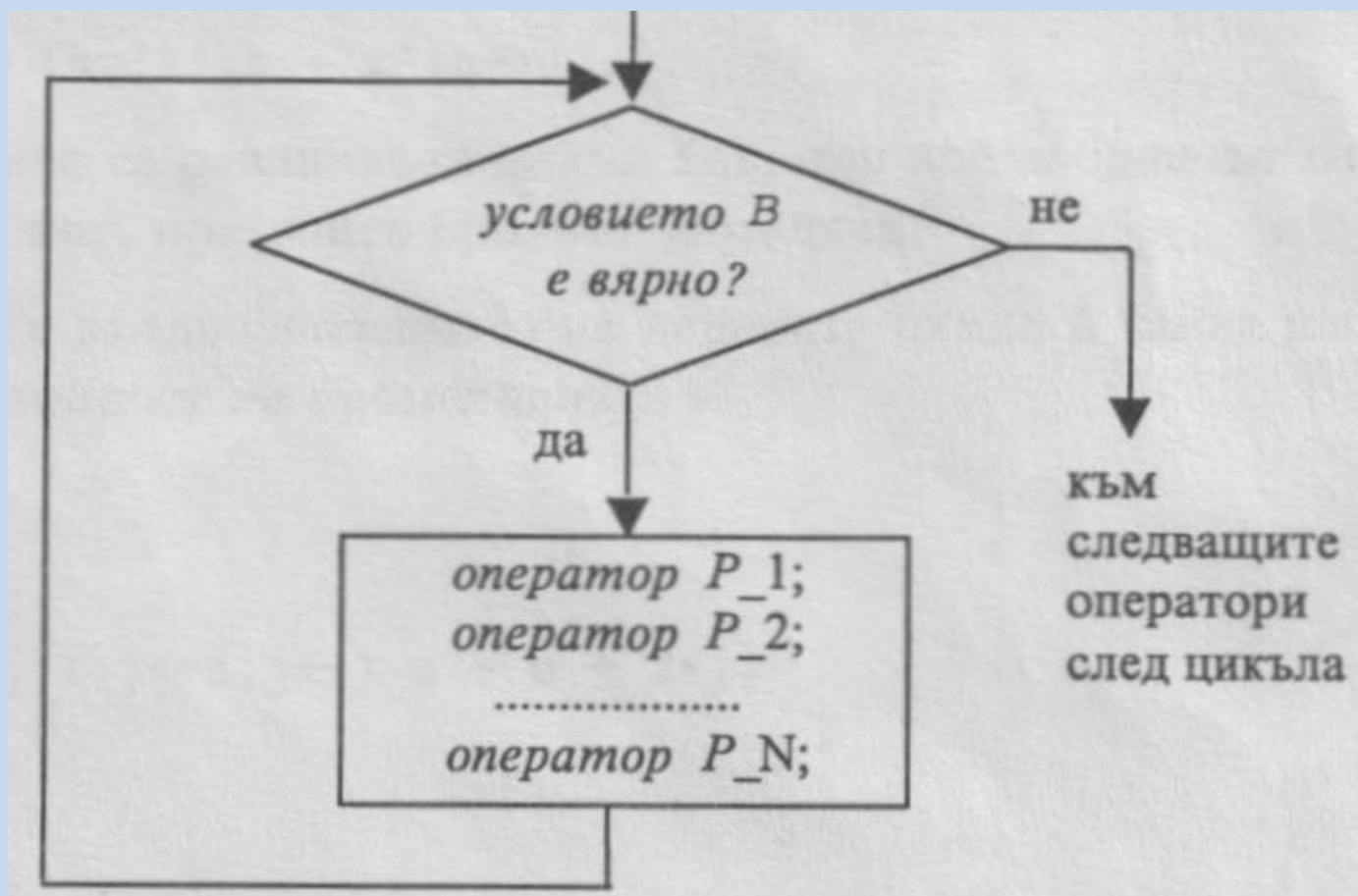
```
#include<iostream>
#include<iomanip>
using namespace std;

int main()
{
    for(int i=1;i<=10;i++)
    {
        for(int j=1;j<=10;j++)
            cout << " " << setw(3) << i*j;
        cout << endl;
    }
}
```


Таблица на ASCII кодове

```
char c; int n;  
for(int i=0;i<=13;i++)  
{ n = 32+16*i;  
  if(n<100) cout << " ";  
  cout << n << ": ";  
  for(int j=0; j<=15; j++)  
  { n=32+16*i+j;  c=n;  
    cout << c << " ";  
  }  
  cout << endl;  
}
```

Цикъл без брояч: Цикъл с условие (пред-условие)

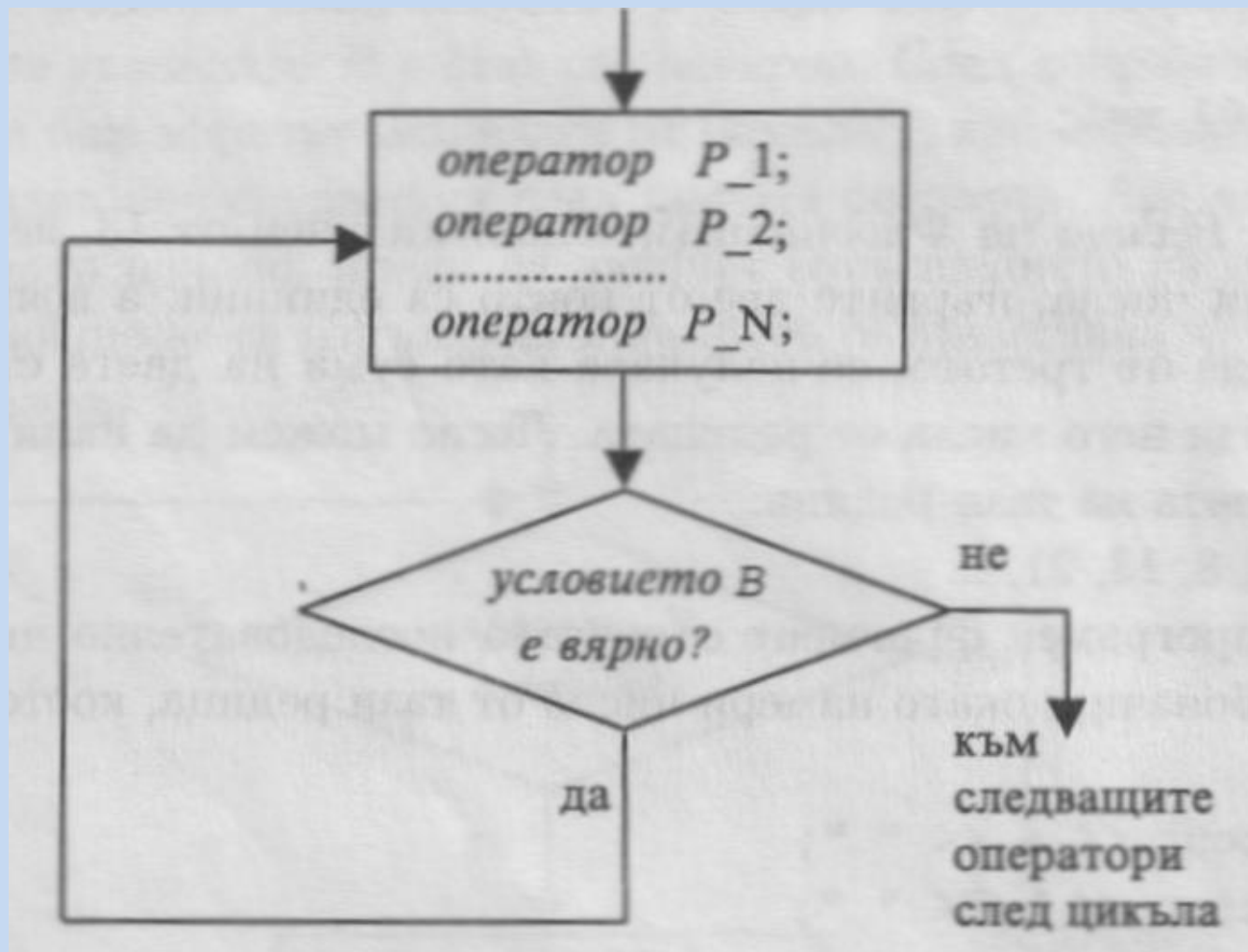


Цикъл с пред-условие

```
int i=1;  
while(i<10)  
{ i++;  
  cout << i << endl;  
}
```

Ще отпечата числата от 2 до 10

Цикъл с пост-условие



Цикъл с пост-условие

```
int i=1;  
do  
{  
    cout << i << endl;  
    i++;  
}  
while(i<10);
```

Ще отпечата числата от 1 до 9

Пример: Безкрайни цикли

```
while(1);
```

```
while(1) { cout << 1;}
```

```
int x=1; while(x>0) x++;
```

Цикъл, чиято продължителност се определя от входните данни:

```
int a=1, s=0;
```

```
while(a!=0) {cin >> a; s += a;}
```

```
cout << s << endl;
```

Чрез оператора while може да програмираме
същото, каквото можем чрез оператора for
(и обратното)

```
for(int i=1; i<N; i++) cout << i << endl;
```

```
int i=1;
```

```
while(i<N) {cout << i << endl; i++;}
```

Числа на Фибоначи

Известната редица:

1 1 2 3 5 8 13 21 34 55 ...

```
int a=1; cout << a << " ";  
int b=1; cout << b << " ";  
while(b<=1000)  
{  
    int c=a+b; cout << c << " ";  
    a=b;  
    b=c;  
}
```


Съчетание на цикли и разклонения

Пример: Цикличен брояч

```
int j=1;
for(int i=1;i<=20;i++)
{
    cout << j << " ";
    j++;
    if(j>5) j=1;
}
```

1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5

Излизане от цикъл чрез break

Пример:

```
while(1)
{int a; cin >> a; if(a==0) break;}
```

Пример:

```
int a; cin >>a;
for(int i=1;i<10;i++)
{a+=i; if(a>20) break; cout << a << " " ;}
```

Изчерващо търсене

Задача: Да се намерят всички цели числа между 1 и 100, които се делят на 7

```
for(int i=1;i<=100;i++)  
if(i%7==0) cout << i << endl;
```

Питагорови тройки цели числа

```
int n=0;  
for(int a=1;a<=1000;a++)  
for(int b=1;b<=1000;b++)  
for(int c=1;c<=1000;c++)  
if(a*a+b*b==c*c) n++;  
cout << n << endl;
```

Усъвършенстване на изчерващото търсене

```
t=clock();  
n=0;  
for(int a=1;a<=1000;a++)  
for(int b=a+1;b<=1000;b++)  
for(int c=b+1;c<=1000;c++)  
if(a*a+b*b==c*c) n++;  
n=2*n;  
cout << n << endl;  
cout << (clock()-t)/1000.0 << endl;
```